

**School of Electrical Engineering, Computing and Mathematical Sciences**

**Discovering Higher-order SNP Interactions in High-dimensional Genomic Data**

**Suneetha Uppu**

**This thesis is presented for the Degree of**

**Doctor of Philosophy**

**of**

**Curtin University**

**October 2018**



## **Declaration**

To the best of my knowledge and belief, this thesis contains no material previously published by any other person, except where due acknowledgment has been made. This thesis contains no material which has been accepted for the award of any other degree or diploma in any University.

Suneetha Uppu

School of Electrical Engineering, Computing and Mathematical Sciences

October 2018



To my daughter Meena

And my husband Prasanna

Their love, support and patience will never be forgotten.



# Acknowledgements

The completion of this thesis has been a long and arduous journey. I would take this opportunity to convey my gratitude to all the people without whom I could never have completed this thesis.

In the first place I would like to express my sincere appreciation and deepest gratitude to my supervisor, Associate Professor Aneesh Krishna. His consistent support in the research exceptionally inspired and enriched my growth as a student and a researcher. I am extremely grateful for his supervision and guidance from the very early stage of this research. He has never failed to motivate and show confidence in my ability through most of my difficult times. His invaluable insights, comments and ideas made him a backbone of completing this research. I gratefully acknowledge Dr.Raj P.Gopalan, my co-supervisor, for unflinching encouragement and support in various ways. I am indebted for his valuable advice, guidance, and supervision. I sincerely thank him for reviewing this thesis and sharing his knowledge.

I convey special acknowledgement to Dr.John Wallace from Ritchie Lab, and Dr.Ryan Urbanowicz Department of Biostatistics and Epidemiology, University of Pennsylvania, for their expert assistance in simulating the datasets. Special thanks go to Dr.John Fielder, Curtin University for his excellent grammar skills. We further thank all the anonymous reviewers of all our publications for their valuable suggestions and comments, which helped us to improve the quality of this research to a great extent.

I acknowledge the generosity of the financial and administrative support that I received from the School of Electrical Engineering, Computing, and Mathematical Sciences to attend conferences during my candidature.

I am deeply indebted to my amazing husband Prasanna, for his love and tremendous support over the course of this study. His patience and understanding has provided me to use every opportunity for being successful over the years. My deep and heartfelt appreciation goes to my daughter, Meena for spending most of her school breaks and holidays along with me in the University. I owe her greatly. It is a pleasure to express my gratitude wholeheartedly to my mum for her love and unconditional support while writing up this thesis. My dad and brother deserve special mention for their inseparable support and courage that they provided throughout my studies.

I convey special thanks to my friends Gladis and Soma for consistent support and encouragement over the years. Many thanks to my friends Chitra, Sreenithya, and Qiao for the wonderful time we shared. Finally, I would like to pay tribute to everybody who has helped me either directly or indirectly in the completion of this thesis.



# **Abstract**

The advancements in sequencing high-throughput human genome and computational abilities have tremendously improved the understanding of the genetic architecture behind the complex diseases. These advances led to the identification of a number of single nucleotide polymorphisms (SNPs) associated with complex diseases. Identifying and characterizing these SNPs are being intensively studied in this new era of genome-wide association studies (GWAS). These univariate studies ignore interaction effects between SNPs. It is widely believed that the etiology of most complex diseases depends on interactions between genetic variants and/ or environmental factors. Exposing these interaction effects are challenging due to biomolecular complexities, the high-dimensionality problem, computational limitations, and the presence of noise in the biological data. Several machine learning and data mining approaches have been consistently successful in addressing these challenges. However, these traditional approaches are still left with several caveats when searching for useful subsets of SNP interactions.

In this thesis, an associative classification (AC) based approach is implemented to identify multi-locus SNP interactions. In light of findings and difficulties observed during the experimental evaluations, a multifactor dimensionality reduction (MDR) based AC was proposed by reducing the dimensionality of the data. These strategies based on conventional methods are yet to produce remarkable results in identifying higher-order SNP interactions. Then the focus of this thesis is turned to explore the application of deep learning techniques by providing new clues into the interaction analysis. A deep neural network (DNN) is trained to detect informative interactions between SNPs. The performance of the deep learning method is maximized by improving network learning, and optimizing hyper-parameters. Further, DNN is unified with a random forest (RF) for achieving reliable interactions in the presence of noise. These methods are evaluated on various balanced and imbalanced simulated datasets for different epistasis models. The performance of these proposed methods are validated, and compared with the previous methods in the presence and absence of noise. Finally, the experimental findings are confirmed over four real world data applications.

# Publications

The work presented in this thesis has been previously appeared in the following publications:

## **Journal Articles**

1. S. Uppu, A. Krishna, and "A deep hybrid model to detect multi-locus interacting SNPs in the presence of noise", *International Journal of Medical Informatics*, vol. 119, pp. 134-151, 2018.
2. S. Uppu, A. Krishna, and R. Gopalan, "A review on methods for detecting SNP interactions in high-dimensional genomic data", *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 15 (2), pp. 599 - 612, 2018.
3. S. Uppu, A. Krishna, and R. Gopalan, "Combining associative classification with multifactor dimensionality reduction for predicting higher-order SNP interactions in case-control studies", *International Journal of Computational Intelligence in Bioinformatics and Systems Biology*, 22nd October 2017 (accepted - under production).
4. S. Uppu, A. Krishna, and R. P. Gopalan, "A Deep Learning Approach to Detect SNP Interactions", *Journal of Software (JSW)*, vol. 11, pp. 965-975, 2016.
5. S. Uppu and A. Krishna, "Evaluation of associative classification-based multifactor dimensionality reduction in the presence of noise", *Network Modeling Analysis in Health Informatics and Bioinformatics*, vol. 5, pp. 1-9, 2016.
6. S. Uppu, A. Krishna, and R. P. Gopalan, "Rule-based analysis for detecting epistasis using associative classification mining", *Network Modeling Analysis in Health Informatics and Bioinformatics*, vol. 4, pp. 1-19, 2015.
7. S. Uppu, A. Krishna, and R. P. Gopalan, "Detecting SNP Interactions in Balanced and Imbalanced Datasets using Associative Classification", *Australian Journal of Intelligent Information Processing Systems*, vol. 14, 2014.

## **Conference Papers**

1. S. Uppu and A. Krishna, "An intensive search for higher-order gene-gene interactions by improving deep learning model", in *18th IEEE International Conference on Bioinformatics and Bioengineering (BIBE)*, 2018, pp.104-109.
2. S. Uppu and A. Krishna, "Tuning Hyper-parameters for Gene Interaction Models in Genome-Wide Association Studies", in *International Conference on Neural Information Processing*, 2017, pp. 791-801.
3. S. Uppu and A. Krishna, "Improving strategy for discovering interacting genetic variants in association studies," in *International Conference on Neural Information Processing*, 2016, pp. 461-469.
4. S. Uppu, A. Krishna, and R. P. Gopalan, "Towards Deep Learning in genome-Wide Association Interaction studies", in *Pacific Asia Conference on Information Systems (PACIS)*, 2016, p. 20.
5. S. Uppu, A. Krishna, and R. P. Gopalan, "A Multifactor Dimensionality Reduction Based Associative Classification for Detecting SNP Interactions", in *Neural Information Processing*, 2015, pp. 328-336.
6. S. Uppu, A. Krishna, and R. P. Gopalan, "An Associative Classification Based Approach for Detecting SNP-SNP Interactions in High-dimensional Genome," in *IEEE International Conference on Bioinformatics and Bioengineering (BIBE)*, 2014, pp. 329-333.
7. S. Uppu and A. Krishna, "Convolutional model for predicting gene-gene interactions," in *25th International Conference on Neural Information Processing (ICONIP)*, pp.127-137, 2018.

# Contents

<b>Acknowledgements</b>	vi
<b>Abstract</b>	viii
<b>Publications</b>	ix
<b>List of Figures</b>	xvi
<b>List of Tables</b>	xxii
<b>List of Acronyms</b>	xxv
<b>1. Introduction</b>	1
1.1 Research Questions	4
1.2 Research Objectives	4
1.3 Research Outcomes and Contributions	5
1.4 Thesis Structure	7
<b>2. Literature Survey</b>	9
2.1 Single Nucleotide Polymorphism	10
2.2 Interactions	11
2.2.1 Gene-Environment Interactions	12
2.2.2 Gene-Gene Interactions	12
2.3 Issues to be considered for designing the models	16
2.3.1 Variable Selection	16
2.3.2 Model Building	18
2.3.3 Model Interpretation	19
2.4 Methods for detecting SNP interactions	20
2.4.1 Multifactor Dimensionality Reduction	20
2.4.2 Random Forest	22
2.4.3 Neural Networks	24
2.4.4 Support Vector Machines	25
2.4.5 Regression Models	27
2.4.6 Bayesian Approaches	28
2.4.7 Ant Colony Optimization Approaches	29
2.4.8 Other methods	30

2.5 Summary of methods in view of searching strategies	35
2.5.1 Exhaustive Search Methods	36
2.5.2 Stochastic search methods	36
2.5.3 Heuristic search methods	37
2.6 Data simulation and Real-world data for evaluating the models	37
2.6.1 Coalescent Simulation	38
2.6.2 Forward-time Simulation	39
2.6.3 Resampling Simulation	39
2.6.4 Real-world Data Applications	42
2.7 Alternative Methods	43
2.7.1 Biological Filters	43
2.7.2 Computational Optimization	45
2.7.3 Pathway Approaches	47
2.8 Chapter Summary	48
<b>3. Associative classification for detecting higher-order SNP Interactions</b>	<b>49</b>
3.1 Associative classification	51
3.1.1 Workflow of the method	53
3.1.2 Definitions	54
3.1.3 Data Representation	55
3.1.4 Rule generation	56
3.1.5 Classifier	57
3.2 Data Simulation	57
3.2.1 Scenario I	58
3.2.2 Scenario II	59
3.3 Real data	63
3.4 Data Analysis	64
3.5 Evaluation of the Approach	65
3.5.1 Preliminary results for two-locus interactions on balanced and imbalanced simulated datasets	66
3.5.1.1 Scenario I	66
3.5.1.2 Scenario II	69
3.5.2 Evaluation for main effect and higher-order interactions	77
3.5.3 Evaluation on Real data	86
3.6 Chapter Summary	87

<b>4. Multifactor Dimensionality Reduction based Associative classification</b>	88
4.1 Integrating AC into MDR (MDRAC)	90
4.2 Datasets	94
4.2.1 Simulated Datasets in the absence of noise	94
4.2.2 Simulated Datasets in the presence of noise	94
4.2.3 Real Datasets	95
4.3 Data Analysis	95
4.4 Evaluation on the proposed method	97
4.4.1 Analysis of simulated data	98
4.4.2 Analysis of real data	107
4.4.3 Discussion	118
4.5 Evaluation and discussion of MDRAC in the presence of genotyping error, missing data, phenocopy, and genetic heterogeneity.	121
4.6 Chapter Summary	127
<b>5. Towards Deep Learning Models for SNP Interaction studies</b>	129
5.1 Deep Neural Networks	130
5.1.1 Forward propagation	132
5.1.2 Backpropagation	135
5.1.3 Mean square error estimate for logistic function	141
5.1.4 Cross entropy error estimate for logistic function	142
5.2 Deep Learning Method	144
5.2.1 Case-control Data	145
5.2.2 Multifactor combinations and Cross-validation	145
5.2.3 Data partition and data analysis	145
5.2.4 Deep learning algorithm	146
5.2.5 Classification for more than two classes	147
5.2.6 Other activation functions	148
5.2.7 Dropout	149
5.2.8 Estimation of variable importance	149
5.3 Evaluation over whole genome data	150
5.4 Evaluations on sporadic breast cancer data for two-locus interactions	155
5.4.1 Evaluation and analysis of the proposed method	155
5.4.2 Evaluation and analysis of the proposed method by changing the parameters	158

5.4.3 Prediction accuracy of the proposed method compared with previous methods	160
5.5 Chapter Summary	164
<b>6. Improving Deep Learning Method for an intensive search of higher-order Interactions</b>	165
6.1 Extended Method	166
6.1.1 Improving learning	169
6.1.1.1 Regularization	169
6.1.1.2 Weight initialization	170
6.1.1.3 Learning rate and batch size	170
6.1.1.4 Momentum and adaptive learning rate	170
6.1.2 Optimising hyper-parameters	171
6.1.2.1 Grid Search	171
6.1.2.2 Random Grid Search	172
6.1.3 Sensitivity analysis	172
6.1.4 Dimensionality reduction	172
6.1.5 Case-control Data	173
6.2 Evaluations for higher-order interactions	174
6.3 Evaluations for unsupervised learning	181
6.4 Evaluations for optimising hyper-parameters	184
6.5 Evaluations by improving learning	192
6.6 Chapter summary	195
<b>7. A Deep Hybrid Method for the noise data</b>	197
7.1 Deep Hybrid Method	198
7.1.1 Parametrisation of Deep Neural Networks	199
7.1.2 Back Propagation	202
7.1.3 Random Forest	203
7.2 Variable importance	206
7.3 SNP Interactions	207
7.4 Simulated datasets	208
7.4.1 Simulated datasets 1	208
7.4.2 Simulated datasets 2	209
7.5 Evaluations	211
7.6 Experimental Results	211

7.6.1 Simulated Studies	211
7.6.1.1 Simulation Study 1	211
7.6.1.2 Simulation Study 2	215
7.6.2 Real world data application	221
7.7 Discussions	226
7.8 Chapter Summary	227
<b>8. Conclusions and Future works</b>	229
8.1 Summary of Contributions	230
8.2 Limitations and Future work	232
8.2.1 Population Stratification	232
8.2.2 Power calculations	232
8.2.3 Training data	233
8.2.4 Activation Function	233
8.2.5 Huge network parameters	234
8.2.6 Presence of Noise	234
8.2.7 Other future directions	234
<b>Bibliography</b>	236
<b>Glossary</b>	257
<b>Appendix</b>	260

# List of Figures

<b>Figure.2.1</b>	Single nucleotide polymorphism in DNA sequence	11
<b>Figure.2.2</b>	Relationship between Biological epistasis and Statistical epistasis	14
<b>Figure.2.3</b>	Methods based on searching strategies to detect epistasis	35
<b>Figure.3.1</b>	Steps involved in Associative classification	52
<b>Figure.3.2</b>	Workflow of the implemented Associative classifier to detect SNP interactions	53
<b>Figure.3.3</b>	Accuracy of 6 models with 1:1 ratio	68
<b>Figure.3.4</b>	Accuracy of 6 models with 1:2 ratio	68
<b>Figure.3.5</b>	Accuracy of 6 models with 1:4 ratio	68
<b>Figure.3.6</b>	Accuracy of 6 models with 1:6 ratio	68
<b>Figure.3.7</b>	Accuracy of 70 models with ratio 1:1 for MAF 0.2	74
<b>Figure.3.8</b>	Accuracy of 70 models with ratio 1:1 for MAF 0.4	75
<b>Figure.3.9</b>	Accuracy of 70 models with ratio 1:2 for MAF 0.2	75
<b>Figure.3.10</b>	Accuracy of 70 models with ratio 1:2 for MAF 0.4	76
<b>Figure.3.11</b>	Accuracy of 70 models with ratio 1:4 for MAF 0.2	76
<b>Figure.3.12</b>	Accuracy of 70 models with ratio 1:4 for MAF 0.4	77
<b>Figure.3.13</b>	Single-locus analysis	78
<b>Figure.3.14</b>	Two-locus analysis	79
<b>Figure.3.15</b>	Three-locus analysis	79
<b>Figure.3.16</b>	Four-locus analysis	80
<b>Figure.3.17</b>	Five-locus analysis	81
<b>Figure.3.18</b>	Six-locus analysis	81
<b>Figure.3.19</b>	Multi-locus analysis	82
<b>Figure.4.1</b>	Workflow representation of the proposed MDRAC method	90
<b>Figure.4.2</b>	Summary of general steps involved in MDRAC	91
<b>Figure.4.3</b>	Single-locus analysis for ratios 1:1, 1:2, and 1:4, sample size 400, 800, and 1600, and MAF 0.2, and 0.4	98
<b>Figure.4.4</b>	Two-locus analysis for ratios 1:1, 1:2, and 1:4, sample size 400, 800, and 1600, and MAF 0.2, and 0.4	99

<b>Figure.4.5</b>	Three-locus analysis for ratios 1:1, 1:2, and 1:4, sample size 400, 800, and 1600, and MAF 0.2, and 0.4	100
<b>Figure.4.6</b>	Four-locus analysis for ratios 1:1, 1:2, and 1:4, sample size 400, 800, and 1600, and MAF 0.2, and 0.4	101
<b>Figure.4.7</b>	Five-locus analysis for ratios 1:1, 1:2, and 1:4, sample size 400, 800, and 1600, and MAF 0.2, and 0.4	102
<b>Figure.4.8</b>	Six-locus analysis for ratios 1:1, 1:2, and 1:4, sample size 400, 800, and 1600, and MAF 0.2, and 0.4	102
<b>Figure.4.8</b>	Summary of multi-locus analysis for ratios 1:1, 1:2, and 1:4 sample size 400, 800, and 1600	103
<b>Figure.4.9</b>	Graphical representation of cell values of five-locus genotype combinations of MDR analysis over sporadic breast cancer data	110
<b>Figure.4.10</b>	Graphical representation of cell values of five-locus genotype combinations of MDR analysis over hypertension data	111
<b>Figure.4.11</b>	Entropy circular graph of five-locus genotype combinations of MDR analysis over sporadic breast cancer data and hypertension data	112
<b>Figure.4.12</b>	Graphical representation of cell values of five-locus genotype combinations of MDRAC analysis over sporadic breast cancer data	113
<b>Figure.4.13</b>	Graphical representation of cell values of three-locus genotype combinations of MDRAC analysis over hypertension data	114
<b>Figure.4.14</b>	Entropy graph of MDRAC analysis over sporadic breast cancer and hypertension data	114
<b>Figure.4.15</b>	Entropy graph of MDRAC analysis in the presence of noise over breast cancer and hypertension data	115
<b>Figure.4.16</b>	Entropy graph of MDRAC analysis by adjusting threshold value over breast cancer and hypertension data	115
<b>Figure.4.17</b>	Entropy graph of MDRAC analysis by adjusting threshold value in the presence of noise over breast cancer and hypertension data	116
<b>Figure.4.18</b>	Accuracy of MDRAC compared with other approaches over sporadic breast cancer data	117
<b>Figure.4.19</b>	Accuracy of MDRAC compared with other approaches over hypertension data	118
<b>Figure.4.20</b>	Graphical representation of cell values of four-locus genotype combinations of MDRAC analysis over whole genome association data.	123

<b>Figure.4.21</b>	Graphical representation of cell values of three-locus genotype combinations of MDR analysis over whole genome association data	125
<b>Figure.4.22</b>	Dendogram diagram of three-locus interaction model of MDR analysis over whole genome association data	125
<b>Figure.4.23</b>	Dendogram diagram of three-locus interaction model of MDRAC analysis over whole genome association data	126
<b>Figure.5.1</b>	Biological structure of a neuron	130
<b>Figure.5.2</b>	Structure of a neuron in a deep neural networks	131
<b>Figure.5.3</b>	An example of a deep neural network	131
<b>Figure.5.4</b>	Basic structure of a four-layer feedforward network	132
<b>Figure.5.5</b>	Example of a feedforward propagation of a four-layer feedforward network	133
<b>Figure.5.6</b>	Example of a backpropagation of a four-layer feedforward network	135
<b>Figure.5.6.1</b>	Path-1 between output and input layer	136
<b>Figure.5.6.2</b>	Path-2 between output layer and input layer	136
<b>Figure.5.6.3</b>	Path-3 between output layer and input layer	136
<b>Figure.5.6.4</b>	Path-4 between output layer and input layer	137
<b>Figure.5.6.5</b>	Path-5 between output layer and input layer	137
<b>Figure.5.6.6</b>	Path-6 between output layer and input layer	137
<b>Figure.5.6.7</b>	Path-7 between output layer and first hidden layer	139
<b>Figure.5.6.8</b>	Path-8 between output layer and first hidden layer	139
<b>Figure.5.6.9</b>	Path-9 between output layer and first hidden layer	139
<b>Figure.5.6.10</b>	Path-10 between output layer and second hidden layer	140
<b>Figure.5.6.11</b>	Path-11 between output layer and second hidden layer	140
<b>Figure.5.7</b>	Overview of the deep learning method	144
<b>Figure.5.8</b>	ROC curve for training, validation and testing	151
<b>Figure.5.9</b>	Scoring history of samples vs metrics of the deep learning model	151
<b>Figure.5.10</b>	Predicting test data on the deep learning model	152
<b>Figure.5.11</b>	Top 20 two-locus SNP interactions identified by the deep learning model	153
<b>Figure.5.12</b>	Comparing metrics of the deep learning model with GBM, RF, and LR	154
<b>Figure.5.13</b>	Accuracy of the deep learning model compared with the previous approaches	154

<b>Figure.5.14</b>	Model metrics the deep learning model by changing activation functions	156
<b>Figure.5.15</b>	Top 20 single-locus, and two-locus SNP interactions	156
<b>Figure.5.16</b>	Performance of the model while training, validating, and testing	157
<b>Figure.5.17</b>	Scoring history of epochs vs metrics of the model	157
<b>Figure.5.18</b>	Training speed vs Training time of the model by varying parameters which includes, Network topology, Scoring selections, Adaptive learning rate, and Training samples per iteration	159
<b>Figure.5.19</b>	Training speed vs Training time of the model by varying parameters Training time vs Test set error to obtain the model with low test set error, and AUC in distributed model	160
<b>Figure.5.20</b>	Performance of MDR. A) Allocation of high-risk and low-risk cells in the two-locus contingency table of genotype combinations. B) The line graph represents the overall adjusted balanced accuracy of top two-locus interacting models. C) An interaction dendrogram summarising the information gain associated with constructing pairs of SNPs	161
<b>Figure.5.21</b>	Performance of LR. A) The graph plots the variable importance of sporadic breast cancer data using LR analysis. B) The graph shows the proportions of models that contain the interactions of interest C) The top 5 important interactions were identified by Logic Regression	162
<b>Figure.5.22</b>	Scoring history of deep learning method, RF, GBM, and LR	163
<b>Figure.5.23</b>	Prediction accuracy compared with other machine Learning methods	163
<b>Figure.6.1</b>	Overview of the extended deep learning method	166
<b>Figure.6.2</b>	ROC plots for one-locus to ten-locus interactions	174
<b>Figure.6.3</b>	ROC plots of higher-order interaction model a) AUC for training data b) AUC for validation data	179
<b>Figure.6.4</b>	Scoring metrics of higher-order interaction model	179
<b>Figure.6.5</b>	Top 20 higher-order SNP-interactions for sporadic breast cancer data	181
<b>Figure.6.6</b>	PCA on single-locus SNP	182
<b>Figure.6.7</b>	PCA on two-locus SNP interactions	182
<b>Figure.6.8</b>	a) Deviation plot and b) Cumulative variance plot of PCA for higher-order interactions	183
<b>Figure.6.9</b>	Cutoff offset plot for unsupervised higher-order model learning	183
<b>Figure.6.10</b>	Reconstruction error plot for unsupervised higher-order	184

	model learning	
<b>Figure.6.11</b>	Model metrics (accuracy, auc, err, logloss, mse, precision, recall, and specificity) of DNN using Grid Search on sporadic breast cancer data.	185
<b>Figure.6.12</b>	Model metrics (accuracy, auc, err, logloss, mse, precision, recall, and specificity) of DNN using Random Grid Search on sporadic breast cancer data	186
<b>Figure.6.13</b>	Prediction of the test data on the best model for sporadic breast cancer data	186
<b>Figure.6.14</b>	Model metrics (accuracy, auc, err, logloss, mse, precision, recall, and specificity) of DNN using Grid Search on hypertension data	187
<b>Figure.6.15</b>	Model metrics (accuracy, auc, err, logloss, mse, precision, recall, and specificity) of DNN using Random Grid Search on hypertension data	187
<b>Figure.6.16</b>	Prediction of the test data on the best model for hypertension data	188
<b>Figure.6.17</b>	Model performance on breast cancer data (auc vs models)	189
<b>Figure.6.18</b>	Model performance on breast cancer data (mse vs models)	189
<b>Figure.6.19</b>	Model performance on hypertension data (auc vs models)	190
<b>Figure.6.20</b>	Model performance on hypertension data (mse vs models)	190
<b>Figure.6.21</b>	Top 30 SNP interactions of sporadic breast cancer data	191
<b>Figure.6.22</b>	Top 30 SNP interactions of hypertension data	191
<b>Figure.6.23</b>	Performance analysis of improved DNN for two-locus SNP interactions. <b>(a)</b> True positive rate (tpr) vs false positive rate (fpr) for both training and validation, <b>(b)</b> scoring history of classification error at epochs, <b>(c)</b> scoring history of logloss at epochs, <b>(d)</b> scoring history of rmse at epochs, <b>(e)</b> scoring history of area under curve (auc) for training and validation in epochs, <b>(f)</b> accuracy plot at which maximum accuracy is obtained for a threshold value, <b>(g)</b> Precision plot at which maximum accuracy is obtained for a threshold value, and <b>(h)</b> Testing performance	193
<b>Figure.6.24</b>	Sensitivity analysis of two-locus SNP interactions	194
<b>Figure.6.25</b>	Top 20 two-locus SNP interactions	194
<b>Figure.6.26</b>	Accuracy of DNN compared previous approaches	195
<b>Figure.7.1</b>	Overview of the proposed hybrid deep neural network (DNN) – Random forest (RF) method	198
<b>Figure.7.2</b>	An example of Deep neural networks (DNN) with an input layer $s$ , multiple hidden layers $h$ , and an output layer $y$ is parameterized by $\varphi$ , <b>(b)</b> An autoencoder network used	201

to train the initial parameters of the first layer of a deep neural network, (c) A single layered autoencoder network to train the initial parameters of the second layer, (d) An autoencoder that finds the initial parameters for  $L^{\text{th}}$  hidden layer, (e) Classification using Random forest

<b>Figure.7.3</b>	An example of a decision tree in the random forest to identify interactions between SNPs	204
<b>Figure.7.4</b>	Hybrid DNN-RF model	205
<b>Figure.7.5</b>	Comparison of power of DNN (Table 7.2) and DNN-RF (Table 7.3) in the presence of noise for the balanced datasets	215
<b>Figure.7.6</b>	Power of DNN-RF for ratio 1:1 for various minor allele frequencies, heritability, and sample size	219
<b>Figure.7.7</b>	Power of DNN-RF for ratio 1:2 for various minor allele frequencies, heritability, and sample size	219
<b>Figure.7.8</b>	Power of DNN-RF for ratio 1:4 for various minor allele frequencies, heritability, and sample size	220
<b>Figure.7.9</b>	Performance analysis of DNN-RF hybrid method for two-locus SNP interactions. (a) True positive rate (tpr) vs false positive rate (fpr) for both training and validation, (b) accuracy plot at which maximum accuracy is obtained for a threshold value, (c) accuracy vs precision plot during training and validation, (d) Precision plot at which maximum accuracy is obtained for a threshold value, (e) scoring history of logloss at n trees, (f) scoring history of mse at n trees, (g) scoring history of area under curve (auc) for training and validation in n trees, and (h) scoring history of classification error	222
<b>Figure.7.10</b>	Top 30 two-way SNP interactions identified by DNN-RF model on Chronic dialysis patients' dataset	223
<b>Figure.7.11</b>	Accuracy of hybrid DNN-RF model compared with some of the previous algorithms	223
<b>Figure.7.12</b>	Top 30 three-way SNP interactions identified by DNN-RF model on Chronic dialysis patients' dataset	226

# List of Tables

<b>Table 2.1</b>	Extended types of epistasis	14
<b>Table 2.2</b>	Variable selection methods in detecting SNP interactions	17
<b>Table 2.3</b>	Data mining and machine learning approaches for detecting SNP interactions	32
<b>Table 2.4</b>	Data simulators for evaluating the models	40
<b>Table 2.5</b>	Biological knowledge based databases	44
<b>Table 3.1</b>	Epistasis models exhibiting interactions between two SNPs in the absence of main effects.	58
<b>Table 3.2</b>	Penetrance table for 70 epistasis models generated in simulated scenario II	60
<b>Table 3.3</b>	Accuracy of 6 models with 1:1 ratio of cases and controls	66
<b>Table 3.4</b>	Accuracy of 6 models with 1:2 ratio of cases and controls	67
<b>Table 3.5</b>	Accuracy of 6 models with 1:4 ratio of cases and controls	67
<b>Table 3.6</b>	Accuracy of 6 models with 1:6 ratio of cases and controls	67
<b>Table 3.7</b>	Accuracy of 70 models with 400 samples of 1:1 ratio of cases and controls	69
<b>Table 3.8</b>	Accuracy of 70 models with 400 samples of 1:2 ratio of cases and controls	71
<b>Table 3.9</b>	Accuracy of 70 models with 400 samples of 1:4 ratio of cases and controls	72
<b>Table 3.10</b>	Accuracy of single-locus models	83
<b>Table 3.11</b>	Accuracy of two-locus models	83
<b>Table 3.12</b>	Accuracy of three-locus models	84
<b>Table 3.13</b>	Accuracy of four-locus models	84
<b>Table 3.14</b>	Accuracy of five-locus models	85
<b>Table 3.15</b>	Accuracy of Six-locus models	85
<b>Table 3.16</b>	Average balanced accuracy of single-locus to six-locus models	85
<b>Table 3.17</b>	Evaluation of sporadic breast cancer	86
<b>Table 4.1</b>	Evaluation for single-locus models	104
<b>Table 4.2</b>	Evaluation for two-locus models	104

<b>Table 4.3</b>	Evaluation for three-locus models	105
<b>Table 4.4</b>	Evaluation for four-locus models	105
<b>Table 4.5</b>	Evaluation for five-locus models	106
<b>Table 4.6</b>	Evaluation for six-locus models	106
<b>Table 4.7</b>	Average balanced accuracy of one-locus to six-locus models	106
<b>Table 4.8</b>	Experimental results of sporadic breast cancer data on cross-validation	108
<b>Table 4.9</b>	Experimental results of hypertension data on cross-validation	108
<b>Table 4.10</b>	Experimental results of sporadic breast cancer data	109
<b>Table 4.11</b>	Experimental results of hypertension data	109
<b>Table 4.12</b>	Accuracy of MDRAC over previous algorithms	116
<b>Table 4.13</b>	Accuracy of MDRAC to detect two-locus SNP interactions	122
<b>Table 4.14</b>	Power of MDRAC to detect two-locus SNP interactions	122
<b>Table 4.15</b>	Best model predicted for the data obtained from the whole genome association study	124
<b>Table 5.1</b>	Metrics of the deep learning method compared with the previous approaches	161
<b>Table 6.1</b>	Top 10 single-locus SNPs	174
<b>Table 6.2</b>	Top 10 two-locus SNPs	175
<b>Table 6.3</b>	Top 10 three-locus SNPs	175
<b>Table 6.4</b>	Top 10 four-locus SNPs	175
<b>Table 6.5</b>	Top 10 five-locus SNPs	176
<b>Table 6.6</b>	Top 10 six-locus SNPs	176
<b>Table 6.7</b>	Top 10 seven-locus SNPs	177
<b>Table 6.8</b>	Top 10 eight-locus SNPs	177
<b>Table 6.9</b>	Top 10 nine-locus SNPs	178
<b>Table 6.10</b>	Top 10 ten-locus SNPs	178
<b>Table 6.11</b>	Top 10 higher-order SNPs	180
<b>Table 7.1</b>	Epistasis models generated using various penetrance functions, allele frequencies p and q, and heritability H using GAMETES tool	209
<b>Table 7.2</b>	Power of DNN in various simulated scenarios among six models to detect two-locus SNP interactions	213
<b>Table 7.3</b>	Power of DNN-RF in various simulated scenarios among six models to detect two-locus SNP interactions	214

<b>Table 7.4</b>	Power of DNN-RF in various simulated scenarios among 24 models to detect two-locus SNP interactions	215
<b>Table 7.5</b>	Summary of power of DNN-RF to detect known two-locus SNP interactions	220
<b>Table 7.6</b>	W-test for predicted two-locus SNP interactions	224

## List of Acronyms

ACO	Ant Colony Optimization
GWAS	Genome Wide Association Studies
SNP	Single Nucleotide Polymorphisms
GWAIS	Genome Wide Association Interaction Studies
MDR	Multifactor Dimensionality Reduction
SVM	Support Vector Machines
NNs	Neural Networks
CARTs	Classification and Regression Trees
RF	Random Forest
GE	Genotyping Error
MS	Missing data
PC	Phenocopy
GH	Genetic Heterogeneity
AC	Associative Classification
MDR-AC	Multifactor Dimensionality Reduction based Associative Classification
DNNs	Deep Neural Networks
DNN-RF	Random forest integrated with Deep Neural Network
DNA	Deoxyribonucleic Acid
H	Heritability
G*E	Gene-Environment Interactions
G*G	Gene-Gene Interactions
OOB	Out-Of-Bag
LR	Logistic Regression
LogicFS	Logic Feature selection
LD	Linkage Disequilibrium

PCA	Principal Component Analysis
KDD	Knowledge Discovery from Databases
ARM	Association Rule Mining
CARs	Class Association Rules
CPAR	Classification based on Predictive Association
LAC	Lazy Associative Classifier
TP	True Positive rate
FP	False positive rate
FN	False negative rate
MAFs	Minor Allele Frequencies
CBA	Classification based on Associations
CVC	Cross Validation Consistency
MLPs	Multilayer Perceptrons
SGD	Stochastic Gradient Descent
GBM	Gradient Boosted Machines
MDA	Mean Decrease in Accuracy



# Chapter 1

## Introduction

In the current era of big genome data, there has been growing interest in identification and characterisation of genotype-phenotype relations to reveal the susceptibility of complex diseases. Many researchers have focused on genome-wide association studies (GWAs) to interpret the genetic architecture behind the manifestation of a disease. Single nucleotide polymorphisms (SNPs) have become most commonly used biomarkers in GWAs. Several hundreds to more than ten million SNPs from different individuals have been investigated to discover the underlying causes of traits [1]. Predominantly, GWAs are univariate analysis, which depends on a single-locus SNPs, leading to the problem of “missing heritability” [2]. The conventional single-locus SNP methods, such as, Chi-squared test or Fisher’s exact test unravel genetic basis by testing SNPs for the association of a disease. Highly ranked SNPs are considered to be highly associated with the disease. The major limitation of these linear methods is the need to analyze the association of each SNP with a disease in GWAs. Prediction errors due to multiple testing are high. Most of these studies also ignore interaction effects due to genetic and environmental factors.

In reality, it is widely believed that many complex diseases are influenced by various genetic, and environmental factors acting together causing interaction effects [2] [3]. As a step forward, genome-wide association interaction studies (GWAIS) have emerged to discover gene-gene interactions in disease association models by providing further insights into genetic architecture of complex human diseases [4]. These gene-gene interactions (epistasis) can be either biological or statistical [5]. Biological epistasis is a phenomenon of physical interactions between biomolecules such as DNA, RNA, proteins, and enzymes. Statistical epistasis occurs at population level due to inter-individual variation in DNA sequences. It is intuitively difficult to produce biological interpretations from statistical results due to the inherent nonlinearity. However, detecting these SNP interactions remains a biggest challenge in high-dimensional genome data due to the curse of dimensionality, biomolecular complexities, absence of

marginal effects, missing heritability, and computational limitations [2, 6]. These challenges have been partially addressed by a number of machine learning and data mining techniques [6, 7].

Current research studies have explored the use of varying, and modifying logic regression [8], penalized logistic regression [9], classification and regression tree (CART) [10], multivariate adaptive regression splines (MARS) [11], focused interaction testing framework [12] and automated detection of informative combined effects (DICE) [13]. The evolution of huge high-dimensional data in genomics has led to the application of data mining and machine learning approaches including data reduction and pattern recognition. These approaches discover interesting interactions by considering all genomic variables in vast search spaces. Data reduction approaches reduce high-dimensional data to low dimensional data. They include combinatorial partitioning method (CPM) [14], restricted partition method (RPM) [15], set association [16], and multifactor dimensionality reduction (MDR) [17]. Pattern recognition approaches extract patterns from the data using techniques such as cluster analysis [18], support vector machines (SVM) [19], self-organizing maps (SOM) [20] and neural networks (NNs) [19]. Variable selection, model building and model interpretation are the three primary challenges in these approaches. In addressing these challenges, new strategies and methods are developed to improve genome-wide interaction studies.

Tree based epistasis association mapping (TEAM) [21], Boolean operation based screening and testing (BOOST) [22], and GPU-based BOOST (GBOOST) [23] are some of the exhaustive search approaches. Even though these approaches are feasible, they are computationally intensive and execution time increases exponentially by the number of SNPs. In order to overcome these limitations, filtering approaches are used to analyse interesting SNPs. They include Random Forest (RF) [24], epiFOREST [25], SNPInterForest [26], Random Jungle (RJ) [27], forest based haplotype approach [28], Bayesian epistasis association mapping (BEAM) [29], fast epistatic interactions detection using markov blanket (FEPI-MB) [30], Bayesian network based epistatic association studies (bNEAT) [31], and Mega SNPHunter [32] and biological filters [2]. Though several aforementioned approaches are developed, SNPs with weak marginal effect may be filtered out that may significantly contribute to the disease. The contingency table had many empty cells which may lead to unstable estimation with large variance. Despite substantial advances in multi-

locus interaction studies, the progress in detecting these SNP interactions by the conventional approaches is still left with several challenges.

- **A SNP at a locus can be associated with more than one disease.**

A SNP at a locus can have stronger association with a disease and a weaker association for other diseases. In some cases, a SNP may indirectly associate with a disease by influencing the nearest SNP in a different locus. This challenge is worth considering due to the genetic complexity stated in the previous research study [33]. The research exposed SNPs in chromosome 9p21 are associated with diseases like artery diseases, diabetes, and multiple cancers. The SNPs in this region did not alter any sequences. They influenced three nearest genes to be associated with these diseases. They are strongly correlated with gene ANRIL and weakly correlated with CDKN2A, CDKN2B genes.

- **SNPs at different region in a genome can cause various impacts over a disease.**

A SNP in coding region (the region in a DNA/RNA sequence that codes for a protein) may alter the functionality of protein. A SNP in promoter region (the region in a DNA sequence that initiates transcription of a gene) may affect transcription (the processes in which DNA sequence is copied to RNA sequences) regulation. A SNP in intron region (the region in a DNA sequence which is removed by RNA splicing) may affect splicing and expression of a gene. Identification of such disease causing SNPs is proven to be more challenging [34].

- **Combinations of SNPs exponentially grow as number of SNPs increases.**

One million of SNPs would take approximately 3 years to analyse [35]. It is infeasible to find all the combinations of SNPs through exhaustive search. Power of current approaches decreases as dimensionality increases. Hence, efficient methods are required to handle high voluminous data being produced by genome sequencing.

- **The power of the method is reduced in the presence of noise.**

Some of the current approaches can handle noise reasonably well [36, 37]. However, they fail to address genetic heterogeneity and phenocopy, and their

combined effects [36]. The power of the current methods is drastically dropped by increasing the classification errors.

These challenges include the development of efficient methods for detecting a true causal subset of SNP interactions responsible for a disease.

## 1.1 Research Questions

The following research questions are raised while developing the efficient methods in this thesis:

- How efficiently methods can be developed to detect two-locus SNP interactions?
- Can the proposed methods be flexible to extend for higher-order interactions?
- How effectively the performance of the methods can be improved?
- Can the proposed methods be applied for unsupervised tasks?
- Can the proposed methods be implemented for imbalanced datasets?
- How effectively the power of the methods can be improved in the presence of noise due to genotyping error (GE), missing data (MS), phenocopy (PC), and genetic heterogeneity (GH)?

## 1.2 Research Objectives

The overall objective of this thesis is to develop methods that could enhance the understanding of the genetic architecture of complex diseases. The main objectives of this research are as follows:

### **Objective 1: Develop methods for search of two-locus SNP interactions**

- ❖ An associative classification based approach (AC)
- ❖ Incorporating associative classification into MDR (MDR-AC)
- ❖ Deep learning method (DNN)
- ❖ Deep hybrid method (DNN-RF)

### **Objective 2: Extend and improve the performance of the proposed methods**

- ❖ Investigate the proposed methods for higher-order SNP interactions
  - Three-locus to ten-locus interactions
  - Combined effect of higher-order interactions and main effects

- ❖ Investigate the methods for unsupervised tasks
  - High-dimensional data to low dimensional data
  - Discovering anomalies in the reduced representation of the data
- ❖ Improve the performance of the methods
  - Optimising hyper-parameters
  - Perform sensitivity analysis
  - Investigate and explore various optimising algorithms to improve the network learning

**Objective 3: Evaluating the proposed methods on both simulated and real data**

- ❖ Simulation studies
  - Balanced and imbalanced datasets
  - Absence of the noise and presence of the noise due to MS, GE, GH, PC, and their combined effects.
- ❖ Real data application studies
  - Sporadic breast cancer data
  - Hypertension data
  - Chronic kidney dialysis patients' data
  - Data obtained from whole genome study

**Objective 4: Comparing the performance of the proposed methods with the previous approaches**

- ❖ Validate the methods with previous methods on various simulated scenarios for the models exhibiting epistasis effects
- ❖ Experimental findings are confirmed on real datasets

## 1.3 Research Outcomes and Contributions

This thesis makes following contributions to the current multi-locus interactions studies in genetic epidemiology.

**Associative classification:** An association classification (AC) based approach is implemented to detect two-locus SNP interactions in Chapter 3. The goal of this contribution is to implement associative classification, and study its effectiveness for detecting the epistasis in balanced and imbalanced simulated datasets. The approach is

evaluated for single-locus models to six-locus models for various simulated scenarios. The method is evaluated for various simulated scenarios and real datasets. The simulated datasets are generated for five different penetrance functions by varying heritability, minor allele frequency, and sample size. The experimental results demonstrated significant improvements in the performance of the method on imbalanced datasets when compared with the previous methods. These research outcomes were published in [38-40].

**MDR based association classification:** Even though, the studies in Chapter 4 were encouraging for some of the balanced datasets, MDR performed significantly better than AC in all balanced datasets. Hence, the research proceeded by proposing MDR based AC (MDRAC) for revealing the unexplained hidden interactions behind complex diseases in Chapter 4. The proposed MDRAC is evaluated for same simulated scenarios, and real world datasets as presented in Chapter 3. The method is further studied in depth by adjusting threshold levels, and adding noise to the datasets. The experimental results demonstrated improved accuracy over previous method by reducing the classification errors. The proposed method identified some of the interesting interacting SNPs at various locations that are not exposed by traditional approaches. Further, the proposed method is successfully extended to higher-order SNP interactions (more than two-locus). The work presented in this chapter has been published in [41-43]

**Deep learning method:** In Chapter 5, it was observed that the power of MDRAC was much reduced in the presence of PC, and GH, and their combined effects with other sources of noise. Hence, deep learning strategies are explored in Chapter 5 for addressing this research problem. DNN is trained to predict two-locus polymorphisms due to interactions in genome-wide data. The performance of the method is studied on same simulated settings and on a published genome-wide dataset. The performance of the DNN is validated by varying the parameters of the models under various scenarios. The experimental evaluations over real data demonstrated significant improvements in the prediction accuracy over the previous machine learning approaches. This contribution has been published in [44, 45].

**Improving deep learning method:** Further studies are performed on DNN to maximise the prediction accuracy of best models by improving the method in Chapter 6.

The proposed method is extended by implementing dimensionality reduction using PCA, and studied the behaviour of the method for multi-locus SNP interactions (more than two-locus). The method is also studied for unsupervised feature learning tasks, and discovered anomalies using Autoencoder. The predictive performance of the models is improved by optimizing hyper-parameters, and improving network learning algorithms. Sensitivity analysis of input variables with respect to the response variables is also performed. Further, evaluations are performed to learn the behavior of the network in the presence of noise due to MS, GE, PC, and GH, and their combined effects. It is observed that, the power of the improved method is better than the previously proposed methods. However, still better approaches are investigated to improve the power of the extended DNN method in the presence of noise. These research outcomes were published in [46-48].

**Deep hybrid method:** In Chapter 7, a hybrid method is proposed by integrating DNN and RF (DNN-RF), to enrich the capabilities of DNN with the RF. The performance of the method is evaluated on number of simulated datasets, and findings are confirmed on a real world data application. On an average, the method outperformed for all the simulated models in the presence and absence of noise. The power of DNN-RF in the presence of PC, and GH, and their combined effects with other sources of noise are much higher than the previous methods. This is due to the incorporation of RF into the method, which improved overall predictive accuracy of the models. The work presented in this chapter is accepted [49].

## 1.4 Thesis Structure

This thesis is written based on peer reviewed publications. Hence, the contents of the chapters will be partially overlapped with the published work. The outline of this thesis has been organised as follows.

**Chapter 2:** “*Literature Survey*” reviews the relevant background related to this thesis. The chapter provides an overview of current approaches for identifying interactions that contributes complex diseases. It includes establishing the terminologies used in this thesis. It also investigates the challenges to be considered while developing the new methods. The chapter further discusses the data simulation tools, and sources of real

data to evaluate the performance of the methods. The review presented in this chapter is published in [50].

**Chapter 3:** “*Associative classification for detecting higher-order SNP Interactions*” provides an associative classification based approach to detect multi-locus interactions.

**Chapter 4:** “*Multifactor Dimensionality Reduction based Associative classification*” describes the integration of multifactor dimensionality reduction into associative classification to address the problems that arise in rule based approach.

**Chapter 5:** “*Towards Deep Learning Models for SNP Interaction studies*” explores deep learning techniques to improve the performance of the methods in the absence and the presence of noise. The models are trained and validated for various simulated datasets and real datasets

**Chapter 6:** “*Learning Method for an intensive search of higher-order interactions*” investigates optimising techniques and applied on deep learning models to improve learning. The chapter also provides the behaviour of networks over higher-order interactions and their combined effects.

**Chapter 7:** “*A Deep Hybrid Method for the noise data*” presents the details of a hybrid approach, which incorporates random forest into deep learning method. It further discusses and analyses the performance of the models in the presence of noise.

**Chapter 8:** “*Conclusions and Future works*” provides a summary of the contributions presented in this thesis. It further includes the potential limitations of this research work. The chapter concludes by suggesting the potential areas of future work. Suggestions towards future works may partially presented in the published papers [49, 51].

# Chapter 2

## Literature Survey

This chapter will provide some basic understanding of SNP interactions, and their association with complex diseases in Section 2.1 and Section 2.2. Section 2.3 addresses the issues to be considered for designing the methodologies in the current literature. Section 2.4 and Section 2.5 reviews the current methods and the related software packages to detect the SNP interactions. The strengths and limitations of current methodologies are also discussed in this section to highlight the gaps to be considered while designing the new methodology. The chapter further reviews the achievements in data simulation to evaluate the performance of current methodologies in Section 2.6. Finally, Section 2.7 discusses the alternative approaches of SNP interaction analysis.

---

This chapter is based on the previously published manuscript:

- S. Uppu, A. Krishna, and R. Gopalan, "A review on methods for detecting SNP interactions in high-dimensional genomic data," *IEEE/ACM Transactions on computational Biology and Bioinformatics*, vol.15, pages 599-612, 2018: © 2018 IEEE, "The original publication is available at <https://ieeexplore.ieee.org/document/7765022>".

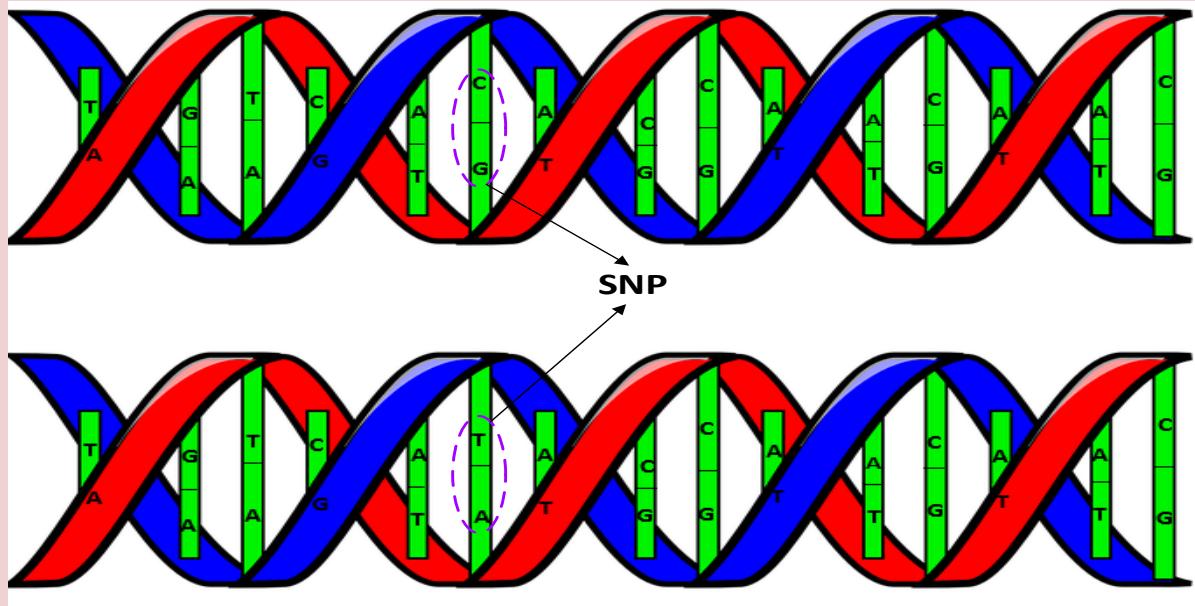
## 2.1 Single Nucleotide Polymorphism

The genetic epidemiology of human disease is complex, where no single factor on its own has been the cause of a disease. Rather, they are influenced by the combination of genetic, environmental, and life style factors. Some examples include breast cancer, diabetes, cystic fibrosis, multiple sclerosis, hypertension, obesity, asthma, anemia, parkinson's and alzheimer's disease. About 99% of the human genome is identical, with only 1% variations in deoxyribonucleic acid (DNA) [52]. Understanding the role of these variants related to a disease mechanism will improve the effectiveness of treatments. There have been an increasing number of studies that depict genotype-phenotype relationships by identifying genetic variants associated with a disease. The modern unit of genetic variation is the single nucleotide polymorphism (SNP, commonly pronounced as "snip") [53]. A polymorphism is a region of the genome that varies between different individuals [54]. A SNP is a common genetic variation caused by the change of a single nucleotide (A, T, C and G) in the DNA sequence. SNPs are the most abundant form of genetic variations, and they are not evenly distributed across the human genome [53]. About 93% of all genes contain at least one SNP [54]. On average, one SNP occurs in every 300 nucleotides, such that there are around 12 million SNPs in the human genome [55]. The frequency of SNPs is at least 1% of the population. SNPs may occur in the coding sequence of genes, noncoding regions of genes, or regions between genes. The majority of SNPs (used as markers of a genomic region) have minimal impact on biological systems [53]. However, the functional consequences of SNPs causes changes to amino acid, mRNA transcript stability, and transcription factor binding affinity [53, 56]. In rare genetic disorders, such as cystic fibrosis [57], disease manifestation is due to extremely rare genetic variants that initiate a change in protein function [53]. These low frequency rare genetic variants in the population are sometimes referred as mutation (structurally similar to a SNP)[53]. The frequency of rare variants is less than 1% in a population.

### Box 2.1: Single Nucleotide Polymorphism (SNP)

**DNA is a polymer composed of nucleotides, which are bonded asymmetrically [58]. Each nucleotide consists of a single nucleobase from four possible bases of adenine (A), thymine (T), cytosine (C) and guanine (G). Hydrogen bonds of A with T and C with G forms a double helical structure with two DNA molecules. DNA consists of genes and encodes information for**

**synthesizing proteins and ribonucleic acid (RNA).** Even though, DNA is a stable molecule, about 1% genetic variations occur between individuals. The locations that differ from other individual are known as SNPs. That is, a base pair in a sequence is replaced by another base pair is a SNP. They are the most common variants in humangenome, whose frequency is greater than 1% in a population.



**Figure.2.1 Single nucleotide polymorphism in DNA sequence (adopted from [59]).**

## 2.2 Interactions

Recently, genome-wide association studies (GWAS) have been substantially expanding the knowledge on the role of SNPs and their associations in revealing the genetic epidemiology of a disease. GWAS identifies the SNPs that are candidate biomarkers for genes that could indicate complex diseases in an individual. Despite the success of GWAS in identifying disease causing SNPs, the progress in identifying interactions between SNPs and their associations with the disease is still limited. The biology behind complex diseases is characterized by multiple factors acting together and/or independently [2]. This “underground network” of multiple factors may eventually induce disease manifestations [60]. Some factors may cause the disease by themselves while some others may cause the disease in the presence of other factors. When two or more factors depend on each other to develop a disease, it is often due to interaction

effects [2]. The Webster's dictionary meaning of interactions has two meanings: "*intermediate action and action on each other; reciprocal action or effect*" [61]. There is no universally accepted meaning for interactions by biologists and statisticians [61]. The author broadly defined the term interactions as the factors or objects that do not act independently. However, in the literature, the definition of interaction falls mostly under two main categories [2]. The first category is antagonistic interaction, where the expected measure of risk factors is lower compared to the sum of individual risk factors. In contrast, synergistic interaction occurs when the expected measure of risk factors exceeds the sum of individual risk factors. As part of the complexity of diseases, some of many phenomena are due to genetic heterogeneity, and non-linear interactions between genetic variants and environmental exposure rather than individual factors [6, 62]. Exposing these interactions such as, gene-environment interactions and / or gene-gene interactions play an important role in better understanding the development of complex diseases. Hence, the study of such interactions has been the subject in genome-wide interaction studies (GWAIS) as a step forward from GWAS.

### **2.2.1 Gene-Environment Interactions**

Garrod first studied the consequence of gene-environment ( $G \times E$ ) interactions by suggesting that the various effects of genes can be modified by environmental exposures [63]. Since then, there have been a number of studies aimed at describing the joint effects of genetic and environmental factors that influence a disease. Hunter and Thomas reviewed emerging studies associated with  $G \times E$  interactions in human diseases [64, 65]. Widely in epidemiological studies,  $G \times E$  interactions are statistically defined by estimating the degree of risk attributable to the combined effect of a genetic and an environmental factor on a disease, which exceeds or decreases when compared to the expected joint effect (additive) [65-67]. Alternatively in the context of intervention studies,  $G \times E$  interactions are described as the risk conveyed due to specific genotypes depending on the level of one or more environmental exposures (such as diet and physical activity) [66].

### **2.2.2 Gene-Gene Interactions**

Gene-gene ( $G \times G$ ) interaction also known as epistasis or genetic interaction has been one of the topics for many researchers to address several enduring questions of

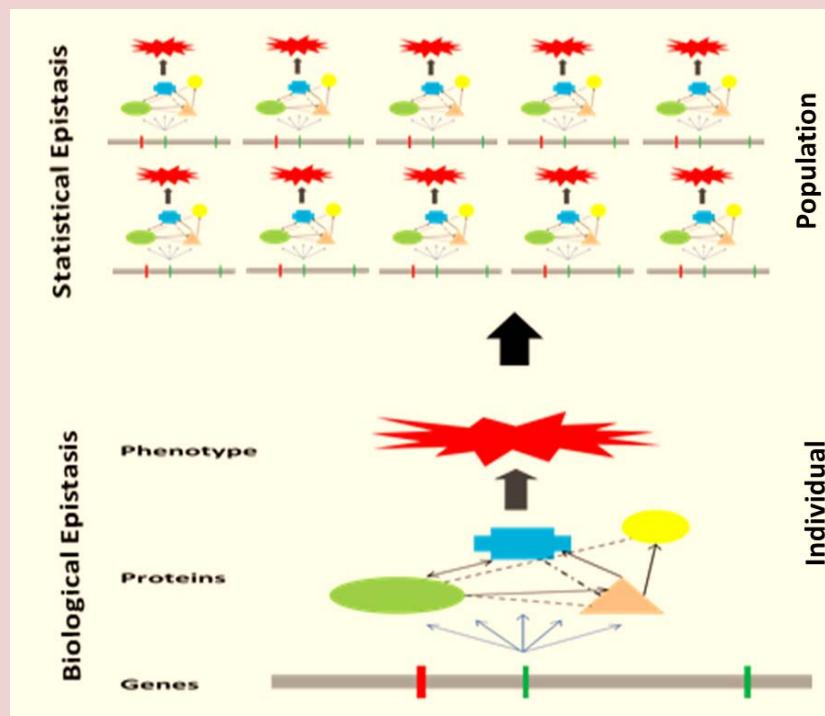
underlying genetic architecture of complex diseases [3, 6, 35]. Epistasis has diverse definitions, given by biologists, statisticians, epidemiologists, and geneticists. Bateson first coined the term “epistasis” (a Greek word which means “standing upon”) to explain deviations from Mendelian inheritance [68]. The author described epistasis as an allele at a locus masks the expression of an allele at another locus. Subsequent descriptions of epistasis have given rise to various definitions in the literature. Fisher [69] defined epistasis as a deviation from additivity in a linear model. Cockerham [70] divided epistasis into three types: additive \* additive, additive \* dominant and dominant \* dominant [71]. Further, the language of epistasis is discussed by Phillips [72] who extended epistasis into six types: dominant epistasis, recessive epistasis, duplicate genes with cumulative effect, duplicate dominant genes, duplicate recessive genes, and dominant & recessive interaction [71]. In genetic epidemiology, frequently epistasis can be defined as the interactions between alleles at different loci [61]. The locus not only represents location of a gene, it also represents the location of genetic variants nearby or within a gene. One gene or any genetic factor masks or suppresses the effect or action of other (s) is one of the other alternative definitions of epistasis provided by the author.

However, according to the recent reviews, epistasis can be viewed in terms of biological and statistical effects [3, 35, 73]. Bateson’s biological epistasis is a contrast to Fisher’s statistical epistasis [73]. Moore defined biological epistasis as physical interactions between biomolecules (such as DNA, RNA, proteins and enzymes) within gene regulatory networks and biochemical pathways [5, 74]. In biological epistasis, the effect of a gene on a phenotype is dependent on one or more other genes. Statistical epistasis can be defined as the deviation from additivity in a generalized model [5]. This non-additivity of genetic effects is measured mathematically from population level data. The distinction between biological epistasis and statistical epistasis is discussed in detail by Moore [5]. Even though, the author considers the significance of this distinction for drawing a biological conclusion from a statistical model that describes a genetic association [75], the definition of epistasis still causes confusion when interpreting statistical findings. Hence, a clear meaning of epistasis is crucial to clarify the underlying genetic networks of biological systems. However, the classic definition of epistasis

provided by Bateson and Fisher is still considered to be a good starting point of explaining gene-gene interactions [76].

### Box2.2: Epistasis

The interaction between two or more genes to affect a phenotype, such as disease susceptibility, is called epistasis [5]. An allele at a locus masks the expression of an allele at another locus. Biologically, epistasis likely arises from physical interactions occurring at individual molecular level [5]. Statistically, epistasis refers to the average effect of substitution of alleles at combinations of loci, with respect to the average genetic background of the population [6].



**Figure 2.2** Relationship between Biological epistasis and Statistical epistasis (adopted from [5]). Biological epistasis occurs at an individual level which involves variations in DNA sequence (vertical bars), biomolecules (circle, cross, oval, and triangle), and their physical interactions (dashed lines) giving rise to a phenotype (star). Statistical epistasis occurs at a population level due to inter-individual variability in genotypes, biomolecules, and their physical interactions.

**Table 2.1:** Extended types of epistasis as reported by Phillips [72]

Gene interaction	Inheritance pattern	A-/B-	A-/bb	aa/B-	aa/bb	Ratio
Additive	Each genotype results in a unique phenotype.	9	3	3	1	9:3:3:1
Dominant	The dominant allele of one gene	12		3	1	12:3:1

epistasis	masks the expression of either allele of the second gene.			
Recessive epistasis (Supplementary interaction)	The recessive allele of one gene masks the expression of either allele of the second gene.	9	3	4
Duplicate recessive genes (Complementary genes)	At least one dominant allele from each of two genes needed for phenotype.	9	7	
Duplicate dominant genes	One dominant allele from either of two genes needed for phenotype. The duplicate genes are also called as pseudoalleles.	15		1
Duplicate genes with cumulative effect	A new phenotype is expressed, on the presence of both the dominant non allelic alleles of two genes. However, they express phenotype separately, when they are allowed to express independently. In the absence of any dominant allele, the recessive allele is expressed.	9	6	1
Dominant recessive interaction	The dominant allele either in homozygous or heterozygous condition of one gene, and the homozygous recessive allele of other gene produce the same phenotype.	13	3	13:3

According to recent reviews, identifying these gene-gene interactions have become more pervasive, extremely important to consider and challenging to detect [2]. The current methods for identifying gene-gene/SNP-SNP interactions are mathematically challenging and computationally complex. The challenges include the high-dimensionality problem, computational limitations, absence of marginal effects, missing heritability, and genetic heterogeneity [6, 19, 24, 26, 35, 73, 77, 78]. These challenges

have been addressed by a number of data mining and machine learning approaches. The current methods and the related software packages that are being used to detect SNP interactions are reviewed in this chapter. The issues that need to be considered when developing these methods are also addressed. In addition, the chapter reviews the achievements in data simulation for evaluating the performance of these approaches to identify the SNP interactions. Further, the future of SNP interactions and the methods to be considered are also discussed.

## **2.3 Issues to be considered for designing the models**

A number of strategies are developed for identifying SNP interactions to reveal the underlying genetic architecture of complex diseases [79]. Understanding the role of SNPs and their interaction effects in the context of disease susceptibility will improve diagnosis, prevention and treatment [77]. The major problems associated with uncovering these interactions are variable selection, model building and model interpretation [2].

### **2.3.1 Variable Selection**

The challenge of variable (SNP) selections poses a needle-in-a-haystack problem. The variables are selected from high-dimensional datasets to understand the underlying phenomena of interest. There are about 12 million SNPs along the 3-billion-base human genome [80]. Evaluating combinations of SNPs from these candidates and identifying an optimal combination among an astronomical number of possible combinations is computationally infeasible [77]. For example, there will be  $4.5 \times 10^{10}$  two-way interactions and  $4.5 \times 10^{15}$  three-way interactions to be examined in a study of 300,000 SNPs in GWA [2]. These computational challenges have been addressed in the current research either by using feature selection or feature extraction approaches.

Feature extraction reduces the dimensionality of the variables by aggregation and projection [81]. Feature selection extracts an optimal subset of features to avoid over fitting, improve model performance, enhance data analysis and reduce execution time [35]. In contrast to the feature extraction methods, feature selection methods do not

reduce the number of variables. They select a subset of all the existing features. Hence, the number of tests for searching the combinations of SNPs is reduced. These methods reduce the burden of multiple testing by substantially decreasing the chances of rejecting the null hypothesis when it is true. The Feature selection methods are classified as filter, wrapper and embedded methods [35]. Filter methods select a subset of variables for analysis as a pre-processing step, by algorithmically assessing the quality of the variables independently. Wrapper methods use deterministic or stochastic approaches and iteratively select a subset of variables in accordance with their predictive power. Embedded methods select variables during the training step of the algorithm. They are more specific to the learning machine. These methods include interactions with classification models and they are computationally more efficient than filter and wrapper methods. The detailed reviews of feature selection are given by Guyon [82], and Saeys [83].

**Table 2.2:** Variable selection methods in detecting SNP interactions [35]

Methods	Approaches	Description	Reference
Filter	Entropy based	These methods detect SNP-SNP interaction (model free option ESNP2-S) and the best model is identified from various SNP interaction models (model based ESNP2-Mx).	[84]
	Synergy based	It is a system based approach which identifies synergistically interacting SNPs by introducing information theory of multivariate synergy.	[85]
	ReliefF	It chooses the instances randomly based on nearest neighbours from the entire vector values of all attributes by changing the weights.	[86]
	TuRF	ReliefF is systematically improved by filtering worst attributes and re-estimating the weights of remaining attributes.	[87]
	SURF	Spatially uniform ReliefF selects all neighbours which are in predefined epsilon to improve the sensitivity of the approach to identify epistasis.	[88]
Evaporative cooling			
Attribute quality measure is used on thermodynamic free energy which integrates ReliefF and entropy to obtain the subset of features that contains SNPs associated with phenotype by removing the noisy			

---

		features.
<b>Wrapper</b>	GPAS	Genetic programming for association studies uses [90] genetic programming and multi-valued logic in disjunctive normal form (DNF) to enable related categorical response variables.
	Ant-colonization optimization	It is a positive feedback approach that modelled on [91] the behaviours of ants (to find the shortest paths) and is easily parallelizable.
	AntEpiseeker	It is a two stage modified generic optimization of ant colony algorithm. The first stage searches SNP sets using $\chi^2$ scores and pheromone levels. The second stage searches exhaustively for the SNP interactions within the subset set of highly suspected SNPs. [92]
<b>Embedded</b>	Decision tree based	They are used for modelling the relationship between [93] SNPs and a discrete end point by sorting through a tree.
	Recursive partitioning	A population is segmented into non-overlapping [94] groups in which the response of interest is more homogeneous within the segments. It greatly reduces effective search and inference space.
	Random forests	They consist of regression trees generated from [95] random vector using a bootstrap sample of instances from the data. Each tree is grown without pruning.
	Logic regression	Logic regression adapts Boolean combinations of [96] logical variables as predictors for the response in GWAS. LogicFS uses bagging in logic regression to detect SNP interactions associated with a binary response.

---

### 2.3.2 Model Building

The second challenge that needs to be addressed is the developing of powerful statistical and computational methods for uncovering the SNP interactions. Characterizing the correlation between interacting SNPs, environmental exposures, and other genetic variants with disease susceptibility is more challenging while designing the models. This is due to the increase in dimensionality associated with multi-locus

genotype combinations when compared to assessing SNPs individually. That is, a SNP has two possible alleles, one common allele (A) and the variant or minority allele (a). Three possible genotypes (AA,Aa/aA, and aa) are yielded due to duplication of DNA in each cell. Hence, the interaction between two SNPs will yield  $3^2 = 9$  two-locus genotype cells. There will be 27 three-locus genotype cells for third order interactions and so forth [78]. This dimensionality problem (curse of dimensionality) [97] has been analytically challenging to detect SNP interactions in sparse data. The traditional parametric statistical approaches (such as logistic regression) do not deal effectively with curse of dimensionality [77]. Hence, several data mining and machine learning approaches were developed and they are more powerful than linear approaches [6, 7, 19, 35, 71, 73, 77]. However, these approaches are susceptible to learn rare patterns in datasets that results in false positives. The false positives can be reduced by implementing permutation testing [98] and cross validation [99]. Further, these non-parametric approaches do not have underlying theoretical distributions that can lead to over-fitting the data. These issues are to be addressed while designing models that can identify the interactions effectively in a large search space.

### 2.3.3 Model Interpretation

The interpretation of SNP interaction models is still a challenging issue to be addressed. Once the multi-locus SNP interaction model is detected for disease susceptibility, a biological interpretation is to be developed for that model. For example, in [2], a four-way SNP interaction model is identified in a particular biochemical pathway. Increase or decrease of disease susceptibility by the particular genotype combination is more challenging than identifying an interaction model. There will be 81 possible genotype combinations for four SNPs. At least 81 experiments are required to evaluate the effects of SNPs on the enzyme activity in the biochemical pathway. Hence, the biological interpretation of four-locus model creates 81 transgenic mouse lines for each genotype at a single-locus. A number of software packages are developed to assess biological plausibility. GenePattern [100] integrates a set of analysis tools and knowledge sources to facilitate biological interpretation. Exploratory visual analysis is designed to integrate results with biological knowledge from public databases and have been applied to GWA data [101]. Some of the other packages for interpreting results include Pathway Studio from Ariadne and Ingenuity Pathway Analysis from Ingenuity Systems [77]. Ontology

based methods [102] and literature based systems [103] have also been proposed for aiding the biological interpretation [77]. The biological interpretations inferred from statistical interaction models are still limited. Some recent studies focus on strong prior biological models that have been extensively studied in the current literature [104, 105] to assess the fitness of the models from the out-breeding population.

## 2.4 Methods for detecting SNP interactions

The conventional statistical approaches [6, 77] focus on a single-locus analysis where Mendelian laws hold. A series of statistical tests are performed on genome-wide association data by examining each SNP individually for the association to a phenotype. These statistical tests (depend on various factors) are different for quantitative traits and case-control studies [106] [107]. Generally, quantitative traits are analyzed by generalized linear model (GLM) approaches, and analysis of variance (ANOVA). On the other hand, case-control traits are generally analyzed by using contingency table tests, and logistic regression. Contingency table tests (such as, Chi-squared test, and Fisher's exact test) examine the SNPs that are associated with a phenotype. Highly ranked SNPs are considered to be highly related to the disease. Logistic regression (the outcome of linear model is transformed using logistic function) is the most widely used approach that allows adjustment for clinical covariates, and can provide adjusted odds ratios. In addition to single-locus analysis, a number of recent reviews suggested that exploring interaction effects will also play an important role in revealing the full extent of genetic architecture [3, 6]. Consequently, this led to an intensive research of discovering multi-locus SNP interactions in GWAS. Multi-locus analysis is not as simple as single-locus analysis due to numerous statistical, logistical, and computational complexities. By addressing current challenges, efficient statistical and computational techniques have been consistently explored. The following subsections briefly review some of the current data mining and machine learning methodologies to detect SNP interactions.

### 2.4.1 Multifactor Dimensionality Reduction

Multifactor Dimensionality Reduction (MDR) [17, 36, 108] is a model free, nonparametric, exhaustive method to identify gene-gene interactions in higher-order data. It reduces high-dimensional genotypic combination data to a single dimension

using the constructive induction approach. The model selects  $n$  set of factors (SNPs) from all possible combinations of  $n$  loci. SNPs are bi-allelic and can be genotyped in three possible combinations. The combinations for selected  $n$  SNPs are represented by cells in  $n$ -dimensional space as contingency tables. The case to control ratio for each cell is calculated and is classified as either a high risk or a low risk cell based on a threshold value. The model is evaluated using Naive Bayes classifier and the balanced accuracy is reported. Finally, an optimal predictive model is obtained by performing cross-validation. The statistical significance of the final model is achieved by performing permutation testing using 1000 permutations. The null hypothesis of no associations was rejected when the p-value from permutation test is  $\leq 0.05$ . The MDR tool [109] implemented in java is publicly available for download from <http://www.epistasis.org/>.

Even though MDR could successfully identify SNP interactions by reducing the false positive rate, it still suffers from major drawbacks. The model does not scale up to analyze a large number of SNPs. It may eliminate useful interactions due to reduction of high-dimensional genotype combinations to one dimension. It is unable to assess the proportion of risk level and is restricted to balanced datasets. Further, its efficiency is reduced in the presence of genetic heterogeneity and phenocopy. Hence, several aforementioned strategies are extended by addressing the limitations of MDR. Velez [108] extended MDR for imbalanced datasets. Lee [110] applied regression to MDR to deal with non-binary and continuous variables. Pattin [111] used interaction networks to reduce the computational time. Two-way ANOVA statistical tests are performed to identify significant interactions supported by text mining and experimental evidences. Namkung [112] extended MDR to address missing data. The author evaluated the performance of MDR using weighted balanced accuracy and ordinal association measures. This improved the power of MDR to detect interactions. Odds Ratio based MDR (OR-MDR) [113] uses odds ratio as a quantitative measure to identify the interacting SNPs. The odds ratios of each combination of genotype are ordered from highest to lowest. Further, a confidence interval is calculated for each combination of genotypes using an empirical distribution (bootstrap or sampling theory) to identify the significant odds ratio. The significance of the model can still be determined by permutation tests based on cross validation consistency.

Generalized MDR (GMDR) [114] is a generalized unified linear model based on the scores that allows adjustments for discrete and quantitative covariates. The framework can handle both quantitative and dichotomous phenotypes. Model based MDR (MB-MDR) [37, 78, 115] is a flexible framework that combines multi-locus genotype cells using disease status. The model introduces the concept of no evidence cells (0) to increase the power of MDR. Robust MDR (RMDR) [116] uses Fisher's exact test instead of a predetermined threshold for objective statistical criteria. It solely considers statistically significant genotype combinations obtained from MDR to make the final model more robust. Some of these approaches are successfully applied over genetic data [17, 113-116]. The statistical significance of MDR based models are generally evaluated using 1000-fold permutation testing. The permutation based testing (for accurate multiple testing corrections) is computationally infeasible for high-dimensional genome data [35]. It is also a time consuming strategy as tests have to be performed for all permutations to find the threshold. Non-fixed and omnibus alternative permutation testing strategies are explored by Motsinger [117]. The author demonstrated improved performance of MDR by using omnibus permutation testing with reduced false positives. In further studies, the author recommended reduction in cross validation from 10-fold to 5-fold that will reduce the runtime of the algorithm without losing the performance [118]. However, MDR performed poorly in the presence of genetic heterogeneity for both 10-fold and 5-fold (even worse) CVs. MaxT (permutation based gamma distributions) is an efficient technique implemented to control the false positive rate by improving the memory efficiency and speed [119]. Detailed study of MDR and extended methods are reviewed by Gola [120].

## 2.4.2 Random Forest

Random forest (RF) [121, 122] is an ensemble classification and regression trees (CARTs) that are grown in a random subspace of data. Each tree in the forest is grown using a bootstrap sample of instances from the data. One third of samples are left in out-of-bag (OOB) to estimate the prediction error. The attributes in the tree are chosen using random feature selection. A binary split is represented as a node, which terminates at a leaf. The data is recursively split into distinct subsets using appropriate splitting rules. Each subset of child nodes is purer than the corresponding parent node. Finally, the subjects are classified by majority votes over all trees in the forest. RF

outperforms the prediction when the trees do not exhibit a correlation with each other. It can estimate the importance (Gini) of each attribute in the training data. It can also determine the pairwise proximity among the samples. These features of RF uncover the interactions between genes in the absence of main effect [123]. The algorithm is implemented in a number of open source software packages, such as R [124], Rapid miner [125], Weka [126] and Willows packages [127]. RF can be very suitable for handling large p-value problems. However, RF contradictorily requires a marginal effect in at least one of the SNP interacting pair. This confound reduces the power of detecting interacting SNPs.

These limitations are addressed by several extensions of RF. EpiForest [25] uses RF analysis to obtain the Gini importance of individual SNPs. It merges with Sliding window sequential forward feature selection (SWSFS) algorithm to obtain a subset of SNPs by reducing the classification errors. The statistical significance of two-locus and three-locus models is identified using a hierachal procedure based on B statistic. Random Jungle (RJ) [27] is a computationally efficient publicly available software package specially designed to analyze the high-dimensional genome data. RJ uses variable backward elimination to measure the features. It executes in parallel by using the multithreading and message passing interface (MPI) parallelization [27]. Important SNPs responsible for interactions are identified by permutation based importance measure. RFCouple [128] constructs a RF based pre-screening method by aggregating information of multiple SNPs. Further, it is coupled with the MDR approach to improve the performance. Permutation based measures of MDR are used to determine statistically significant interactions. SNPIInterForest [26] reconstructs RF by selecting multiple-SNPs at a node and bounds a procedure for extracting SNP interactions. Although the approach is computationally demanding, it outperforms BOOST [22] and SNPHarvester [129], by reducing the FDR. Stratified sampling RF (SRF) [130] constructs RF using the stratified sampling method to select the feature subspace. SRF uses the equal width discretization algorithm in which, width of in-formativeness of each set of variables is equal. TRM [131] is a hybrid model that merges RF with multivariate adaptive regression splines (MARS). RF identifies the subset of useful SNPs, followed by detecting interacting SNPs using MARS. Mutual information network guided random forest (MINGRF) integrates mutual information network (MIN) with RF to reduce bias

on the marginal main effect and to avoid random sampling of variables [132]. The method demonstrated reduced false positive rates compared to MARS. Most of these RF based methods still suffer from multiple testing problems. High performance statistical tests with adjustments for multiple testing are desirable. RF and its extensions are successfully implemented over real genetic data [26, 27, 130-133].

### 2.4.3 Neural Networks

Neural Networks (NNs) [121] are popular machine learning models inspired by the brain's ability to solve problems. The basic units of NNs are neurons. NNs represent a directed graph in which nodes denote neurons (computational processing unit), arcs represent connection between neurons, and directionality of arcs represents the direction of information flow [73, 134]. These nodes are portrayed by layers with an output node. The input layer with one or more nodes are linked to nodes that are in hidden layers through arcs. Weights are assigned to the arcs to minimize the prediction errors. NNs are trained using known SNPs as inputs and disease traits as output. Linkage (identifying linkage between disease locus and a marker) and association (identifying linkage disequilibrium of disease locus and a marker) analysis are the two different methods used to discover the associated SNPs related to a disease [7].

Tomita [135] implemented artificial neural networks along with the parametric decreasing method to predict the SNP interactions associated with childhood allergic asthma. Evaluation of significant interactions is determined by an effective combination value (ECV). Keedwell [136] combined a Genetic Algorithm (GA) with NN to identify the interactions in temporal gene expression datasets. It finds the gene network that fits the gene expression data in reasonable execution time. Back propagation NN (BPNN) is traditional NN to detect the interactions between non-functional SNPs. Genetic programming NN (GPNN) [137] is an optimized NN using genetic algorithm to identify the associations in human disease. The algorithm optimizes input variables, nodes, weights, arcs and hidden layers. Power and predictive ability are improved against BPNN for both functional and non-functional SNPs on two-locus models. It is successfully implemented on real genetic data [134, 138].

However, the factors that hinder GPNN are two connections per node that are apparently more simplified for the complex data. Any changes in the program will lead

to an unstable network and will increase the execution time. Moreover, it may not cope well with complex data due to a restricted number of layers. The method did not perform well on three-locus models. It had more false positives with low power. In addressing these concerns, Grammatical Evolution NN (GENN) [134, 138] was developed. GENN is an evolutionary strategy that uses the linear genome of individuals sheared into codons. These codons are translated into phenotype using a grammar. The computational scalability of the method is demonstrated on 500,000 SNPs. Even though it outperforms the previous methods, it could only identify strong pair-wise interactions. Further, the interaction effect is influenced by genetic heterogeneity, polygenic inheritance, high phenocopy rates and incomplete penetrance [19]. Hardison [139] combined GENN with quantitative traits (QGENN) and the approach was evaluated using simulated genetic models. It investigated the quantitative trait associations and demonstrated high performance than traditional statistical approaches. The final model obtained by this method has all causative loci with high CVC. Although, the grammar used in the model provides high generality, the power of the model is reduced by increased computational time. The analysis tool for the heritable and environmental network association (ATHENA) [140] uses alternative tree based crossover backpropagation for locally fitting NNs weights. It initializes the search for SNP interactions by incorporating the domain knowledge obtained from publicly available biological databases.

#### 2.4.4 Support Vector Machines

Support Vector Machines (SVMs) [121] are supervised pattern learning methods that are as powerful as NNs in identifying SNP interactions. They are non-probabilistic binary linear classifiers as they identify a linear hyper-plane to separate data points of two classes [19]. SVM can be used to detect interacting SNPs by learning from the features [7]. The training data of SVM consists of two sets of feature vectors that are labelled as positive and negative. Positive vectors indicate the presence of interactions between SNPs and negative vectors indicates a lack of interactions between SNPs. Each feature vector characterizes a pair of SNPs. These features are mapped into a high-dimensional space and are separated into genetically interacting pairs and non-interacting pairs using a hyper-plane [7]. SVMs are successfully implemented in both simulated and real data to detect SNP interactions [141-143]. They provide

interpretable results and reduce the instability in response variables. However, SVM approaches face a number of major challenges. They produce unacceptable type I errors. These models are applicable only for a small number of SNPs. They do not cope well with missing data and genetic heterogeneity. The pioneering works are reviewed in detail by Koo and some of the methods are reviewed below [7].

Chen [142] proposed a framework by combining SVM with combinatorial optimization techniques to form four models: They are: Recursive Feature Elimination (SVM-RFE), Recursive Feature Addition (SVM-RFA), local search (SVM-local), and Genetic Algorithm (SVM-GA). Even though, the computational cost of the method is high, the results demonstrated a strong capacity for identifying the SNP interactions with less concern for over-fitting. The approach handles unbalanced input data sets effectively by introducing penalized SVM formulation. Ozgur [144] proposed an intelligent method by combining automatic text mining and network analysis to extract disease related SNPs. An interaction network is modelled from known disease related SNPs by using text mining based on dependency parsing and SVMs. The SNPs in the network are ranked by using the degree, eigenvector, and closeness centrality metrics. The genes identified by degree and eigenvector methods are significantly better than by the baseline method (Fisher's exact method). The approach is robust to random errors by randomly removing the edges of the constructed gen-interaction network. Shen [145] proposed a two-stage approach to detect disease associated SNP interactions. In the first stage candidates are selected using a model selection method (SVM with L1 penalty). In the second stage, p-values are calculated for the candidates identified from the first stage using the ad hoc logic regression method [145]. Non-significant candidates are removed using Bonferroni correction to reduce type I error. The approach was analyzed on Parkinson disease data with 540 cases and controls individually with 408,000 SNPs. Single-locus logic regression was used to compute p values for each SNP. SNPs whose p values exceeded 0.001 were excluded. A subset of 401 SNPs was obtained and was further analyzed using the proposed approach. It identified the interaction between two SNPs (rs2351881, rs7924316), whose p-value was less than 0.05. Fang [146] proposed an extended SVM and SVM based pedigree based GMDR (PGMDR) to identify the interactions in the presence and the absence of main effects by adjusting covariates. The approach demonstrated high prediction accuracy, consistency, and an ability to detect

SNP interactions by controlling type 1 error rates. Grammatical SVM (GESVM) [147] combines SVM with GE to select features and parameters. Zhang [148] coupled SVM with Binary matrix shuffling filter (BMSF) to overcome the limitations of traditional wrapper methods and over-fitting. It automatically conducts multiple rounds of filtering for the potential interactions during gene selection and provides highly accurate classification of the samples.

## 2.4.5 Regression Models

Regression models are a well-established statistical data analysis technique that determines the relationships between input variables and an output variable. In linear regression, a quantitative outcome  $y$  is a function of predictor variable  $x$  [6]. It is represented using regression equation  $y = mx + c$ , where,  $m$  is the regression coefficient or slope of the best fit line and  $c$  corresponds to the intercept [6]. Multiple regression models of three variables are represented as  $y = m_1x_1 + m_2x_2 + m_3x_3 + c$ . It is assumed that there is a linear relationship between each of the input variables ( $x_1$ ,  $x_2$  and  $x_3$ ) with the outcome variable  $y$  [6]. Logistic regression (LR) [6, 24] tests for statistical interactions using log odds  $\ln(p/(1-p))$ , where,  $p$  is the probability of disease risk. In particular, these regression models use Boolean combinations to improve the prediction ability of the models. LR is successfully applied over SNP data to analyze the SNPs associated with the phenotype [96, 149-152]. However, the major challenges of these methods are model specific, over fitting, and computationally intensity (that is,  $n$  way interactions require  $2^n$  regression equations). Generally, parametric regression models perform hypothesis based statistical tests by performing multiple comparisons that can lead to incorrectly rejecting the null hypothesis. This multiple testing problem is usually corrected with Bonferroni correction by adjusting p-values to control family wise error rate (FWER) (the probability of at least one type 1 error). In a study that performs 1000 hypothesis tests with a conventional 5% statistical significance, the Bonferroni correction for each test will be set to  $0.05/1000$ . Benjamini-Hochberg is a powerful procedure to correct false discovery rate (FDR) (proportion of false positives among the set of rejected hypothesis). These corrections can lead to identifying SNPs with strong interaction effects by ignoring many weaker interaction associations with the phenotype. Hence, adjusting for multiple testing may reduce the power of these methods to detect weaker interacting associations.

Several strategies have been proposed to increase the capability of LR. Park [9] combines a penalization of the  $L_2$ -norm of the coefficients with LR. It reduced the problem of empty cells in the contingency table. The approach does not reduce the stability of the fits and does not degrade the collinearity among the variables. PLINK [153] is an open source whole-genome association analysis toolset providing a logistic regression test for interaction that assumes an allelic model for the main effects and interactions [154]. Logic Feature selection (LogicFS) [96] combines logic regression with bootstrap analysis. It improves the variable selection and interpretation so that SNP interactions are directly identifiable. Monte Carlo Logic regression (MCLR) [155] is proposed by merging the Markov chain Monte Carlo and logic regression, an adaptive regression methodology to identify SNP interactions associated with a disease outcome. Full Bayesian logic regression (FBLR) [150] is proposed by combining logic regression with the Bayesian method. Briggs proposed a four-stage methodology by combining RF and logistic regression models to test for epistasis. Genetic programming for association studies (GPAS) [90] directly searches logical expression in disjunctive normal form (DNF) using genetic programming (GP). It iterates until the number of generations reaches the predetermined number of generations or till it achieves the desired fitness level [24]. Modified logic regression-gene expression programming (MLR-GEP) [24] is based on LR that uses a stochastic searching algorithm, and gene expression programming (GEP) to increase the computational efficiency. GEP has the advantages of both genetic algorithm (GA) and GP.

#### 2.4.6 Bayesian Approaches

Bayesian models are an alternative strategy for regression models that incorporate prior knowledge and accumulated experience into probability calculations. Lunn [156] proposed Bayesian based stepwise regression that detects the single-locus models and is implemented in the software WinBUGS. Further, inclusion of interaction effects is straight forward. Bayesian epistasis association mapping (BEAM) [29] is a Markov chain Monte Carlo (MCMC) based approach that detects multi-locus SNP interactions in GWAS. It uses the Metropolis-Hastings method to assign group labels to a locus. The distribution of genotypes is different between cases and controls if SNPs are related to the disease. Predictors are divided into three groups: group 0, group 1, and group 2. Group 0 consists of SNPs which do not relate to the disease. Group 1 consists of SNPs

which contribute to disease in the presence of the main effects. Finally, group 2 contains SNPs that contribute to disease risk due to interaction effects [6]. The statistical significance of the algorithm is inferred using B-statistic. The results demonstrated that the approach is more powerful and computationally feasible than existing methods. BEAM is successfully applied to Wellcome trust case control consortium (WTCCC) Crohn's data [6]. Fast epistatic interactions detection using Markov blanket FEPI-MB [30] is a novel and fast Markov Blanket method to detect SNP interactions by heuristic search. The Markov Blanket is defined as a set of variables that are probabilistically independent of target variables from all other variables [157]. Bayesian Network based epistatic association study (bNEAT) [31] is proposed to address the small sample problems. The method employs the Branch and Bound (B&B) algorithm for training the data. The results demonstrated that bNEAT outperformed Markov based methods when the sample size is small. Although, Bayesian approaches increase the power and decrease false positives, they are not suitable for detecting SNP interactions in the absence of main effects. These approaches perform poorly when the sample size is small.

#### **2.4.7 Ant Colony Optimization Approaches**

Ant colony optimization (ACO) is a probabilistic and positive feedback approach, which is modeled by the behaviors of ants [158]. Ants communicate with each other through pheromone levels to find the optimal path to a food source. If an ant finds the shortest distance, it increases pheromone levels along the path it travels. Subsequently, other ants follow the path with increased levels of pheromone by creating a positive feedback. AntEpiSeeker is an iterative procedure, which searches parallelly for SNP interactions in large-scale association studies [92]. It adopts a two-stage optimization derived from ACO. In the first stage, highly suspected SNP sets determined by  $\chi^2$  and reduced SNP sets determined by top ranking pheromone levels are searched. The second stage exhaustively searches for the SNP interactions. The approach detects pure interactions and the interactions among SNPs with large marginal effects. Further, the approach implements a new procedure to reduce false positives. The results demonstrated that the method with minimized false positives significantly outperforms other existing methods on large scale data by reducing FDR. MACOED is an ACO based multi-objective heuristic search approach to detect genetic interactions [159]. It combines Logistic

regression and Bayesian network methods resulting in high power with low false-positive rate. A memory based multi-objective ACO is designed to reduce space and time complexity of the high-dimension problem.

Ant colony optimization algorithm (ACA) coupled with logistic regression is proposed for gene interaction studies involved with large number of SNPs [160]. The method is implemented using sliding window (SW/H) and single-locus genotype association (RG). It obtains optimal solution in less number of iterations compared to other methods that adopts ACO. An ACO based approach is proposed by integrating a novel encoding and tournament selection mechanism to discover SNP interactions associated with type II diabetes, obtained from WTCCC [161]. The likelihood of two-way interaction associations due to chance is determined by the p-value of each combination. The study further analyzed the higher-order interactions (combinations of 3 SNPs) within reasonable computation time. Decision tree and contingency table models are hybrid with ACO to identify SNP interactions in GWAS [162]. The study searched vast space of interactions between SNPs by using ACO, decision tree and contingency table models discovering the most discriminatory arrangements of those interacting SNPs from a statistical perspective. A prototype of probabilistic search wrapper is developed with expert knowledge for pheromone updating rule by ACO metaheuristic [91]. Expert knowledge is obtained from Tuned ReliefF (TuRF). The wrapper is integrated with MDR to analyze interactions in GWAS [163]. The performance of the method is assessed by varying distribution parameter, the retention factor and weights of expert knowledge. However, currently studies are not transparent about how interactions are tested [3]. Detecting higher-order interactions (such as three-locus interactions) require multilayer hypothesis testing, which makes these methods as difficult as regression methods.

#### 2.4.8 Other methods

Some of the other exhaustive methods to detect SNP interactions are briefly reviewed in this section. Fast ANOVA [164] is an efficient algorithm for performing ANOVA tests on SNP pairs in a batch mode. ANOVA test is a widely used statistical method for quantitative phenotype association studies. FastANOVA constructs an upper bound (derived from sum of two terms) of the two-locus ANOVA test. The two terms are based

on single-locus ANOVA test, and on SNPs and independent of any phenotype permutation. SNP pairs are organized into groups by sharing same upper bound for each group. The experimental results demonstrated FastANOVA is faster than the brute force implementation of two-locus ANOVA test, by identifying the most significant SNP pairs. COE [165] is a generalized approach to find an optimal solution for detecting two-locus interactions. It analyses large scale data by utilizing the convexity of statistical tests. SNP pairs are grouped and indexed by their genotypes. These SNP-pairs are pruned by utilizing the upper bound and the indexing structure without compromising the optimality of the results. The statistical significance of the findings is assessed by permutation.

Boolean operation based screening and testing (BOOST) [22] discovers all pairwise interactions in genome-wide SNP data. A Boolean representation of the data used in the method increases space and CPU efficiency. It is a model based approach and performs two stage search method (screening and testing). It has demonstrated improved performance over PLINK [153]. Null simulation is performed under two scenarios (without linkage disequilibrium (LD) and with LD) for testing type I errors. Type I error rate of BOOST is reduced compared to PILINK as p-values are calculated by using  $\chi^2$  distribution with degrees of freedom  $\leq 4$ . Tree based epistasis association mapping (TEAM) [21] is a model free approach that addressed the heavy computation arising due to permutation test. The contingency tables for epistasis tests are updated by using the minimum spanning tree without scanning all individuals. Family-wise error rate (FWER) and false discovery rate (FDR) controlling procedures are used to reduce error rate by supporting statistical tests based on contingency tables. Genome-wide interaction search (GWIS) is a platform based on classification and novel rigorous statistical tests (based on receiver operating characteristic curve) [166]. It is a model free approach based on analytical solutions to hypothesis-based testing, rather than computationally expensive cross validation and permutation methods. The approach exhaustively searches all pair-wise SNP interactions, and is implemented on both CPU and GPU by reducing the execution time over previous methods.

**Table 2.3:** Data mining and machine learning approaches for detecting SNP interactions [7]: © 2018 IEEE

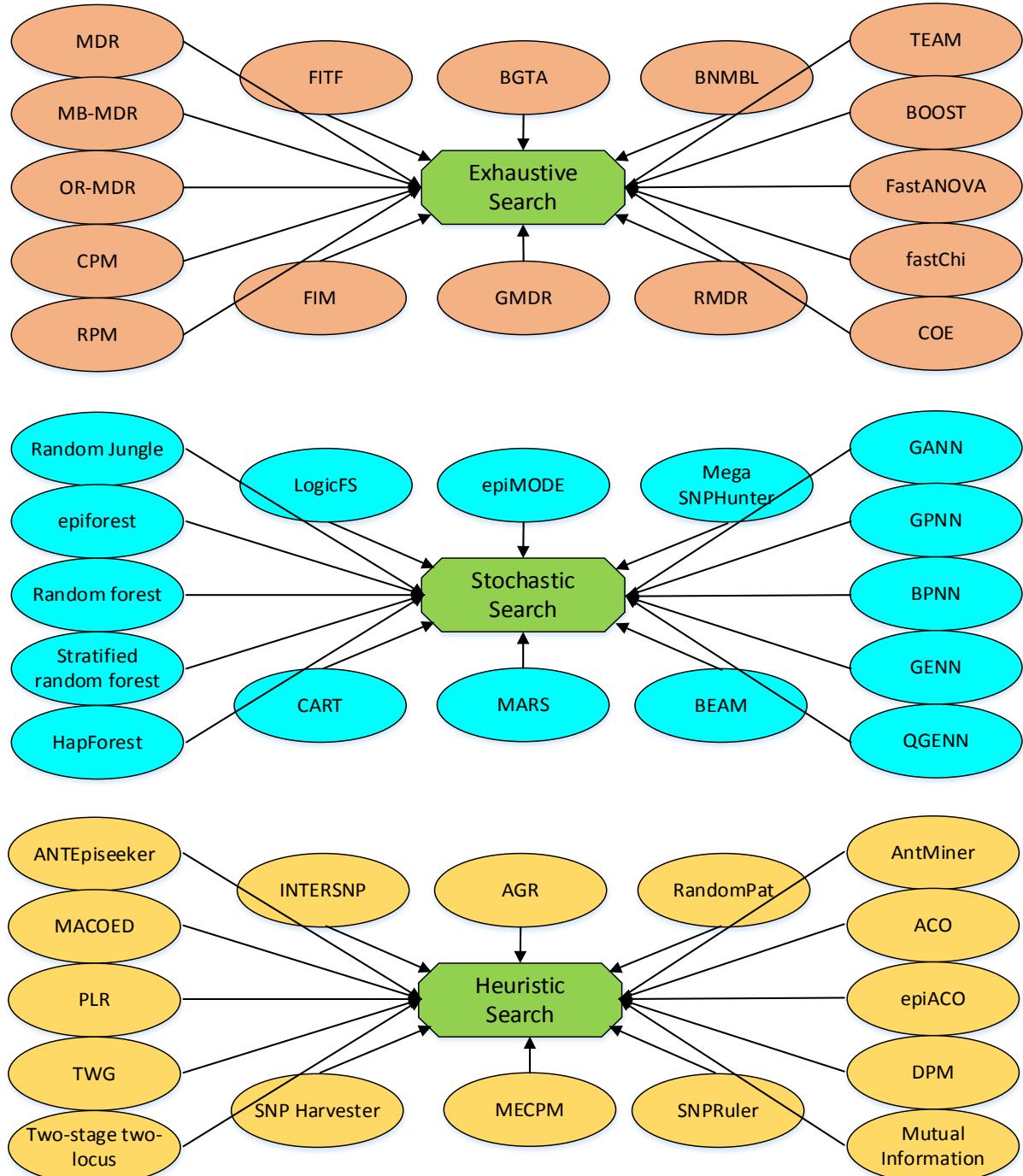
Methods	Examples	Real Datasets	Strengths	Limitations
<b>Multifactor dimensionality reduction</b>	MDR [17], Velez [108], Lee [110], Pattin [111], Namkung [112], OR-MDR [113], MB-MDR [78], GMDR [114], and RMDR [116].	Sporadic breast cancer [17], smoking and bladder cancer [116], chronic fatigue [113], bladder cancer [115], and nicotine dependence [114].	<ul style="list-style-type: none"> <li>➤ Model free, non-parametric approach that identifies higher-order interactions.</li> <li>➤ Reduces false positive rates by cross-validation.</li> <li>➤ Power remains high with 5% missing data and genotypic error.</li> <li>➤ A quantitative measure of disease can be identified in OR-MDR.</li> <li>➤ Models are easily interpreted in RMDR.</li> <li>➤ MB-MDR improves power and reduces false positive errors.</li> </ul>	<ul style="list-style-type: none"> <li>➤ Model scale up to analyse small number of SNPs.</li> <li>➤ It lacks in assessing the risk levels.</li> <li>➤ Restricted to balanced datasets.</li> <li>➤ The power of the model is significantly reduced nearly by 50% in the presence of noise.</li> <li>➤ Computationally intensive when SNPs exceed 10.</li> <li>➤ Empty cells are not classified.</li> <li>➤ False positive errors and false negative errors are high when the test data and the whole dataset are equal.</li> </ul>
<b>Random Forest</b>	RF [122], EpiForest [25], RJ [27], RFCouple [128], BOOST [22], SNPHarvester [129], SRF [130], TRM [131], and MINGRF [132].	Alzheimer's disease [167], sclerosis [168], Crohn's disease [27], Bladder cancer [132], Familial combined hyperlipidemia [169], Colon cancer [169], and ovarian cancer [170], asthma related gene ADAM33 [133], and	<ul style="list-style-type: none"> <li>➤ It does not overfit the data.</li> <li>➤ It outperforms the prediction when the trees do not exhibit correlation with each other.</li> <li>➤ It determines the pair-wise proximity among the samples and it estimates the importance of each attribute in the training stage.</li> <li>➤ RJ is computationally efficient and</li> </ul>	<ul style="list-style-type: none"> <li>➤ Power of detecting interacting SNPs is reduced, as it requires a marginal effect for at least one of the interacting SNP.</li> <li>➤ In few cases, it underestimates important scores of SNPs without marginal effects.</li> <li>➤ RF detects interactions only in large data.</li> </ul>

	rheumatoid arthritis [171] and Cancer [131].	designed to analyse the genome-wide scale.	➤ RJ fails when main effects are weak.
<b>Neural networks</b>	GANN [136], GPNN [137], GENN [138], [135], Parkinson's disease and QGENN [139].	Childhood allergic asthma [172], alzheimer's disease [137], breast disease [137], colorectal disease [137], and prostate's disease [137].	<ul style="list-style-type: none"> <li>➤ These models can deal with large datasets and improves the power.</li> <li>➤ Performance is high in the presence of noise.</li> <li>➤ It can predict the data where the disease outcome is unknown.</li> <li>➤ GPNN and GENN optimises NNs to discover the presence of non-functional SNPs.</li> </ul>
<b>Support vector machines</b>	Chen [142], Ozgur [144], Shen [145], Fang [146], GESVM [147], and Zhang [148].	Prostate cancer [142], Prostate cancer [144], Parkinson disease [145], Type 2 diabetes [173], COGA [146], Human cancer [148], and Disease model M1 and M2 [147].	<ul style="list-style-type: none"> <li>➤ Output can be easily interpreted and robust to noise.</li> <li>➤ It does not trap at local minima and not prone to over-fitting.</li> <li>➤ Does not require user defined decisions for classification and is ready to be generalized to new structures.</li> </ul>
<b>Regression models</b>	LR [24], PLINK [153], Bladder cancer [149],	➤ They enhance the prediction of the	<ul style="list-style-type: none"> <li>➤ Large number of genotype</li> </ul>

	LogicFS [96], MCLR [155], FBLR [150], and MLR-GEP [24].	Sporadic breast cancer [96, 150], Trachomatoustrichiasis [151], Heart disease [155], Myocardial infarction and Stroke [152].	models by using Boolean combinations. ➤ Performs well on a dynamic phenotype variable with expected marginal effect.	combinations requires small p-values. ➤ Computationally intensive due to curse of dimensionality. ➤ Model specific, with too many empty cells in contingency table.
<b>Bayesian models</b>	Lunn[156], BEAM [29], FEPI-MB [30], and bNEAT [31].	Crohn's disease [6], and age related macular degeneration [174].	➤ It computes measures of evidence. ➤ The biological knowledge is adapted in the models using rational and quantitative way. ➤ The limitations of p-values are reduced by modelling assumptions.	➤ It requires explicit assumptions on associated SNPs due to computational constraints. ➤ Specifying prior probability distribution for the models is complex. ➤ Computationally intensive.
<b>Ant colony optimization approaches</b>	AntEpiSeeker [92], MACOED [159], ACA [160], and Wrapper intergrated with MDR [163].	Rheumatoid arthritis [92, 175], type II diabetes [161, 162, 175], and Late-onset Alzheimer's disease [159].	➤ It searches parallelly for SNP interactions in large-scale association studies. ➤ Positive feedback leads to identify the optimised solutions. ➤ Minimises the false positives by reducing FDR.	➤ These studies are not transparent on how SNP interactions are tested. ➤ These methods are as difficult as regression methods due to multilayer hypothesis testing.

## 2.5 Summary of methods in view of searching strategies

The computational approaches to investigate the multiple interactions between SNPs are divided into three categories based on the search strategies used: exhaustive search, stochastic search, and heuristic search [176].



**Figure 2.3** Methods based on searching strategies to detect epistasis (adopted from [176]).

### **2.5.1 Exhaustive Search Methods**

The simplest way to search for interactions between multi-locus SNPs is by searching exhaustively using chi-square, exact likelihood ratio, and entropy based tests. The exhaustive algorithms enumerate all possible combinations of SNPs in genotype data and perform the interaction test for each combination. Mostly, exhaustive methods are used to discover two-locus interactions. More than two-locus or higher-order interactions are not scalable due to high computational burden (number of tests increases exponentially with the order of interactions that are considered) by losing statistic power. Hence, the success of these methods depends on sample size, model complexity, and expected effect size. Use of optimal algorithms can improve the performance and stability of these methods. Pioneering works in exhaustive search [176] include the combinatorial partitioning method (CPM) [14], multifactor dimensionality reduction (MDR) [17], extensions of MDR based approaches [120], focused interaction testing framework (FITF) [12], restricted partitioning method (RPM) [15], full interaction model (FIM) [177], 1-degree-of-freedom [178], backward genotype-trait association (BGTA) [179], FastANOVA [164], COE [165], fastChi [180], Boolean operation-based screening and testing (BOOST) [22], tree-based epistasis association mapping (TEAM) [21], and Bayesian network minimum bit length score (BNMBL) [181].

### **2.5.2 Stochastic search methods**

Stochastic search methods use random sampling to search the space of SNP interactions, instead of enumerating all possible combinations. The performance of these methods depends on random chance to select SNPs associated to a phenotype. As number of SNPs increases, the chances of identifying correct interacting SNPs drops. Further, execution time increases as number of iteration for sampling is huge. Hence, they may not be suitable for genome-wide datasets. Some of the pioneering works in stochastic search include classification and regression trees (CART) [121], random forests [182], multivariate adaptive regression spline (MARS) [11], Monte Carlo logic regression [155], genetic programming optimized neural network (GPNN) [172], epistatic module detection (epiMODE) [183], Bayesian epistasis association mapping

(BEAM) [29], HapForest [28], LogicFS [96], detection of EPIstatic interactions using random FOREST (epiForest) [25], and Random Jungle [27].

### **2.5.3 Heuristic search methods**

Heuristic search methods adopt machine learning techniques (such as neural networks and predictive rules) to search for higher order SNP interactions. These methods run fast by adopting optimal heuristic information and prior biological knowledge data. Power is high for moderate GWA datasets with efficient screening process. However, in these strategies, power is lost at higher-order interactions with insignificant marginal effect. Further, these methods lack in controlling false discovery rate (FDR). Some of the pioneering works of heuristic approaches are Trimming, weighting and grouping (TWG) [184], two-stage two-locus [185], penalized logistic regression (PLR) [9], INTERSNP [186], association graph reduction (AGR) [187], SNPHarvester [129], maximum entropy conditional probability modelling (MECPM) [188], AntEpiSeeker [92], Discriminative pattern mining (DPM) [189], mutual information [190] and SNPRuler [191].

## **2.6 Data simulation and Real-world data for evaluating the models**

Along with the rapid development of SNP interaction methods, data simulation plays an important role in the aforementioned performance assessments and in studying hypotheses on genomic variations affecting the disease. It is difficult to evaluate the performance of novel interaction models when real data are used due to unknown SNPs related to disease susceptibility. Hence, the performances of the models are evaluated on simulated datasets before their findings are confirmed on real-world data. A number of accurate and efficient simulators have been developed to generate simulated data under realistic scenarios, which includes the effect of natural selection, recombination, gene conversion and complex demographic and environmental factors [192]. This section based on the review [71], introduces some of the currently popular software packages for simulating the genetic data. Most of the data simulators in the literature fall under three categories: coalescent, forward-time, and resampling [192-195].

Furthermore, this section introduces briefly introduces the real-world data available publically to test the performance of the models, which detect SNP interactions in GWAS.

### 2.6.1 Coalescent Simulation

The coalescent or backward-time approach was originally developed by Kingman in 1982 [196, 197]. The basic approach was extended by including recombination, selection and other complex evolutionary methods [71]. Coalescent simulators [193, 198-200] have two processes: coalescent and permutation. The ancestral history of a sample of individuals originating from MRCA (most recent common ancestor) is described in the coalescent process [192]. That is, all SNP alleles are tracked to a single ancestor based on the derivation of an ancestor with survived progeny. The permutation process is followed by the coalescent process where mutation, recombination, selection, and other complex evolutionary models are introduced [201]. The process describes how and when alleles are mutated over time. The simulated data is generated through evolutionary changes originating from MRCA. The Wright-Fisher (W-F) model is implemented in the coalescent process. It describes the transmission of genes from one generation to the next and the evolution of an idealized population [71].

The coalescent simulation is computationally efficient as it only traces the history of lineages based on backward-in-time with survived offspring in the current population [192, 202]. Further, it maximizes the probability of the given data by estimating the parametric values [203]. These coalescent simulators are used to simulate genotypes under different demographic scenarios, but do not generate phenotypes. However, generating phenotypes are also required to detect associations between genetic and phenotypic variations. Some of the coalescent simulators are listed in Table 2.4, by extending the models with more realistic scenarios of evolution and demography. The considerable drawbacks of coalescent simulators are: ignored unsampled members of the population, not suitable to track complete ancestral information, and cannot handle diploid-specific effects in complicated scenarios (such as in complex epistasis models) [192]. Hence, these simulation frameworks cannot simulate multiple epistatic effects into a single dataset. Tools such as CoaSim, has an option of mapping qualitative phenotypes onto simulated genotypes [200]. Phenosim

[204] is a software used to add phenotypes onto genotypes that are generated by coalescent simulation tools. It simulates both quantitative and qualitative phenotypes by differentiating additive and epistatic effects between causal genetic variants. Hence, the tool can simulate case-control based samples, and can be used to search quantitative trait nucleotides (QTNs) of a complex disease. Additionally the tool provides a user-defined architecture, by allowing the users to assess the influence of different factors for association mapping, varying number of markers, and estimating optimal sample size for a given study. The output of the tool can be directly fed as an input to different GWAS tools.

## 2.6.2 Forward-time Simulation

Forward-time simulation [192, 205-211] starts from an initial ancestral population and follows its evolution over a number of generations under various evolutionary and demographic scenarios. The final population is achieved by the stopping criteria of the specified number of generations [192]. In this simulation framework, the complete ancestral information can be tracked at any generation. Hence, the approach is more flexible and can simulate arbitrarily complex genetic disease models under various migration and demographic scenarios. They can simulate more than one causal factor, including epistasis models. Computational time and memory usage are the critical issues to be considered when large scale genome-wide SNP samples are generated [192]. Further controlling minor allele frequencies (MAFs) between multiple epistasis models are still challenging.

## 2.6.3 Resampling Simulation

Random sampling simulators [192, 201, 212] generate samples by random selection (such as bootstrapping) from existing datasets. These models can generate simulated datasets of LD patterns and MAFs. They can generate a single simulation dataset from a multiple epistasis models. This method is useful to validate the genotype-phenotype associations and evaluate the SNP interaction models. Though resampling simulators are faster than coalescent and forward-time simulators, the length of simulated SNPs are limited and are not suitable for generating genomic diversity [192, 201]. A number

of simulators based on coalescent, forward-time, and resampling simulations are listed in Table 2.4.

**Table 2.4:** Data simulators for evaluating the models [71, 213]: © 2018 IEEE

Framework	Simulator	Purpose	Weblink	Reference
<b>Coalescent</b>	CoaSim	CoaSim is a tool (based on Monte Carlo approach) for recombination and gene conversion under various demographic models. It maps qualitative phenotypes onto simulated genotypes.	<a href="http://users-birc.au.dk/mailund/CoaSim/index.html">http://users-birc.au.dk/mailund/CoaSim/index.html</a>	[200]
	fastsimcoal2	It generates multi-locus allelic data based on infinite allele mutation model under arbitrarily complex evolutionary models.	<a href="http://cmpg.unibe.ch/software/fastsimcoal2/">http://cmpg.unibe.ch/software/fastsimcoal2/</a>	[214]
	GENOME	A rapid coalescent based simulator that simulates SNP data with LD and MAF.	<a href="http://www.sph.umich.edu/csg/liang/genome/">http://www.sph.umich.edu/csg/liang/genome/</a>	[199]
	macs	It simulates whole-genome SNP data as a Markovian process.	<a href="http://www-hsc.usc.edu/~garykche/">http://www-hsc.usc.edu/~garykche/</a>	[198]
	mlcoalsim	Multi-locus coalescent simulation is designed to perform multi-locus analysis by including heterogeneity.	<a href="http://code.google.com/p/mlcoalsim-v1/">http://code.google.com/p/mlcoalsim-v1/</a>	[215]
	Msms	It generates sequence samples under neutral, epistasis, and a single-locus selection models.	<a href="http://www.mabs.at/ewing/msms/index.shtml">http://www.mabs.at/ewing/msms/index.shtml</a>	[216]
	SIMCOAL2	It is an extended version of SIMCOAL that simulates datasets with arbitrary recombination between partially linked loci under various demographic models and also simulates SNP data along a single chromosome.	<a href="http://cmpg.unibe.ch/software/simcoal2/">http://cmpg.unibe.ch/software/simcoal2/</a>	[217]

---

<b>Forward-time</b>	SimRare	It is a rare variant simulation and analysing tool, that implements realistic evolutionary models (includes multi-locus) by generating variant data for different population.	<a href="http://code.google.com/p/simrare/">http://code.google.com/p/simrare/</a> [218]
	genomeSIMLA	It uses Hardy-Weinberg equilibrium to simulate the data from generation to generation through forward-time. It generates purely epistatic gene-gene interactions in family based and case-control based studies.	<a href="http://chgr.mc.vanderbilt.edu/genomesimla/">http://chgr.mc.vanderbilt.edu/genomesimla/</a> [205]
	Nemo	It simulates the evolution of genetic markers, life history traits, life cycle of events, and phenotypic traits.	<a href="http://nemo2.sourceforge.net/">http://nemo2.sourceforge.net/</a> [206]
	QuantiNemo	It is designed to analysis the quantitative traits with explicit genetic architecture.	<a href="http://www2.unil.ch/opgen/softwares/quantinemolo/">http://www2.unil.ch/opgen/softwares/quantinemolo/</a> [207]
	SimAdapt	It combines with a landscape cellular automaton to represent evolutionary processes using NetLogo environment.	<a href="http://www.openabm.org/model/3137">http://www.openabm.org/model/3137</a> [208]
	simuGWAS	It simulates data for genome-wide association studies.	<a href="http://simupop.sourceforge.net/cookbook/simugwas">http://simupop.sourceforge.net/cookbook/simugwas</a> [209]
	SimuPOP	It is a population genetics simulation environment based on forward-time simulation.	<a href="http://simupop.sourceforge.net/cookbook/simugwas">http://simupop.sourceforge.net/cookbook/simugwas</a> [210]
<b>Resampling</b>	GWAsimulator	It simulates genotype data for case-control or population	<a href="http://biostat.mc.vanderbilt.edu/wiki/Main/">http://biostat.mc.vanderbilt.edu/wiki/Main/</a> [212]

---

		samples.	<u>GWAsimulator</u>	
HAPGEN		It simulates case-control datasets at SNP markers.	<a href="https://mathgen.stats.ox.ac.uk/genetics_software/hapgen/hapgen2.html">https://mathgen.stats.ox.ac.uk/genetics_software/hapgen/hapgen2.html</a>	[219]
HAP-SAMPLE		It is a web application that simulates SNP genotypes for case-control and affected-child trio studies.	<a href="http://www.hapsample.org/">http://www.hapsample.org/</a>	[220]
<b>Other Frameworks</b>	epiSIM	It simulates multiple epistasis, linkage disequilibrium patterns, and haplotype blocks for genome-wide interaction analysis.	<a href="https://sourceforge.net/projects/episimsimulator/files/">https://sourceforge.net/projects/episimsimulator/files/</a>	[201]
	GAMETES	It generates complex SNP models with pure, strict, epistatic models with random architecture.	<a href="http://sourceforge.net/projects/gametes/?source=navbar">http://sourceforge.net/projects/gametes/?source=navbar</a>	[221]

## 2.6.4 Real-world Data Applications

There are a number of available real datasets to evaluate the effectiveness of the interaction models. WTCCC is a group formed in 2005 by over 50 research groups in the UK to explore the utility, design, and analyses of GWAS [175]. The initial GWA studies were performed on 2000 cases and 3000 shared controls for 7 complex diseases (type 1 diabetes, type 2 diabetes, coronary heart disease, hypertension, bipolar disorder, rheumatoid arthritis and Crohn's disease) [222]. Further there were two additional GWA studies. The first study was performed for tuberculosis with 1500 cases and 1500 controls. The second study was performed for malaria, breast cancer, multiple sclerosis, ankylosing spondylitis and autoimmune thyroid disease in 1500 shared controls and 1000 cases. The data is obtained from WTCCC data access committee to qualified researchers for appropriate use. The cancer genome atlas (TCGA) dataset, comprise two petabytes of genomic data in 33 types of cancer [223, 224]. It is publicly available from genomic data commons (GDC) (<https://gdc-portal.nci.nih.gov/>). The Leukemia dataset

is available publicly from gene patterns with 38 bone marrow samples for training and 35 bone marrow samples for testing [225]. The GWA study of celiac disease dataset can be downloaded from <https://www.ncbi.nlm.nih.gov/gap/>.

Many research studies have analyzed the performance of their interaction models by using real datasets, to discover unknown multi-locus SNP interactions associated with complex diseases. Some of these studies based on real data are listed in Table 2.3. The power of the models to detect SNP combinations are reduced (when compared to the evaluations performed on simulated datasets) due to the challenges posed by real-world data. These challenges are due to the presence of noise (such as genotyping error, phenotype, missing data, and genetic heterogeneity [36]), missing heritability, and small sample size. Hence, new effective and flexible methods are required to discover highly significant SNP combinations that contribute to a disease.

## 2.7 Alternative Methods

The era of GWAS continues and the field of SNP interaction analysis will become more prevalent. A number of strategies were developed to identify the SNP interactions responsible for disease susceptibility. It is apparent that detecting these interactions in the large scale genome data is still a challenging and critical issue to be considered. Currently, many researchers are exploring alternative filtering methods, efficient computational algorithms and pathway approaches for interaction analysis.

### 2.7.1 Biological Filters

The role of epistasis and its biological relevance to complex disease is still developing and its impacts need to be determined [35]. The SNP interactions which do not have independent effects would be missed by filtering with simple statistics, such as ranking the SNPs by using either p-value or effect size (selects highly ranked SNPs), and removing the SNPs based on a threshold value (selects SNPs exceeding a threshold value). Filters based on machine learning approaches may or may not pick up the interacting SNPs, depending on the nature of the interactions. It is worthwhile to explore novel or alternative approaches which apply existing biological knowledge to filter the SNPs that may interact with each other. Some of the biological knowledge based databases are Biomolecular Interaction Network Database (BIND) [226],

Molecular Interaction Database (MINT) [227], Human Protein Reference Database (HPRD) [228], Database of Interacting Proteins (DIP) [229], Biological General repository for interaction datasets (BioGrid) [230], and Reactome [231] [232]. Researchers can extract the required gene information from these knowledge databases to filter out the non-potential interacting SNPs. Consequently, a number of biologically inspired filters have been developed to detect SNP interactions by reducing the search space. The biological filters can be derived from the knowledge obtained from the interaction of SNPs through their biochemical pathways, networks, and location of SNPs. They are used to analyse the data efficiently and retain the knowledge for future use. For instance, Bio-filter uses the extrinsic biological knowledge to filter SNPs and prioritises SNP combinations based on the biological knowledge [233]. The success of these filters depends on the quality of information in the knowledge base. The quality of information can be assessed by the accuracy and completeness of the information. Further, these filters depend on biological patterns at cellular level and statistical patterns of SNPs observed at population level [77]. Some of the other drawbacks of using biological knowledge in association studies include [2], in reality, the publicly available biological knowledge being incomplete and restricted to the information available in the scientific domain. There is inherent knowledge bias as filtering is done purely based on the published knowledge [234]. These knowledge bases are flooded with false positive discoveries and by the non-publication of negative findings. Additionally, interacting analyses are performed only on the currently available knowledge prohibiting novel biology.

**Table 2.5:** Biological knowledge based databases

Database	Description	Web link
<b>Human protein Reference Database (HPRD)</b>	It stores the information related to domain architecture, post-translational modifications, interaction networks, and disease association for each protein.	<a href="http://www.hprd.org">http://www.hprd.org</a>
<b>Biological General Repository for Interaction Datasets (BioGrid)</b>	The interaction database complied through comprehensive efforts.	<a href="http://thebiogrid.org">http://thebiogrid.org</a>

<b>Bimolecular Interaction Network Database (BIND)</b>	It stores the description, molecular complexes and pathways.	<a href="http://binddb.org">http://binddb.org</a>
<b>Molecular Interaction Database (MINT)</b>	It stores experimentally verified protein interactions mined from scientific literature.	<a href="http://mint.bio.uniroma2.it/mint/Welcome.do">http://mint.bio.uniroma2.it/mint/Welcome.do</a>
<b>Database of Interacting Proteins (DIP)</b>	Database with experimentally determined interactions between proteins.	<a href="http://dip.doembi.ucla.edu/dip/Main.cgi">http://dip.doembi.ucla.edu/dip/Main.cgi</a>
<b>Reactome</b>	Manually created peer-reviewed pathway database with PubMed literature.	<a href="http://www.reactome.org/ReactomeGWT/entrypoint.html">http://www.reactome.org/ReactomeGWT/entrypoint.html</a>

## 2.7.2 Computational Optimization

Computational burden increases exponentially as the order of interaction analysis increases. For example, if data consist of 100,000 SNPs, there will be  $\binom{100000}{2} = 5 \times 10^9$  SNP combinations for 2<sup>nd</sup> order interaction search. Similarly, there will be  $1.7 \times 10^{14}$ ,  $4.2 \times 10^{18}$ ,  $8.3 \times 10^{22}$ ,  $1.4 \times 10^{27}$  SNP combinations for 3<sup>rd</sup>, 4<sup>th</sup>, 5<sup>th</sup> and 6<sup>th</sup> order interactions respectively. Searching exhaustively for SNP interactions (responsible for a disease) in the full set of higher-order SNP combinations is computationally infeasible. It is estimated that it would take years to examine three-way interactions in large scale GWAS datasets by using current machine learning approaches [23]. Hence, a number of current computational strategies are being explored to overcome these computational limitations. Parallel computing, grid computing and cloud computing are some of the optimization strategies explored in interaction studies. Parallel computing is increasingly used in statistical and computational genomics due to the rapid advancements achieved in the field of supercomputing. Parallelization is adopted in the algorithms to improve the speed and efficiency of the current methods to analyze SNP interactions. Grid computing and cloud computing are the alternatives to parallel computing infrastructure advancements that have attracted growing interest from many researchers. These approaches deploy a group of remote servers and software networks for sharing and accessing the genome data online. However, a grid is

not always a cloud. A grid is a collection of computer resources from multiple domains to reach a common goal. Grids are a form of distributed computing with non-interactive workloads [235]. Cloud computing is the use of computing resources that are delivered as a service over a network [235]. It allows extension of many algorithms to parallel implementations for increasing computation speed by acquiring more cloud computing resources, such as memory and processing power.

FastEpistasis [236] is the extension of PLINK [153] by implementing parallel computing to detect epistasis of a continuous phenotype. It has been successfully implemented on 5000 samples with 500000 SNPs. EPISNPmpi [237] is developed by extension of the Kempthorne model [238] by using parallel computing. It requires 20 hours to detect two-way interactions of 100000 SNPs. GPUs are high performance parallel processors with high computation speeds for accessible programming interfaces. MDRGPU [239] is a tool that runs MDR using the PyCUDA library to boost performance and to run faster than MDR. EpiGPU [240] is a tool developed by using graphics processing units (GPUs) that parallelize exhaustive searches with quantitative traits. The authors reported a  $92 \times$  increase in the speed of an exhaustive pairwise SNP-SNP interaction scan [240]. SHEsisEpi [241] is developed by combining GPUs parallel computation and attribute selection strategy to improve the efficiency of detecting SNP interactions. eCEO [242] uses cloud computing to detect SNP interactions and has demonstrated the efficiency of the model on 40 node clusters. Hybrid cluster cloud high performance computing (HCC-HPC) [202] is a distributed analytical approach to detecting SNP interactions and their associations allowing for interactions (omnibus) by using cloud and cluster grids. GBOOST is a GPU implementation of BOOST by reducing the execution time to discover pairwise interactions [23]. GMDR-GPU implements parallel computation of GPU to analyze GWAS data to run faster than the previous version of GMDR [243]. EpistSearch is a parallelized hybrid tool that combines Pthreads and CUDA by taking advantage of CPU/GPU architecture [244]. SNPsyn is a heterogeneous, GPU and Intel MIC-accelerated software for exploration and discovery of SNP interactions using information-theoretic approach [245]. A cloud based dynamic clustering for higher-order genome-wide epistatic interactions detecting (DCHE) is proposed for discovering higher-order multi-locus (two-locus and three-locus) interactions by reducing the runtime [246].

Almost all the researchers have reported that, the speed of exhaustive search for pairwise interactions increase by adopting parallelization. However, only a few studies are successful in searching for third order interactions that too on a smaller size of GWAS [246-249]. The computational burden for exhaustive search of higher-order interactions (three or more) are still infeasible to handle, as search space increases dramatically even for small to medium GWAS datasets. In addition, due to advancements in high-throughput genotyping techniques, the number of SNPs are consistently increasing, which will hugely increase the search space. Hence, single handedly parallel computing cannot eventually replace the machine learning approaches with current computational optimization techniques. However, exploration of super computing and efficient optimization techniques in depth on interaction studies could be promising.

### **2.7.3 Pathway Approaches**

Development of methods and tools related to pathway analysis is ongoing and dynamic [250] . GWAS pathway analysis (GWASPA) integrates the results of GWAS and the genes in a known molecular pathway to test for association with a disorder [251]. These association results are assigned to the pathways and tested with computational tools and pathway databases [251]. Pathways represent a series of biological processes that lead to a cell function, factors leading to human disease, biosynthesis, metabolic process, and immune response [251]. The set of genes in the pathway indicates the order of gene interactions to achieve a specific task [252]. Alternative paths can also be considered with the same genes or different genes that lead to the same results. The pathway analysis tests the association of genes in the pathway that lead to complex traits. There are two approaches to identifying these pathways [251]. The first approach integrates the results of GWA studies and genes in the known pathway to test for the association with the disease. The second approach formulates a prior hypothesis of pathways which may be involved in the disorder. There is growing interest in pathway based approaches whose pathways are stored in web-based pathway databases.

Among 300 databases listed in Pathguide [253], some of the commonly used databases are Pathway interaction database (PID) [254], Kyoto encyclopaedia of genes and genomes (KEGG) [255], Gene ontology (GO) [256], Visualization and integrated

discovery (DAVID) [257], WTCCC [175], Biocarta [258], and Protein analysis through evolutionary relationships (PANTHER) [259]. The recent databases, such as alzGene [260] and UCSC cancer genomics browser [261], aggregate diverse information on a particular disease [250]. A number of pathway based studies in the current literature include PLINK set test [153], SNP ratio test [262], MEGENTA [263], PARIS [264], PATH [265], interSNP [266], GenGen [267], DAPPLE [268], and Gene set enrichment analysis (GSEA-SNP) [269]. These approaches have been successfully implemented for Crohn's disease [270], bipolar disorder [271], multiple sclerosis [272], Parkinson's disease [273], schizophrenia [274], rheumatoid arthritis [275] and type 1 diabetes [275]. The human genome constitutes only 1-2% of genes. Hence, efficient strategies to leverage both genic and non-genic data for pathway analysis may provide an increased capability to detect interactions [250]. These interaction results can be used to generate subnetworks from enriched pathways. The role of pathways and networks can be vital to revealing biological mechanisms behind complex diseases.

## 2.8 Chapter Summary

This chapter has provided a better understanding of SNP interactions and their relations with complex human diseases. Detecting these interactions in high-dimensional genomic data is difficult due to the growing number of genetic variants in human genetics. A number of efficient methodologies and computational techniques have been reviewed in this chapter. This chapter has briefly reviewed the factors to be considered while designing those methodologies. Advantages and disadvantages of current methodologies were also discussed in order to highlight the gaps to be considered while designing the new methodology. Further, the chapter has focused on the achievements in data simulation for evaluating the performance of these methodologies. This chapter also introduced some of the other alternative methods by combining knowledge in biology, statistics, and genetic epidemiology. In the next chapter, an associative based approach is implemented and studied for detecting SNP interactions in balanced and imbalanced datasets.

# Chapter 3

## Associative classification for detecting higher-order SNP Interactions

In current genetic epidemiology, a number of research studies have been reviewed in the previous chapter to detect epistasis with application of new computational techniques, and methods. Despite their limitations, the existing approaches identify the existence of major proportions of interacting genes at multi-locus. However, none of these models could expose SNPs at a locus which can have a stronger association with a disease, and a weaker association for another disease. In a few cases, a SNP may not be directly associated with the disease, but may influence the nearest genes to be associated with the disease. Further, the accuracy of the current models is degraded in imbalanced datasets by increasing the classification errors. Hence, there is no single model, which can reveal the complexity of genetic architecture by identifying disease causal SNPs, and their interaction effects between SNPs.

---

This chapter is based on the following publications:

- S. Uppu, A. Krishna, and R. P. Gopalan, "An Associative Classification Based Approach for Detecting SNP-SNP Interactions in High-dimensional Genome," in *Bioinformatics and Bioengineering (BIBE), 2014 IEEE International Conference on*, pp. 329-333, 2014 : © 2014 IEEE, "The original publication is available at <https://ieeexplore.ieee.org/document/7033602>".
- S. Uppu, A. Krishna, and R. Gopalan, "Detecting SNP interactions in balanced and imbalanced datasets using associative classification," *Australian Journal of Intelligent Information Processing Systems*, vol. 14, pp. 7-18, 2014.
- S. Uppu, A. Krishna, and R. P. Gopalan, "Rule-based analysis for detecting epistasis using associative classification mining," *Network Modeling Analysis in Health Informatics and Bioinformatics*, vol. 4, pp. 1-19, 2015: © 2015 Springer, "The original publication is available at <https://link.springer.com/article/10.1007/s13721-015-0084-3>".

Many researchers have shown that Associative Classification (AC) is more accurate than traditional classifiers [276]. The rules generated in AC can be stored and can provide reasoning to the classification. AC is also suitable for both categorical and discrete data. Mapping SNP-SNP interactions to a disease can be improved by integrating association rules, and classification. In this chapter, a new approach based on AC is implemented to identify the interactions more effectively than the existing methods. The rule based approach will classify the subjects by determining the complexity of interactions and their associations with the disease. The goal of this study is to evaluate the rule based approach on the simulated data by varying heritability, minor allele frequencies and case control ratio. The preliminary studies identified two-locus SNP interactions for both balanced and imbalanced datasets. Further, the approach is validated in terms of accuracy and compared with previous methods (such as MDR, Naïve Bayes, RF, NNs, and SVM) under same simulated scenarios. Though, AC showed only relatively small improvement in accuracy for balanced datasets, it outperformed existing approaches in imbalanced datasets.

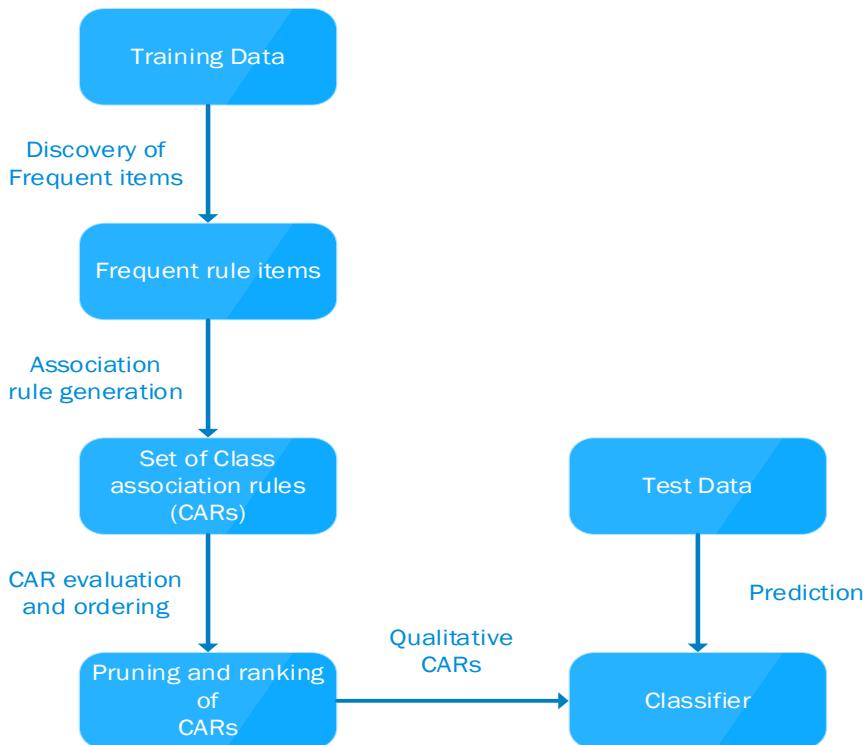
Subsequently, in this chapter, the research has been further extended to confirm the findings of the rule based approach over single-locus to six-locus models by varying heritability, minor allele frequencies, sample size and case-control ratios. Several experiments are conducted over two simulated scenarios to demonstrate the performance of the method. Further, the experiments are conducted to evaluate the performance of the method with some of the unsupervised algorithms, such as, KMeans and principal component analysis (PCA). Finally, the rule based approach is applied to a genetic dataset (sporadic breast cancer dataset) to identify the interacting SNPs that could lead to the disease.

In this chapter, the associative classification is briefly reviewed, and applied to the present problem in Section 3.1. Data simulation scenarios and real dataset are explained in Section 3.2 and Section 3.3 respectively. Section 3.4 includes data analysis of the rule based method. Finally, experimental results are evaluated and discussed in Section 3.5.

## **3.1 Associative classification**

Data mining is a process of searching for important information in terms of patterns or rules from vast data. Fayyad defined data mining as the important stage in knowledge discovery from databases (KDD) that extracts the useful patterns from data [277]. Some of the other stages of KDD are data selection, pre-processing, transformation, and evaluation. Data mining performs various tasks, such as, classification, clustering, regression, associative rule mining, and pattern recognition [278]. Classification is a key function in data mining, which builds a classifier (model) to predict the class labels of an unknown data. For example, in case-control studies, patients and non-patients (healthy) are classified as cases and controls respectively. The data classification is of two-step process by partitioning the data for training and testing. In training, the classifier is learned by analyzing training data with known class labels. In testing, the trained model predicts the class labels of unknown data objects. Association rule mining (ARM) is an another important task in data mining, which is proposed by Agrawal [279]. Initial studies were performed on market basket analysis to predict the relationships between the items purchased by the customer. ARM finds the interesting correlations between the frequently occurring data items in a data repository. The data items are considered as frequent, if they occur greater than or equal to the predefined threshold. These frequently occurring items (frequent items) are used to generate a special subset of association rules called class association rules (CARs).

Associative classification (AC) is a promising approach that integrates ARM and classification to build a classifier for prediction [276]. ARM discovers the descriptive knowledge from databases, and classification builds the model for categorizing new data. In general, the association rules generated from frequent item sets are used to classify data based on the class labels. The classification based on ARM (so called association classification mining (ACM)) builds classification system (also called as associative classifier) [280]. The steps involved in AC are illustrated in Figure 3.1 (based on [276]): a) Identifying frequently occurring conjunctions of attribute – value pairs (frequent item sets) in training dataset, b) Generating class based association rules (CARs) from frequent item sets, which satisfy minimum confidence and support criteria, c) Pruning and ranking these CARs to organize for the classification, and d) classifying the test dataset into predefined class labels.

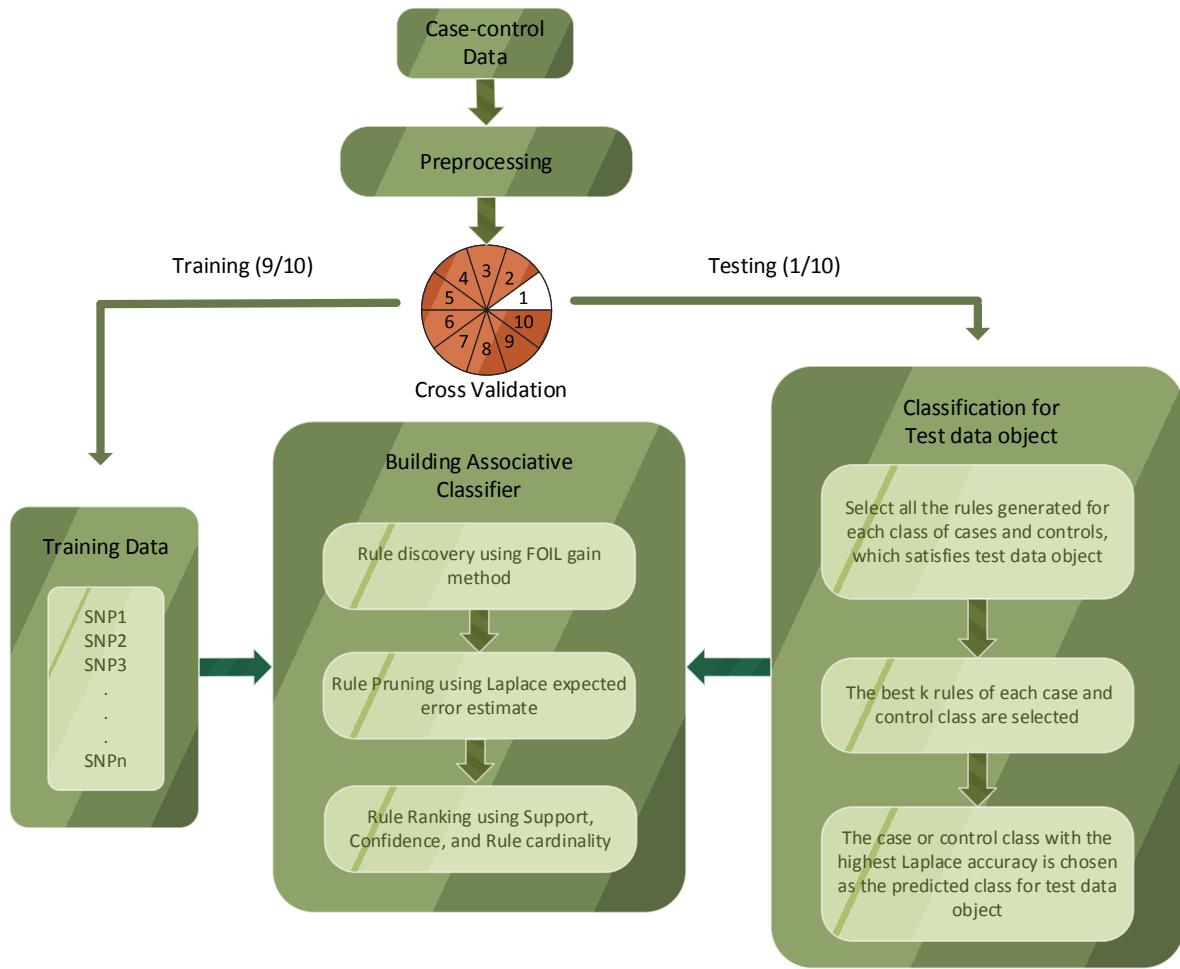


**Figure.3.1** Steps involved in Associative classification

A number of studies have been successfully implemented AC in data mining to build more efficient, and accurate classifiers than traditional techniques [276]. AC extracts all the hidden rules that would have missed by other classification algorithms. The rules produced by AC can be easily interpreted as they are simple if-then rules. This hybrid approach of association rules and classification mining was first proposed by Liu [281]. Apriori algorithm is used to generate CARs that satisfies the minimum threshold of support and confidence. It is a rule based classifier with single rule that cannot handle large number of rules. Hence, the research further progressed by implementing FP-growth algorithm to generate multiple CARs for the classification. Some of the popular AC algorithms are: classification based on association (CBA) [281], classification based on multiple association rules (CMAR) [282], classification based on predictive association rules (CPAR) [283], An associative classifier with negative rules (ACN) [284], positive and negative rules [285], live and let live ( L<sup>3</sup>) [286], adaptive associative classification (ADA) [287], multi-class multi-label associative classification (MMAC) [288], and multi-class classification based on association rule (MCAR) [289]. Different algorithms use different data layout, rule discovery, rule ranking, rule pruning, and prediction methods. The review and comparison of some of the these AC algorithms are studied in detail by Thabtah [276].

### 3.1.1 Workflow of the method

The main goal of AC is to generate a model with set of efficient rules. The generated model is used to predict the class labels of unknown data. That is, a classifier  $l \in R$  is trained to maximize the probability of  $l(v) = y$  for each test tuple [276]. Where mapping form of classifier is:  $R: S \rightarrow Y$ ,  $S$  is set of *itemsets*, and  $Y$  is set of class labels. Figure 3.2 (drawn based on [290]) represent the workflow of associative classification based on algorithm [283] is implemented to detect SNP interactions.



**Figure 3.2** Workflow of the implemented Associative classifier to detect SNP interactions

Attributes can be either categorical or continuous in AC. All the values are mapped to positive integers for categorical attributes, where else, discretization method is used for continuous attributes. Once, case-control data is preprocessed, ten-fold cross validation is performed for training and testing. Training dataset is used to build associative classifier. Rules are generated using FOIL gain method. Generated rules are pruned for

high quality rules using Laplace expected error estimate. Further, the rules are ranked using support, confidence, and cardinality. Initially, the rules are sorted based on confidence. If more than one rule has same confidence, they are sorted based on support. If two rules have same confidence and support, they are sorted based on lower antecedent length of the rule. If two rules have same confidence, support, and cardinality, they are sorted randomly. Finally, the trained classifier predicts appropriate class labels for test dataset. The classifier maps set of *itemsets* on to a set of classes.

### 3.1.2 Definitions

The current problem is formulized by using AC definitions (as reported in [276]) :

**Definition 1:** A row or a training case in  $D$  is represented by the name of an attribute  $SNP_i$  and its value  $v_i$ . Hence, an item is denoted as  $\langle (SNP_i, v_i) \rangle$ . Where,  $D$  is a training dataset of tuples  $|D|$ .

**Definition 2:** An *itemset* in a training object is represented as list of attribute names and it's values with a class  $c_i$ , denoted as  $\langle (SNP_{i1}, v_{i1}), (SNP_{i2}, v_{i2}), \dots, (SNP_{in}, v_{in}) \rangle$ .

**Definition 3:** A *ruleitem*  $r$  is defined as  $\langle itemset, c \rangle$ , where *itemset* is a antecedent, and  $c$  is a class,  $c \in C$ . Set of classes  $C$  is used to predict the unknown classes of test data. In case-control datasets,  $C$  can be either a case or a control.

**Definition 4:** The actual occurrence (*actoccr*) of  $r$ ,  $(actoccr(r))$  in  $D$  is the number of cases in  $D$  that matches  $r$ 's antecedent.

**Definition 5:** The support count (*suppcount*) of  $r$ ,  $(suppcount(r))$  in  $D$  is the number of cases in  $D$  that matches  $r$ 's antecedent and belongs to class  $c$  of *ruleitem*  $r$ . The support of  $r$  is represented as:  $sup(r) = \frac{suppcount(r)}{|D|}$ .

**Definition 6:** The occurrence of *itemset* (*occitm*)  $i$ ,  $(occitm(i))$  in  $D$  is the number of cases in  $D$  that matches  $i$ .

**Definition 7:** An *itemset*  $i$  passes threshold of *minsupp* when  $\left( \frac{occitm(i)}{|D|} \right) \geq minsupp$ .

All the *itemsets* that passes *minsupp* threshold are called as frequent *itemsets*.

**Definition 8:** A *ruleitem*  $r$  passes threshold of  $\text{minsupp}$  when  $\left(\frac{\text{suppcount}(r)}{|D|}\right) \geq \text{minsupp}$ . All the *ruleitems* that passes  $\text{minsupp}$  threshold are called as frequent *ruleitems*.

**Definition 9:** A *ruleitem*  $r$  passes threshold of  $\text{minconf}$  when  $\left(\frac{\text{suppcount}(r)}{\text{actocc}(r)}\right) \geq \text{minconf}$ . The confidence of  $r$  is represented as:  $\text{conf}(r) = \frac{\text{suppcount}(r)}{\text{actocc}(r)}$ .

**Definition 10:** A CAR is represented as  $\text{antecedent} \rightarrow c$ . Where *antecedent* of the rule is an  $(SNP_{i1}, v_{i1}) \wedge (SNP_{i2}, v_{i2}) \wedge \dots \wedge (SNP_{in}, v_{in})$ , and  $c$  is a class or *consequent* of the rule.

### 3.1.3 Data Representation

Data is represented horizontally for generating association rules. Horizontal layout, in which, the training dataset consist of number of rows with each row associated to a list of attribute values. Consider  $D$  be a relation of tuples, whose schema is represented by  $n$  distinct attributes  $SNP_1, SNP_2, \dots, SNP_n$ , and list of class attribute  $C$ . Let  $C$  be a finite set of class labels with case  $c_1$  and control  $c_2$  respectively, where,  $c_i \in C$ . The attributes are treated as categorical, where the class labels are known in training data instances in  $D$ , and the class labels are unknown in testing data instances. SNPs are bi-allelic that are mapped to a set of positive integers. The number of rows in  $D$  is represented by  $|D|$ . Each instance tuple in  $D$  is represented as  $t_i = (v_{i1}, v_{i2}, \dots, v_{in}, c_i)$  where  $v_{i1}$  is an item value for  $SNP_1$ ,  $v_{i2}$  for  $SNP_2$ , so on, and  $c_i$  is a class label. Association rule  $R$  is generated in the form of  $S \rightarrow Y$  which matches a tuple  $t \in D$  when  $S \subseteq t$ .  $S$  is the antecedent, which represents interacting SNPs associated with the class label and  $Y$  the consequent which represents either case or control. The rules are generated, whose support ( $\text{minsupp}$ ) and confidence ( $\text{minconf}$ ) are greater than or equal to minimum threshold values. Support and Confidence are the two parameters used to measure the quality of association rules. Support is the number of tuples in  $D$  containing  $S \cup Y$ , and confidence is the number of tuples matching  $S \cup Y$  divided by the number of tuples containing  $S$ . The time complexity of generating these rules and rule selection is huge when the datasets have a large number of rows and/or columns. Hence, FOIL (First Order Inductive Learner) Gain method is used to generate the rules.

### 3.1.4 Rule generation

The rules are generated by using the weighted FOIL gain method (greedy approach) used in CPAR [283]. The approach measures the information gained by the current rule by adding an item to its antecedent. For class C, let  $|P|$  be the positive data objects (training objects that contain class C) and  $|N|$  the negative data objects (training objects that class C never occurs) in the training dataset, which satisfy the current rule  $r$ . After item  $p$  (the best gain attribute) is added to the rule, there will be  $|P^+|$  positive data objects and  $|N^-|$  negative data objects in the new associated rule in the training dataset. The FOIL gain of item  $p$  is defined as:

$$gain(p) = |P^+| \left( \left( \log \left( \frac{|P^+|}{P^+ + N^-} \right) \right) - \left( \log \left( \frac{|P|}{|P| + |N|} \right) \right) \right) \quad (3.1)$$

The FOIL algorithm seeks the item that yields largest positive gain for a particular class in training dataset. Selecting a single item will generate a lower number of rules and may ignore close FOIL gain values. Hence, accuracy and efficiency of the rule generation was improved by assigning weights to the data objects. Once the data objects associated with the item are covered by the rule, the weight decreases by multiplying a factor. This weighted approach extracts more items and generates more rules simultaneously. The algorithm uses PNArray data structure to store the number of positive and negative objects before and after appending item  $p$  to rule  $r$  [285]. It is utilized to reduce storage space and computational time. The accuracy of rules generated is evaluated using the Laplace expected error estimate. Expected accuracy is calculated for each rule before the test objects are classified. The expected accuracy of a rule  $r$  is given by:

$$Laplace\ Accuracy(r) = \frac{P_c(r) + 1}{P_{tot}(r) + m} \quad (3.2)$$

where  $m$  is the number of class labels,  $P_{tot}(r)$  is the total number of objects in training dataset that satisfies the antecedent of  $r$ , and  $P_c(r)$  is the number of objects are covered by rule  $r$  that belongs to class  $c$ . The case and control class rules are ranked based on highest to lowest Laplace accuracy estimation.

### 3.1.5 Classifier

The rules generated for training dataset  $D$  are organized in the order to form the classifier [283]. The size of the rule set is reduced in the rule pruning phase to improve efficiency and accuracy. Laplace expected accuracy is calculated for each rule. The rules with the highest expected accuracy are selected. The best  $k$  rules generated for each class in a rule set are used for the prediction. To classify the test object  $t$  using the classifier  $R$ , the algorithm selects all the rules whose antecedent satisfies  $t$ . The best  $k$  rules are selected for each class from the generated rules. Finally, it compares the average expected accuracy of the best  $k$  rules of each class, and chooses the class with the highest expected accuracy as the predicted class. The accuracy of the classification in this chapter is evaluated using  $F_1$ -score ( $F$ -score or  $F$ - measure) [280].

$$F_1 = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (3)$$

$$Precision = \frac{TP}{TP + FP} \quad (4)$$

$$Recall = \frac{TP}{TP + FN} \quad (5)$$

where,  $TP$ ,  $FP$ , and  $FN$  are true positive, false positive and false negative rates respectively. True positive (TP) rate is the proportion of positive cases that are correctly classified. False positive (FP) rate is the proportion of negative cases that are incorrectly classified as positive. False negative (FN) rate is the proportion of positive cases that are incorrectly classified as negative.

## 3.2 Data Simulation

A number of studies have been published on the basis of simulated data [36, 37, 108, 182, 291] to identify the interacting genes related to the disease. Hence, in this section, a simulation based study is performed. The  $n$  locus interaction models are generated from publicly available tool GAMETES [221]. The tool generates randomly pure and strict n-locus disease models with specified heritability, minor allele frequency and population size. It generates the datasets from these models. The goal of this simulated study is to detect interactions between multi-locus SNPs using the AC approach. Two

simulated scenarios are considered to evaluate the accuracy of AC with the previous approaches in the absence of main effect.

### 3.2.1 Scenario I

In the first scenario as reported in Ritchie [36], six two-locus epistasis (gene-gene interactions) models with different penetrance values are simulated for 20 SNPs with two functional SNPs (P1 and P2), and 18 independent non-functional SNPs. Case-control datasets are simulated with 200 cases and 200 controls in accordance to Hardy-Weinberg proportions using GAMETES tool [221]. Table 3.1 represents the overview of model dependent allele frequencies along with their penetrance tables. A simple model of two alleles ( $p$  and  $q$ ) necessarily sums to unity. That is,  $p + q = 1$  where  $p$  is minor allele frequency and  $q$  is the alternative allele frequency. Model 1 is based on nonlinear XOR function described by [292, 293] in which all high risk genotype combinations (AaBB, Aabb, AABb and aaBb) have a penetrance value of 0.1. Model 2 is described by [293, 294] in which high risk genotype combinations (AAAb, AaBb and aaBB) have penetrance values 0.1, 0.05 and 0.1 respectively. Other four models are described by [293] with Minor Allele Frequencies (MAFs) of 0.25, 0.25, 0.1 and 0.1 respectively. Cases and controls of 1:1, 1:2, 1:4, and 1:6 ratios are generated for 400 samples. 100 datasets are simulated for each model in order to evaluate the power of AC by estimating the number of times the approach successfully identified two functional SNPs. In total, 2,400 datasets are generated and analysed [38].

**Table 3.1:** Epistasis models exhibiting interactions between two SNPs in the absence of main effects. Case-control datasets are generated using various penetrance functions and allele frequencies  $p$  and  $q$  (as reported in [36]) updated with heritability using GAMETES tool based on [221]: © 2014 IEEE.

#### Model 1

$p=0.5, q=0.5, H=0.053$

	BB	Bb	bb
AA	0	0.1	0
Aa	0.1	0	0.1
aa	0	0.1	0

#### Model 4

$p=0.25, q=0.75, H=0.033$

	BB	Bb	bb
AA	0	0.01	0.09
Aa	0.4	0.01	0.08

#### Model 2

$p=0.5, q=0.5, H=0.051$

	BB	Bb	bb
AA	0	0	0.1
Aa	0	0.05	0
aa	0.1	0	0

#### Model 5

$p=0.1, q=0.9, H=0.02$

	BB	Bb	bb
AA	0.07	0.05	0.02
Aa	0.05	0.09	0.01

#### Model 3

$p=0.25, q=0.75, H=0.016$

	BB	Bb	bb
AA	0.08	0.07	0.05
Aa	0.1	0	0.1
aa	0.03	0.1	0.04

#### Model 6

$p=0.1, q=0.9, H=0.015$

	BB	Bb	bb
AA	0.09	0.001	0.02
Aa	0.08	0.07	0.005

aa	0.07	0.09	0.03	aa	0.02	0.01	0.03	aa	0.003	0.007	0.02
----	------	------	------	----	------	------	------	----	-------	-------	------

### 3.2.2 Scenario II

In the second scenario, datasets are replicated as in the simulated study performed by Velez, D.R., [108]. Datasets are generated using GAMETES tool, which generates pure, and strict  $n$ -locus disease models [221]. Various datasets with  $i$  SNPs are generated for one to six loci models by varying heritability, minor allele frequency, and population size. There are  $n$  functional and  $i - n$  non-functional SNPs generated for  $n$ -locus models with  $i$  number of SNPs. The datasets are generated for single-locus to six-locus models with 20 SNPs. In the single-locus, one SNP is functional and 19 SNPs are non-functional. In two loci among 20 SNPs, two SNPs are functional and 18 SNPs are non-functional. Similarly, in three, four, five & six loci models, three, four, five & six SNPs are functional and 17, 16, 15 & 14 SNPs are non-functional respectively. Each genetic model is distributed across seven heritability (0.01, 0.025, 0.05, 0.1, 0.2, 0.3 and 0.4) and two different minor allele frequencies (0.2 and 0.4). Where heritability is the proportion of observable difference between individuals due to genetic differences and minor allele frequency is the frequency at which less common allele occurs in a given population. Five models for each 14 heritability-allele frequency combinations are generated to develop 70 epistasis models in accordance to Hardy-Weinberg proportions. Penetrance is the proportion of a genotype combination that expresses the probability of a disease. The penetrance tables of 70 epistasis models in the absence of main effect are available in Table 3.2. For each model, the datasets are simulated with different case-control ratios (1:1, 1:2, and 1:4), and sample size (400, 800 and 1600). One hundred datasets are generated for each model by generating 12,600 datasets for each locus. In total, 75,600 datasets has to be generated for one to six-locus models.

However, only 54,900 datasets were generated due to limited ability of GAMETES to generate models with higher heritability [221]. Extremely low probability penetrance tables are generated for certain values of heritability and prevalence. For example,  $n$ -locus penetrance tables are generated for all heritability  $\leq 1$  having prevalence and MAF equal to 0.5 [221]. However, in 5 and 6 locus, models cannot be generated for heritability  $\geq 0.1$  after 100,000 iterations. In addition, models cannot be generated for

heritability = 1 and prevalence =0.25. The limitations are more severe if MAFs are specified [221]. Hence, a few models could not be generated in simulated scenarios of three loci, four loci and six loci models. In spite of these limitations, the GAMETES tool was still used in this research to simulate genetic models, as it generates pure and strictly epistatic models which constitute worst case to detect the associations of the disease [108, 221]. Further, it considers complex multi-locus effect in generating the disease model.

**Table 3.2:** Penetrance table for 70 epistasis models generated in simulated scenario II.

<b>Model 1</b>	<b>Model 2</b>	<b>Model 3</b>
MAF = 0.2, H = 0.01, K=0.117	MAF = 0.2, H = 0.01, K=0.623	MAF = 0.2, H = 0.01, K=0.096
AA      Aa      aa	AA      Aa      aa	AA      Aa      aa
BB      0.117    0.12    0.095	BB      0.598    0.67    0.661	BB      0.086    0.12    0.062
Bb      0.117    0.123   0.068	Bb      0.664    0.553   0.535	Bb      0.119    0.048   0.107
bb      0.119    0.02    0.851	bb      0.71      0.437   0.736	bb      0.069    0.094   0.544
<b>Model 4</b>	<b>Model 5</b>	<b>Model 6</b>
MAF=0.2, H=0.01, K=0.07	MAF=0.2, H=0.01, K=0.942	MAF=0.2, H=0.025, K=0.065
AA      Aa      aa	AA      Aa      aa	AA      Aa      aa
BB      0.092    0.056   0.062	BB      0.931    0.961   0.961	BB      0.066    0.065   0.039
Bb      0.058    0.124   0.068	Bb      0.961    0.902   0.942	Bb      0.067    0.069   3.706
bb      0.048    0.097   0.444	bb      0.959    0.947   0.622	bb      0.022    0.034   0.995
<b>Model 7</b>	<b>Model 8</b>	<b>Model 9</b>
MAF = 0.2, H = 0.025, K=0.611	MAF = 0.2, H = 0.025, K=0.046	MAF = 0.2, H = 0.025, K=0.089
AA      Aa      aa	AA      Aa      aa	AA      Aa      aa
BB      0.644    0.552   0.57	BB      0.056    0.03    0.017	BB      0.11      0.048   0.081
Bb      0.531    0.759   0.705	Bb      0.03      0.081   0.019	Bb      0.051    0.171   0.036
bb      0.733    0.379   0.529	bb      0.013    0.026   0.733	bb      0.057    0.085   0.627
<b>Model 10</b>	<b>Model 11</b>	<b>Model 12</b>
MAF = 0.2, H = 0.025, K=0.936	MAF = 0.2, H = 0.05, K=0.677	MAF = 0.2, H = 0.05, K=0.593
AA      Aa      aa	AA      Aa      aa	AA      Aa      aa
BB      0.919    0.969   0.952	BB      0.663    0.676   0.927	BB      0.544    0.703   0.515
Bb      0.97      0.867   0.967	Bb      0.676    0.747   0.153	Bb      0.7       0.361   0.753
bb      0.951    0.969   0.449	bb      0.928    0.152   0.883	bb      0.538    0.708   0.573
<b>Model 13</b>	<b>Model 14</b>	<b>Model 15</b>
MAF = 0.2, H = 0.05, K=0.168	MAF = 0.2, H = 0.5, K=0.098	MAF = 0.2, H = 0.5, K=0.052
AA      Aa      aa	AA      Aa      aa	AA      Aa      aa
BB      0.211    0.084   0.149	BB      0.132    0.036   0.048	BB      0.075    0.012   0.015
Bb      0.083    0.34    0.153	Bb      0.033    0.224   0.125	Bb      0.012    0.136   0.038
bb      0.16      0.132   0.588	bb      0.071    0.078   0.694	bb      0.015    0.039   0.778
<b>Model 16</b>	<b>Model 17</b>	<b>Model 18</b>
MAF = 0.2, H = 0.5, K=0.052	MAF = 0.2, H = 0.1, K=0.364	MAF = 0.2, H = 0.1, K=0.178
AA      Aa      aa	AA      Aa      aa	AA      Aa      aa

BB	0.075	0.012	0.015
Bb	0.012	0.136	0.038
bb	0.015	0.039	0.778

### Model 19

MAF = 0.2, H = 0.1, K=0.118

	AA	Aa	aa
BB	0.172	0.016	0.091
Bb	0.021	0.318	0.079
bb	0.047	0.166	0.881

### Model 22

MAF = 0.2, H = 0.2, K=0.070

	AA	Aa	aa
BB	0.609	0.249	0.424
Bb	0.265	0.907	0.661
bb	0.295	0.92	0.076

### Model 25

MAF = 0.2, H = 0.2, K=0.174

	AA	Aa	aa
BB	0.264	0.003	0.117
Bb	0.001	0.519	0.186
bb	0.13	0.16	0.994

### Model 28

MAF = 0.2, H = 0.3, K=0.662

	AA	Aa	aa
BB	0.528	0.928	0.696
Bb	0.937	0.125	0.585
bb	0.628	0.721	0.759

### Model 31

MAF = 0.2, H = 0.4, K=0.604

	AA	Aa	aa
BB	0.448	0.88	0.902
Bb	0.942	0.004	0.01
bb	0.409	0.997	0.596

### Model 34

MAF = 0.2, H = 0.4, K=0.664

	AA	Aa	aa
BB	0.509	0.972	0.696
Bb	0.978	0.043	0.621
bb	0.643	0.727	0.519

### Model 37

MAF = 0.4, H = 0.009, K=0.429

	AA	Aa	aa
BB	0.4	0.456	0.416
Bb	0.404	0.445	0.439
bb	0.571	0.323	0.43

BB	0.44	0.204	0.459
Bb	0.217	0.684	0.182
bb	0.352	0.398	0.317

### Model 20

MAF = 0.2, H = 0.1, K=0.921

	AA	Aa	aa
BB	0.878	1	0.984
Bb	0.999	0.768	0.911
bb	0.992	0.896	0.003

### Model 21

MAF = 0.2, H = 0.2, K=0.470

	AA	Aa	aa
BB	0.389	0.667	0.198
Bb	0.666	0.016	0.982
bb	0.209	0.961	0.735

### Model 22

MAF = 0.2, H = 0.2, K=0.070

	AA	Aa	aa
BB	0.602	0.922	0.743
Bb	0.918	0.295	0.71
bb	0.775	0.646	0.191

### Model 23

MAF = 0.2, H = 0.2, K=0.071

	AA	Aa	aa
BB	0.311	0.028	0.033
Bb	0.025	0.545	0.475
bb	0.059	0.422	0.906

### Model 24

MAF = 0.2, H = 0.2, K=0.209

	AA	Aa	aa
BB	0.541	0.161	0.587
Bb	0.156	0.994	0.077
bb	0.623	0.005	0.516

### Model 27

MAF = 0.2, H = 0.3, K=0.658

	AA	Aa	aa
BB	0.513	0.916	0.926
Bb	0.908	0.224	0.147
bb	0.986	0.027	0.477

### Model 30

MAF = 0.2, H = 0.3, K=0.752

	AA	Aa	aa
BB	0.623	0.999	0.844
Bb	0.992	0.284	0.662
bb	0.903	0.544	0.003

### Model 33

MAF = 0.2, H = 0.4, K=0.656

	AA	Aa	aa
BB	0.505	0.946	0.761
Bb	0.98	0.038	0.429
bb	0.488	0.977	0.796

### Model 36

MAF = 0.4, H = 0.009, K=0.670

	AA	Aa	aa
BB	0.648	0.653	0.777
Bb	0.67	0.673	0.663
bb	0.723	0.703	0.455

### Model 39

MAF = 0.4, H = 0.009, K=0.398

	AA	Aa	aa
BB	0.352	0.449	0.351
Bb	0.456	0.355	0.4
bb	0.331	0.415	0.501

<b>Model 40</b>				<b>Model 41</b>				<b>Model 42</b>			
MAF = 0.4, H = 0.009, K=0.227				MAF = 0.4, H = 0.025, K=0.343				MAF = 0.4, H = 0.025, K=0.370			
	AA	Aa	aa		AA	Aa	aa		AA	Aa	aa
BB	0.283	0.194	0.2	BB	0.387	0.369	0.168	BB	0.307	0.425	0.351
Bb	0.189	0.263	0.205	Bb	0.335	0.343	0.363	Bb	0.344	0.374	0.421
bb	0.215	0.193	0.357	bb	0.269	0.287	0.679	bb	0.593	0.24	0.263
<b>Model 43</b>				<b>Model 44</b>				<b>Model 45</b>			
MAF = 0.4, H = 0.025, K=0.514				MAF = 0.4, H = 0.025, K=0.471				MAF = 0.4, H = 0.025, K=0.780			
	AA	Aa	aa		AA	Aa	aa		AA	Aa	aa
BB	0.633	0.472	0.378	BB	0.356	0.574	0.423	BB	0.698	0.832	0.808
Bb	0.424	0.54	0.643	Bb	0.55	0.415	0.464	Bb	0.837	0.721	0.828
bb	0.52	0.537	0.436	bb	0.496	0.41	0.601	bb	0.794	0.839	0.573
<b>Model 46</b>				<b>Model 47</b>				<b>Model 48</b>			
MAF = 0.4, H = 0.05, K=0.307				MAF = 0.4, H = 0.05, K=0.634				MAF = 0.4, H = 0.05, K=0.395			
	AA	Aa	aa		AA	Aa	aa		AA	Aa	aa
BB	0.358	0.34	0.092	BB	0.726	0.541	0.711	BB	0.41	0.29	0.678
Bb	0.315	0.301	0.307	Bb	0.607	0.633	0.703	Bb	0.357	0.486	0.211
bb	0.168	0.25	0.793	bb	0.513	0.852	0.257	bb	0.478	0.361	0.314
<b>Model 49</b>				<b>Model 50</b>				<b>Model 51</b>			
MAF = 0.4, H = 0.05, K=0.746				MAF = 0.4, H = 0.05, K=0.755				MAF = 0.4, H = 0.1, K=0.675			
	AA	Aa	aa		AA	Aa	aa		AA	Aa	aa
BB	0.616	0.831	0.783	BB	0.615	0.844	0.807	BB	0.583	0.637	0.999
Bb	0.806	0.674	0.829	Bb	0.839	0.675	0.808	Bb	0.676	0.681	0.657
bb	0.859	0.772	0.414	bb	0.821	0.797	0.483	bb	0.882	0.745	0.001
<b>Model 52</b>				<b>Model 53</b>				<b>Model 54</b>			
MAF = 0.4, H = 0.1, K=0.566				MAF = 0.4, H = 0.1, K=0.558				MAF = 0.4, H = 0.1, K=0.585			
	AA	Aa	aa		AA	Aa	aa		AA	Aa	aa
BB	0.718	0.597	0.135	BB	0.527	0.704	0.191	BB	0.319	0.722	0.774
Bb	0.476	0.525	0.897	Bb	0.617	0.417	0.849	Bb	0.774	0.491	0.445
bb	0.499	0.624	0.546	bb	0.452	0.653	0.512	bb	0.62	0.56	0.584
<b>Model 55</b>				<b>Model 56</b>				<b>Model 57</b>			
MAF = 0.4, H = 0.1, K=0.757				MAF = 0.4, H = 0.2, K=0.432				MAF = 0.4, H = 0.2, K=0.403			
	AA	Aa	aa		AA	Aa	aa		AA	Aa	aa
BB	0.573	0.876	0.817	BB	0.192	0.438	0.956	BB	0.108	0.543	0.653
Bb	0.882	0.636	0.842	Bb	0.46	0.498	0.172	Bb	0.589	0.388	0.037
bb	0.801	0.855	0.369	bb	0.888	0.223	0.033	bb	0.515	0.14	0.945
<b>Model 58</b>				<b>Model 59</b>				<b>Model 60</b>			
MAF = 0.4, H = 0.2, K=0.484				MAF = 0.4, H = 0.2, K=0.611				MAF = 0.4, H = 0.2, K=0.828			
	AA	Aa	aa		AA	Aa	aa		AA	Aa	aa
BB	0.645	0.321	0.616	BB	0.88	0.39	0.674	BB	0.622	0.944	0.945
Bb	0.212	0.674	0.529	Bb	0.332	0.79	0.706	Bb	0.948	0.698	0.952
bb	0.941	0.285	0.055	bb	0.847	0.576	0.189	bb	0.935	0.96	0.195
<b>Model 61</b>				<b>Model 62</b>				<b>Model 63</b>			
MAF = 0.4, H = 0.3, K=0.434				MAF = 0.4, H = 0.3, K=0.496				MAF = 0.4, H = 0.3, K=0.460			
	AA	Aa	aa		AA	Aa	aa		AA	Aa	aa

BB	0.173	0.444	0.998
Bb	0.457	0.563	0.002
bb	0.958	0.032	0.468

**Model 64**

MAF = 0.4, H = 0.3, K=0.482

	AA	Aa	aa
BB	0.144	0.693	0.613
Bb	0.846	0.227	0.434
bb	0.156	0.777	0.336

**Model 67**

MAF = 0.4, H = 0.4, K=0.423

	AA	Aa	aa
BB	0.077	0.817	0.022
Bb	0.5	0.244	0.787
bb	0.971	0.075	0.235

**Model 70**

MAF = 0.4, H = 0.4, K=0.729

	AA	Aa	aa
BB	0.361	0.996	0.758
Bb	0.992	0.459	0.947
bb	0.767	0.94	0.012

BB	0.953	0.3	0.059
Bb	0.304	0.558	0.746
bb	0.046	0.756	0.733

**Model 65**

MAF = 0.4, H = 0.3, K=0.743

	AA	Aa	aa
BB	0.439	0.948	0.818
Bb	0.971	0.518	0.908
bb	0.748	0.961	0.083

**Model 68**

MAF = 0.4, H = 0.4, K=0.438

	AA	Aa	aa
BB	0.984	0.17	0.012
Bb	0.092	0.625	0.657
bb	0.247	0.481	0.739

**Model 66**

MAF = 0.4, H = 0.4, K=0.493

	AA	Aa	aa
BB	0.026	0.72	0.865
Bb	0.709	0.478	0.054
bb	0.899	0.029	0.974

**Model 69**

MAF = 0.4, H = 0.4, K=0.575

	AA	Aa	aa
BB	0.047	0.887	0.829
Bb	0.965	0.347	0.384
bb	0.595	0.56	0.58

### 3.3 Real data

Breast cancer is a complex disease that occurs due to various unknown etiological aberrations. It is classified as either hereditary or sporadic. Breast cancer caused by inheriting faulty or mutated genes is known as hereditary or familial breast cancer. The rest is categorized as sporadic breast cancer. More than 80% of breast cancers are sporadic whose risk factors may depend on various independent and interacting factors. The data comprise of 410 samples obtained according to the requirements of the Institutional Review Board of Vanderbilt University Medical School [17]. The study is based on 207 white women with sporadic primary invasive breast cancer patients and 204 controls were treated at Vanderbilt University Medical Centre. The DNA of all the samples was isolated by using a DNA extraction kit (Gentra) [17]. The samples were used to amplify the desired gene segments using polymerase chain reaction (PCR) and were then analysed. The study considers the genetic variants in five genes (COMT, CYP1A1, CYP1B1, GSTM1 and GSTT1), which may affect the metabolism of estrogens that could increase the risk of sporadic breast cancer [17]. Hence, the analysis focused

on the genes COMT (Catechol-O-methyl transferase) on chromosome 22q11.2, CYP1A1 (Cytochrome P450 1 A1 enzyme) on chromosome 15q22-q24, CYP1B1 (Cytochrome P450 1 B1 enzyme) on chromosome 2p21-22, GSTM1 (Glutathione S-transferase Mu 1) on chromosome 1p13.3 and GSTT1 (Glutathione S-transferase theta 1) on chromosome 22q11.2. The polymorphisms in these genes are summarized and reported in the research [17]. The dataset considered 10 SNPs (Cyp1A1m1, Cyp1A1m2, Cyp1A1m4, Cyp1B1-48, Cyp1B1-119, Cyp1B1-432, Cyp1B1-453, COMT, GSTM1, and GSTT1) in five genes for the analysis. Since DNA is duplicated in each cell, three genotypes (common homozygous, heterozygous and rare homozygous) are yield and are numerically represented as zero for AA, one for Aa, and two for aa. There are 19 missing values and these are represented numerically by three. Cases and controls are assigned 1 and 0 in the class attribute.

## 3.4 Data Analysis

Case-control based simulated datasets are generated for both balanced and imbalanced data. The number of cases and controls are equal in balanced data and are not equal in imbalanced data. Balanced datasets are analysed using threshold value (ratio of number of cases to controls) equal to one. Imbalanced datasets are analysed by adjusting threshold (T) values. That is, the T value for 1:2 ratio is 0.5 and 1:4 ratio is 0.25. In this chapter, several experiments are conducted over the proposed approach and compared the results with MDR. The datasets for both simulated scenarios are analysed using the latest MDR software tool available from [www.epistasis.org](http://www.epistasis.org). The data is exhaustively evaluated for single-locus model to six-locus models to identify all possible interactions between SNPs. Balanced accuracy of each model is estimated for both training and testing data. Finally, a best model with high testing accuracy and high cross validation consistency is selected. The power of MDR has been estimated by the number of times the functional SNPs are identified in 100 datasets of each model. The final results are statistically evaluated with a 1000 fold permutation test and whose p-values are compared with 0.05 in determining the significance of the findings.

Further, the datasets for both simulated scenarios are analysed using the associative classifier. The configuration parameters of AC, the support, confidence, and length of antecedent of a rule, are set to 65%, 80%, and 9 respectively. The accuracy of AC

algorithm is analysed for single-locus to six-locus models using Weka tool. Weka tool is an open source machine learning software tool developed in Java [295]. Ten-fold cross validation is performed to reduce the possibility of biased estimation due to the division of data. This process is repeated for each combination of  $n$  locus. Hence, for each combination of  $n$  loci, the algorithm runs  $m$  times. Finally, the hypothesis test has been performed over the best model chosen from both analyses to evaluate its statistical significance. Among the many methods and software implementations that have been used to investigate the interactions between SNPs, the other most prominent approaches for identifying genetic effects in the presence of interactions are RF, SVM and NN. Further, Naïve Bayes algorithm is also considered in this chapter, as it is a well-established machine learning method and has been successfully applied in analysing GWAS data. Both simulated scenarios are analysed using RF, SVM, NN and Naïve Bayes algorithms in Weka tool, and the prediction accuracy is compared with AC. Further, the prediction accuracy of AC is compared with unsupervised learning algorithms, such as, KMeans, and principal component analysis (PCA).

Once, the approach is analysed over two simulated scenarios, it is further evaluated on a real dataset. The sporadic breast cancer data [17] is analysed using a threshold of case / control ratio ( $\frac{207}{204}=1.0147$ ). An exhaustive search was performed for all possible two to nine-locus models. The ten-fold cross validation is performed to minimize the statistical variance. The final results are statistically evaluated with a 1000 fold permutation test and whose p-values are compared with 0.05 in determining the significance of the findings.

## 3.5 Evaluation of the Approach

Several experiments were performed over simulated datasets and sporadic breast cancer data to evaluate the accuracy of AC over MDR, RF, SVM, NN, Naïve Bayes, KMeans and PCA. The goal of this study is to determine whether AC is a better approach for identifying the higher order SNP interactions in the absence of main effect. The approach considers the ratio of cases and controls for each SNP combination at different loci. It generates statistically significant genotype combinatorial associations in terms of rules based on cases and controls. Predicting class labels of test objects from these rules retains higher accuracy in genetic combinations that contribute to a disease. Despite the

increase in the accuracy, the approach reduced the false positive error by permutation testing under the null hypothesis. The results have been obtained by applying the approach over two simulated scenarios. Further, they are applied over sporadic breast cancer data to identify complex interactions associated with the disease.

### **3.5.1 Preliminary results for two-locus interactions on balanced and imbalanced simulated datasets**

The goal of this preliminary study is to determine whether AC is a better approach for identifying the higher order SNP interactions in the absence of main effect. The approach considers the ratio of cases and controls for each SNP combination at different locus. It generates statistically significant genotype combinatorial associations in terms of rules based on cases and controls. Predicting class labels of test objects from these rules retains higher accuracy in genetic combinations that contribute to a disease. Despite the increase in accuracy, the approach will still reduce the false positive error by using permutation testing under the null hypothesis. The results have been obtained on two simulated scenarios to identify complex associations between genotype and phenotype.

#### **3.5.1.1 Scenario I**

The accuracy of AC for all six models in Scenario-I is presented in Table 3.3 to Table 3.6. Table 3.3 represents the accuracy of AC along with other previous approaches with 400 samples of 1:1 ratio of cases and controls. The accuracy of AC is higher for model 5 when compared to other approaches. Table 3.4 represents the results of AC and other current approaches for 1:2 ratios of cases and controls. The accuracy of AC is high for model 3, models 5 and model 6. Table 3.5 and Table 3.6 represent accuracy of AC for 1:4 and 1:6 ratios of cases and controls respectively. The results show that AC outperforms in all 6 models compared to the current approaches. The performance of AC is seen to be better with imbalanced data than with balanced data.

**Table 3.3:** Accuracy of 6 models with 1:1 ratio of cases and controls

<b>Model</b>	<b>MAF</b>	<b>MDR</b>	<b>RF</b>	<b>AC-CPAR</b>	<b>AC-CBA</b>	<b>SVM</b>	<b>NN</b>	<b>Naïve Bayes</b>
1	0.5	76	57.75	59.5	70	48	59.5	47.75
2	0.5	81.5	57.25	62.5	74	55.5	64.75	57

3	0.25	63	58.25	58.75	60	56	56.5	60.5
4	0.25	74.5	66	69.25	65.75	62	61.5	62
5	0.1	47.75	50.5	53.75	52	49.75	51.25	48.25
6	0.1	57	52	55	54.5	57.75	53.25	56.25

**Table 3.4:** Accuracy of 6 models with 1:2 ratio of cases and controls

Model	MAF	MDR	RF	AC-CPAR	AC-CBA	SVM	NN	Naïve Bayes
1	0.5	76.78	63.33	64.83	66.75	66.67	65.17	62.67
2	0.5	81.46	62.5	68.5	69.75	66.75	65.5	62
3	0.25	53.12	65.25	63	66.75	66.75	56.75	62.5
4	0.25	76.75	70.25	73	65.5	71.25	73.75	72.5
5	0.1	51.59	61.25	61.5	66.75	66.75	57	66
6	0.1	56.88	62.5	62.5	66.75	66.75	62.75	62

**Table 3.5:** Accuracy of 6 models with 1:4 ratio of cases and controls

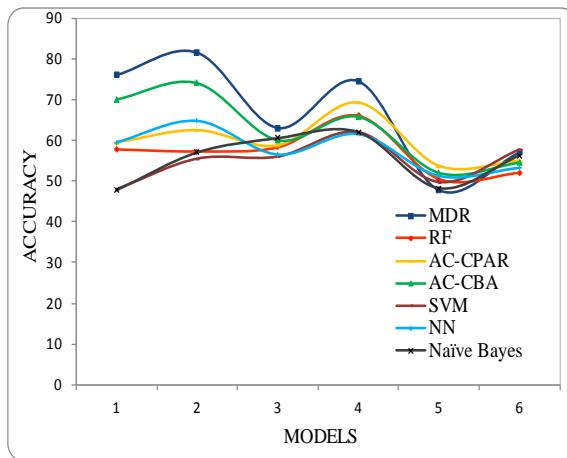
Model	MAF	MDR	RF	AC-CPAR	AC-CBA	SVM	NN	Naïve Bayes
1	0.5	76.25	78.6	76.6	80	79.25	73.9	79.8
2	0.5	82.66	79.75	79.5	80	79.5	72.75	79.25
3	0.25	53.28	78.25	78.25	69.75	80	66.75	77.75
4	0.25	76.09	79	78.25	80.25	72.25	78	79
5	0.1	57.66	76.5	77.5	80	67.5	68.75	78.25
6	0.1	54.84	75.5	78.5	80	68.5	75.5	79

**Table 3.6:** Accuracy of 6 models with 1:6 ratio of cases and controls

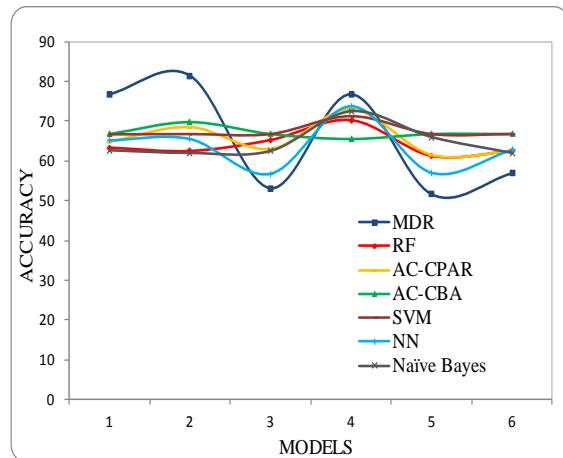
Model	MAF	MDR	RF	AC-CPAR	AC-CBA	SVM	NN	Naïve Bayes
1	0.5	76.16	91.5	91.75	91.25	82.25	86.75	91.25
2	0.5	82.7	91.25	91.75	91.75	85.25	84	90.25
3	0.25	45.94	91.5	91.5	91.75	87	85.75	91.5
4	0.25	71.8	91	91.25	91.25	87	85	90
5	0.1	43.47	90.5	90	93.4	83	85.75	91.5
6	0.1	44.02	91	91.25	91.75	86	87.75	90.25

In the first scenario, as stated in the preliminary results, the approach is validated for both balanced and imbalanced datasets. Figure 3.3 shows the accuracy of AC over MDR, RF, SVM, NN and Naïve Bayes classifiers in 1:1 ratios of cases and controls for 400 samples. On an average of 100 datasets for each model, MDR significantly performed well for 1 to 4 models. However, AC performed better than other algorithms when allele

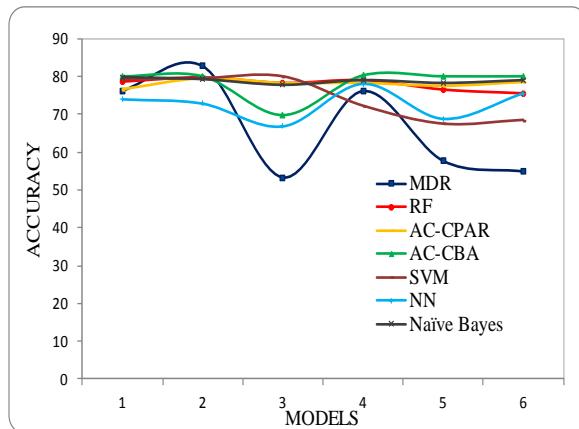
frequencies are 0.1 and 0.9. SVM performed equally as MDR with difference of less than 1% in accuracy for model 6. Figure 3.4 shows the accuracy of AC over other algorithms in 1:2 ratios of cases and controls in a sample size of 400. On average, AC achieves an improvement in accuracy of 13% compared to MDR for model 3, 5 0.1. It is observed that both for balanced and imbalanced data, AC is more accurate when the allele frequencies are 0.1 and 0.9. Figure 3.5 exhibits accuracy of samples with 1:4 ratios. On average, the accuracy of AC is about 12% higher than MDR. However, it is observed that the accuracy of AC is slightly reduced by about 2% in model 2 where allele frequencies are equal. Figure 3.6 shows that the accuracy of AC is much higher than MDR in 1:6 ratios of all 6 models. Accuracy of AC is about 50% higher than MDR when MAF values are 0.1 and 0.25.



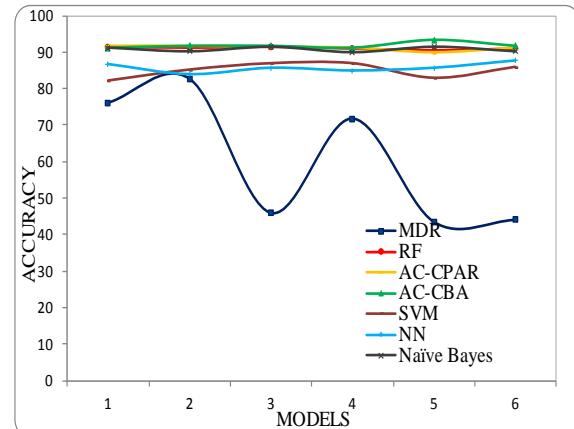
**Figure.3.3** Accuracy of 6 models with 1:1 ratio



**Figure.3.4** Accuracy of 6 models with 1:2 ratio



**Figure.3.5** Accuracy of 6 models with 1:4 ratio



**Figure.3.6** Accuracy of 6 models with 1:6 ratio

### 3.5.1.2 Scenario II

In scenario II, the accuracy of AC for 70 models is presented in Table 3.7 to Table 3.9 and compared with previous methods. Table 3.7 represents the results of AC and other current approaches for 1:1 ratio of 400 samples. The accuracy of AC is higher only in two cases (MAF=0.2, H=0.01 and MAF=0.4, H=0.025) when compared with existing approaches. Table 3.8 and Table 3.9 shows the results of 1:2 and 1:4 ratios of cases and controls respectively. The accuracy of AC is higher in majority of the models in 1:2 ratios. AC performed well when the heritability is  $\leq 0.1$  for both cases of MAF = 0.2 and 0.4. However, as shown in Table 3.9, AC outperformed other approaches in all 70 models for the ratio of 1:4. The results of scenario II confirm that the accuracy of AC is higher over other current approaches in imbalanced datasets.

**Table 3.7:** Accuracy of 70 models with 400 samples of 1:1 ratio of cases and controls

Model	MAF	Heritability	MDR	RF	AC-CPAR	AC-CBA	SVM	NN	Naïve Bayes
1	0.2	0.01	47	53	57	54.75	50.25	50.5	50.75
2	0.2	0.01	49.5	50.25	52.5	49.5	54	49	50.75
3	0.2	0.01	52.25	54.25	54.5	53	53.75	49.25	50.5
4	0.2	0.01	59.25	53.5	52.75	46.75	52.5	51.25	50.25
5	0.2	0.01	49.25	48.25	53.25	55	49	48.5	49.5
6	0.2	0.025	48.75	46.75	52.5	52.25	44	46.5	45.25
7	0.2	0.025	52	48.5	54.25	53.5	49	51.75	52.25
8	0.2	0.025	55.5	49.5	53.75	50.75	52.5	51.75	51.25
9	0.2	0.025	63.25	51.75	55.25	55.75	54.75	51.25	52
10	0.2	0.025	64.75	52.25	57	50.25	54.75	53.25	49.75
11	0.2	0.05	46.5	46.5	49.25	49.5	44.25	47.25	44.5
12	0.2	0.05	61.25	49	54	51.75	47.25	45.25	46.25
13	0.2	0.05	62	50.25	54.75	51.5	51.25	51.75	49.25
14	0.2	0.05	69	51.25	55	57.5	48.75	56	52.25
15	0.2	0.05	67.25	53.5	58.5	60.25	43.5	57.5	43.25
16	0.2	0.1	48.25	45.75	52	49	54	50.5	57.5
17	0.2	0.1	65.75	56.25	55.75	53.5	49.75	56.25	52.5
18	0.2	0.1	66	51	57.75	58	45	54	48.5
19	0.2	0.1	68	57.5	60	56.25	53.25	56	51.75
20	0.2	0.1	70.75	56.75	58.75	67.5	50.75	62.75	49.75
21	0.2	0.2	72	52.75	59.25	58	51	58.5	49.75
22	0.2	0.2	74	56	62	61	48.25	58.25	47.25
23	0.2	0.2	76	52.5	60.25	69.75	52.25	58.75	48
24	0.2	0.2	77.75	55.75	64.75	65.25	54.5	68	51

25	0.2	0.2	76.25	53.75	65.25	72.75	53	67	48.25
26	0.2	0.3	76	56.5	63.5	67.5	49.25	60.75	47.5
27	0.2	0.3	73	59.5	65	64.5	47.75	60.75	52.25
28	0.2	0.3	79.75	58.25	66.75	74.25	53.25	67	49.25
29	0.2	0.3	74.75	56.75	60.5	73.25	44.75	62	48.5
30	0.2	0.3	76.25	56.25	62	75.25	52.25	67.75	51
31	0.2	0.4	78.75	59	69.25	63	58	67	55.75
32	0.2	0.4	76.25	57.5	61.5	65.25	51.25	64.75	50
33	0.2	0.4	79.5	58	74.75	67.25	54	70.25	47.5
34	0.2	0.4	76.75	58.25	66.75	70.75	56.25	67.25	52.75
35	0.2	0.4	80.25	57	73.5	77.25	48.25	67	47.75
36	0.4	0.01	51.25	52.25	51.5	47.25	45.25	46	45
37	0.4	0.01	50.25	45.75	52.25	51.75	47.5	47.25	47.75
38	0.4	0.01	51.25	49	55.5	48.25	54.75	51.25	51.5
39	0.4	0.01	60.25	54.5	51.5	52.5	52.5	53.75	53.25
40	0.4	0.01	53.5	53	53.5	51	55.75	55.5	53
41	0.4	0.025	51.75	50.75	51.5	49.5	42.5	46.5	45.5
42	0.4	0.025	46.5	51.5	49	49.5	46	50.5	48
43	0.4	0.025	51.75	47	51.5	56	52.5	46	52.5
44	0.4	0.025	56.5	51	52.5	52.25	50.5	53.5	48.5
45	0.4	0.025	55.25	52.75	54.5	57.25	57.75	52	54.5
46	0.4	0.05	47.25	50.5	54.25	48.75	50	52.25	51.25
47	0.4	0.05	46.25	46.5	51	51	49.5	57.5	49.25
48	0.4	0.05	62.25	49.5	54	53	50.75	48.75	44.25
49	0.4	0.05	61.25	50.75	51.5	49.5	48.75	50.5	45.75
50	0.4	0.05	57	50.25	56	49.25	55.75	50.25	52.75
51	0.4	0.1	51.25	53	55.5	48.75	53.75	49.25	57.25
52	0.4	0.1	58	47.25	56.5	56.25	44.25	48	44.5
53	0.4	0.1	65.5	50.25	55.25	50.5	55.25	53.75	53
54	0.4	0.1	61.75	47.75	52.5	49.25	48	48	45.5
55	0.4	0.1	66.5	50.5	55.5	56.5	50.5	50.75	45.75
56	0.4	0.2	63.75	56.75	56.25	60.75	56.75	54.5	52.75
57	0.4	0.2	66.75	49.75	58.25	56.25	50.25	48.5	47.75
58	0.4	0.2	70	52.75	61.5	63	56	59.75	55.5
59	0.4	0.2	69.75	50.5	56	55.25	53	58.25	51.5
60	0.4	0.2	76.25	53.5	57.25	64.75	51.75	60	51.25
61	0.4	0.3	66.25	52.75	57.75	60	47	52	50
62	0.4	0.3	75	53.5	58.5	60.5	51	63.5	46.5
63	0.4	0.3	77.25	55	64.5	64.5	50.5	64	51.75
64	0.4	0.3	76.75	52.75	56	72.75	47.25	63.5	44.25
65	0.4	0.3	83.5	58.25	72.75	67.25	51	67	51.25
66	0.4	0.4	73.75	52.75	66	61	51.25	64	54.25
67	0.4	0.4	81.5	57	72	69.75	50.5	65.25	49.75
68	0.4	0.4	79	56	71	70.25	46.5	68.25	50.5

69	0.4	0.4	79.25	59.25	66.25	74.5	56.25	72.25	51.5
70	0.4	0.4	79.5	64	69	70	52	70	53

**Table 3.8:** Accuracy of 70 models with 400 samples of 1:2 ratio of cases and controls

Model	MAF	Heritability	MDR	RF	AC-CPAR	AC-CBA	SVM	NN	Naïve Bayes
1	0.2	0.01	54.63	61.75	59.5	65	54	56.25	62.75
2	0.2	0.01	47.09	62.75	59.75	66.75	55.75	57.75	62.25
3	0.2	0.01	50.84	61.75	59.75	66.75	52	54.25	59.25
4	0.2	0.01	51.05	62.25	59.75	65	60.25	57.75	63
5	0.2	0.01	49.71	61.75	61.25	62	56	53.75	63.75
6	0.2	0.025	49.14	60	58.75	64	55.25	52.25	62.5
7	0.2	0.025	58.9	60.25	61.5	66.75	59.25	61	65.25
8	0.2	0.025	61.37	60.25	57	62	58.75	61.5	62.75
9	0.2	0.025	60.62	61.5	65	59.25	58	65	63.55
10	0.2	0.025	65.45	70	69.5	66.75	57.5	70	63.25
11	0.2	0.05	54.98	62	58.5	66.75	62	56.5	65
12	0.2	0.05	59.1	59.5	61	66.75	54.25	55.25	62.75
13	0.2	0.05	60.63	64	58.5	61	58.75	58.75	62.75
14	0.2	0.05	64.33	63	63.75	66.75	57.75	58.25	63.25
15	0.2	0.05	68.7	59.25	58.75	66.75	51.5	57	61
16	0.2	0.1	61.74	60	58.75	66.75	50.5	52.25	62.5
17	0.2	0.1	62.31	60.25	61.5	66.75	55.25	61	62.25
18	0.2	0.1	64.33	62.5	61.25	65	57	61.5	62.75
19	0.2	0.1	71.89	67.25	67.25	66.75	62.5	63.75	64
20	0.2	0.1	72	67.25	71.5	66.75	66.25	68.75	62.25
21	0.2	0.2	69.59	65.5	67.75	59.25	57.75	62	63.75
22	0.2	0.2	69.43	63.75	64	66.75	58.75	63	62
23	0.2	0.2	69.94	61.25	62.25	63.75	54.75	61	60
24	0.2	0.2	77.46	69.75	71	64.75	66.5	69.25	62
25	0.2	0.2	71.25	65.5	67.75	66.75	63.5	63.5	62.75
26	0.2	0.3	77.88	62.75	65.75	66.75	65.75	66.25	62
27	0.2	0.3	76.53	66	73	67.75	58.5	71	65
28	0.2	0.3	75.44	64.5	74.5	66.75	57.5	71.25	61.5
29	0.2	0.3	78.08	62.25	68.75	65	56.25	67	66.75
30	0.2	0.3	78.83	61.25	70.5	64.75	56.5	66.25	63.5
31	0.2	0.4	79.01	65	71.5	67.5	65.5	71	62
32	0.2	0.4	78.07	67.5	75	67.75	61.5	69.75	61
33	0.2	0.4	77.7	67.25	76.25	66.75	70	73.75	62.5
34	0.2	0.4	80.32	68.5	72.5	67	61	75	61.75
35	0.2	0.4	73.32	63	67	66.75	67.25	68	58.25
36	0.4	0.01	49.54	61.5	56	61.5	55	54.75	58.5
37	0.4	0.01	53.45	61.5	59.5	66.75	49.5	54.25	63.5

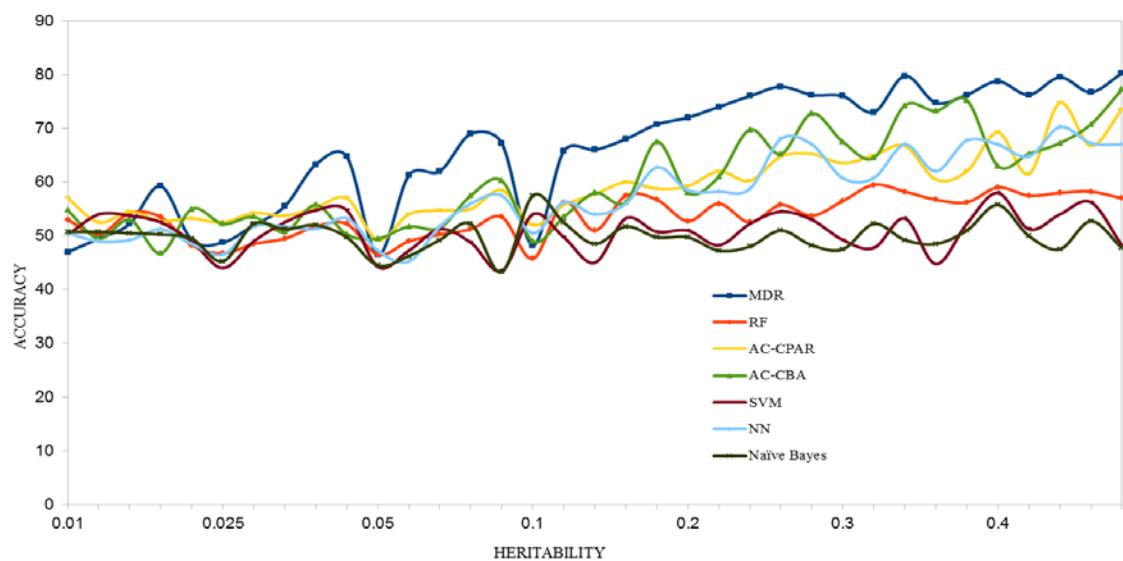
38	0.4	0.01	48.59	61.5	59	66.75	65	59	62.25
39	0.4	0.01	54.42	63.25	56	66.75	58.5	54.5	62.5
40	0.4	0.01	47.85	62.25	59.5	66.5	55.5	55	63.5
41	0.4	0.025	56.47	63	60	64	54.75	54.5	61.5
42	0.4	0.025	49.34	63.5	62.5	63.25	60.75	56.75	62.75
43	0.4	0.025	49.15	65.25	60	66.75	61	57.25	62.75
44	0.4	0.025	62.65	63.75	62.25	66.25	54.25	59.75	60.75
45	0.4	0.025	51.02	62.75	62.5	65.75	62.5	59.5	59.5
46	0.4	0.05	44.27	61.25	59.25	66.75	59.25	56	61.75
47	0.4	0.05	52.7	61.25	58.75	66.5	60.75	57.25	58.5
48	0.4	0.05	55.34	62.75	63.25	66.75	57.75	54.5	61.75
49	0.4	0.05	60.97	62.75	59	66.75	60	55.5	61.5
50	0.4	0.05	65.3	62.25	62.75	64.25	56.5	56.75	60.5
51	0.4	0.1	48.42	61	59	66.5	66.75	57.25	58.75
52	0.4	0.1	43.53	61.25	60	66.75	66.75	61.25	61
53	0.4	0.1	64.92	61.5	63	66.75	67.75	59	62.25
54	0.4	0.1	60.98	60.75	60.25	66.75	49.75	53.25	60.25
55	0.4	0.1	67.16	63.5	63.75	66.75	55.5	57.75	64
56	0.4	0.2	66.96	63.5	63	66.5	61	57.25	60.5
57	0.4	0.2	66.06	63.25	63.75	66.75	57.5	60.5	61.25
58	0.4	0.2	71.48	65.5	66.5	66.5	61.25	63.25	63.5
59	0.4	0.2	76.54	61	63.25	65.75	54	64.25	60.5
60	0.4	0.2	74.69	62.5	67.25	66.75	55.25	63.5	62.25
61	0.4	0.3	73.7	63.5	64.25	73	62.25	59.75	59.25
62	0.4	0.3	73.21	60	64.75	66.25	57.25	61.25	61.5
63	0.4	0.3	72.81	62	61.25	66.25	60.75	59.75	60.25
64	0.4	0.3	72.04	62.75	67.5	66.75	62.5	63.75	65.25
65	0.4	0.3	81.6	62.75	67.5	66.75	62.5	63.75	65.25
66	0.4	0.4	76.59	63.75	66	66.75	63.75	64.75	60.25
67	0.4	0.4	79.56	64.25	69	66.5	63.25	65.25	65.25
68	0.4	0.4	83.32	60.25	67.5	65.75	54.5	69.5	63.75
69	0.4	0.4	84.8	67	84.75	65	63.25	73.25	63
70	0.4	0.4	85.72	68.25	68.5	65.25	57	78.5	60.5

**Table 3.9:** Accuracy of 70 models with 400 samples of 1:4 ratio of cases and controls

Model	MAF	Heritability	MDR	RF	AC-CPAR	AC-CBA	SVM	NN	Naïve Bayes
1	0.2	0.01	44.69	77	72.5	80	64	69	77.75
2	0.2	0.01	52.81	76.25	71	80	66	70.25	77.5
3	0.2	0.01	40	77	75.25	80	68.25	69.75	79.5
4	0.2	0.01	50.78	77.5	71.5	80	68.5	68.5	78.5
5	0.2	0.01	50	78.25	72.5	80	66.25	67.25	78.5
6	0.2	0.025	48.59	80.25	79	74.5	71.5	70.5	78.5

7	0.2	0.025	63.13	78.25	80	80	68.25	69.75	76.75
8	0.2	0.025	56.09	77.25	79.25	79.5	64.75	72	78.5
9	0.2	0.025	62.5	78.5	78.75	80	64.5	70	79.5
10	0.2	0.025	52.5	77	80	80	67.25	72.25	78.5
11	0.2	0.05	56.09	78	79.5	80	69.75	69.75	77.5
12	0.2	0.05	48.29	78.75	79.25	80	67.75	71.25	78.75
13	0.2	0.05	60.16	78	77.5	80	68.5	76.25	79.75
14	0.2	0.05	65.78	79	79.75	76.75	71.25	70.5	78
15	0.2	0.05	62.81	78.25	79.75	79.25	71	71.75	79
16	0.2	0.1	51.56	76.5	72	80	67	68.5	78
17	0.2	0.1	64.22	78	77.25	80	72	68	78
18	0.2	0.1	70.16	78.75	77.5	80	71	74.75	78.5
19	0.2	0.1	68.59	79	77.25	80	71.5	75	77.5
20	0.2	0.1	70.94	78	79	78.75	67	74.5	77.75
21	0.2	0.2	70.16	79.25	78	80	67.5	72	78.5
22	0.2	0.2	75.63	79.5	77.25	80	72.25	74	78.75
23	0.2	0.2	73.75	77.25	80	80	71.5	76	77.5
24	0.2	0.2	74.06	78	79.75	78	71.75	78	77.5
25	0.2	0.2	75.47	79.75	81.75	78.25	72	78.25	78.5
26	0.2	0.3	74.37	79	80	80	68.5	76	77.5
27	0.2	0.3	75	78.75	78	79.5	70.5	80	78.25
28	0.2	0.3	76.72	80.25	84	80	71	79	77.5
29	0.2	0.3	73.75	78.75	78.25	78.75	72.25	75	77.75
30	0.2	0.3	75.63	79	78	80	68.5	70.75	78.5
31	0.2	0.4	81.09	79.5	80.25	80	72.25	78	79.25
32	0.2	0.4	80	78.25	80.25	80	67.5	81.75	77.5
33	0.2	0.4	78.12	78.5	79.25	77.25	75.5	82	78.25
34	0.2	0.4	83.28	79.5	86.75	80	70.25	78	79
35	0.2	0.4	84.22	79.5	89.25	80	75.5	86	79.25
36	0.4	0.01	45.63	76.75	79.25	79.75	68.75	72.75	78.25
37	0.4	0.01	56.25	77.5	80.25	79.75	69	69.25	79.75
38	0.4	0.01	47.66	78	79	80	66.75	68	78
39	0.4	0.01	53.28	77.5	79.75	80	63.5	66	76.75
40	0.4	0.01	50.47	77.5	80	80	66.25	68.75	78
41	0.4	0.025	54.06	77	79.25	80	71.5	71.5	78.75
42	0.4	0.025	44.84	78	79.5	80	68.25	69	77.75
43	0.4	0.025	57.03	79.25	78.75	80	64.75	71.25	75.75
44	0.4	0.025	46.88	76.75	79.75	80	64.5	68.75	79.25
45	0.4	0.025	44.37	78.5	80	80	67.25	68.5	78.75
46	0.4	0.05	59.69	78.25	78.75	80	69	69.25	78.25
47	0.4	0.05	45.78	78.5	79	79.5	69	74	79.25
48	0.4	0.05	49.38	78.25	78.5	80	65.5	70.75	79.25
49	0.4	0.05	43.28	78.75	79.5	80	65	67	79.5
50	0.4	0.05	57.34	77.5	78.75	72.5	64.75	70.5	78.75
51	0.4	0.1	62.34	77.75	79.75	72.25	70	69.25	77

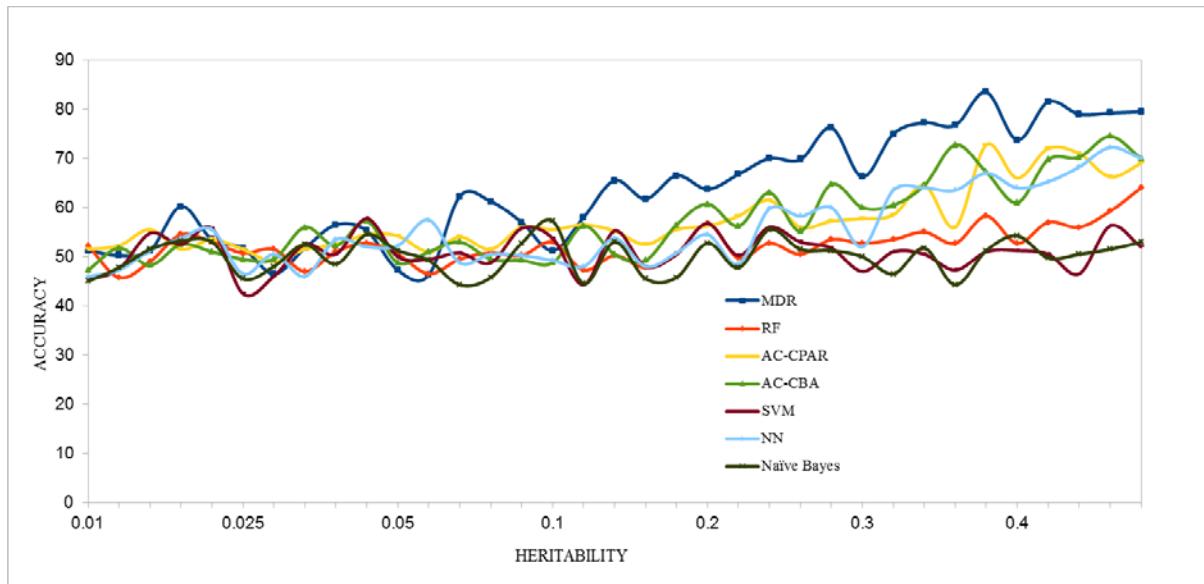
52	0.4	0.1	50.94	77.5	80.25	71.25	67	69.25	79.75
53	0.4	0.1	61.87	78.25	79.5	79.5	67.25	72.25	78.75
54	0.4	0.1	64.06	78.5	79.25	80	67.25	70.75	79
55	0.4	0.1	62.66	76.75	79	80	64.5	70.5	79.25
56	0.4	0.2	67.81	78	80	74.25	73	73.25	78.5
57	0.4	0.2	67.66	77.25	79.5	79.75	64.5	72.25	77.75
58	0.4	0.2	77.81	79.75	79.5	80	65	71.75	79.75
59	0.4	0.2	72.97	80	77.75	80.25	72.5	73.25	78
60	0.4	0.2	69.69	78.5	78.25	80	70.25	71.5	77.5
61	0.4	0.3	69.06	77.75	79	80	70.5	74.25	79
62	0.4	0.3	70	78	79	80	68.25	72.25	78.25
63	0.4	0.3	70.94	76.5	78.5	73.5	73	75.25	79
64	0.4	0.3	75.78	78.25	83	79.25	72.5	73.5	78.5
65	0.4	0.3	80.94	78	80	79.5	68.75	72.75	79.75
66	0.4	0.4	75.31	78.5	78	80	67.75	70.5	79
67	0.4	0.4	78.12	79.5	80	80.25	68.75	74.5	78.5
68	0.4	0.4	81.09	79.25	79.75	79.75	68.5	72.75	79.5
69	0.4	0.4	82.5	79	83.25	79.5	71.25	80.75	79.5
70	0.4	0.4	76.09	78.75	79.25	80.25	74	76.25	77.25



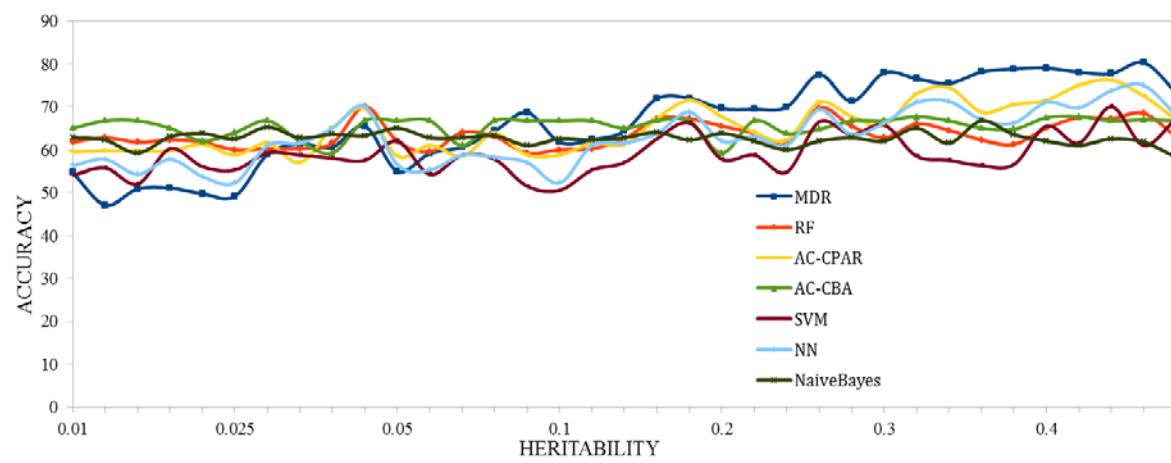
**Figure.3.7** Accuracy of 70 models with ratio 1:1 for MAF 0.2

The results of second scenario of simulations, demonstrated that the AC performed well across a wide range of SNP-SNP interaction models. Figure 3.7 illustrates accuracy of AC over other approaches in balanced data of 400 samples. MDR predominantly outperformed AC and other approaches. However, AC is more accurate than other

approaches up to 10% for allele frequencies 0.2 and 0.8 with heritability of 0.01. Further experiments were performed to observe the performance of AC when there is no genetic influence over the phenotype. It performed significantly better than all other approaches including MDR. Figure 3.8 illustrates the accuracy along y-axis and heritability along x-axis for 1:1 ratio with MAF equal to 0.4. MDR performed significantly better in balanced data compared to other methods. However, accuracy of AC improved up to 4% when heritability is 0.025. It also significantly performed better than other approaches when there is no genetic influence over the phenotype.

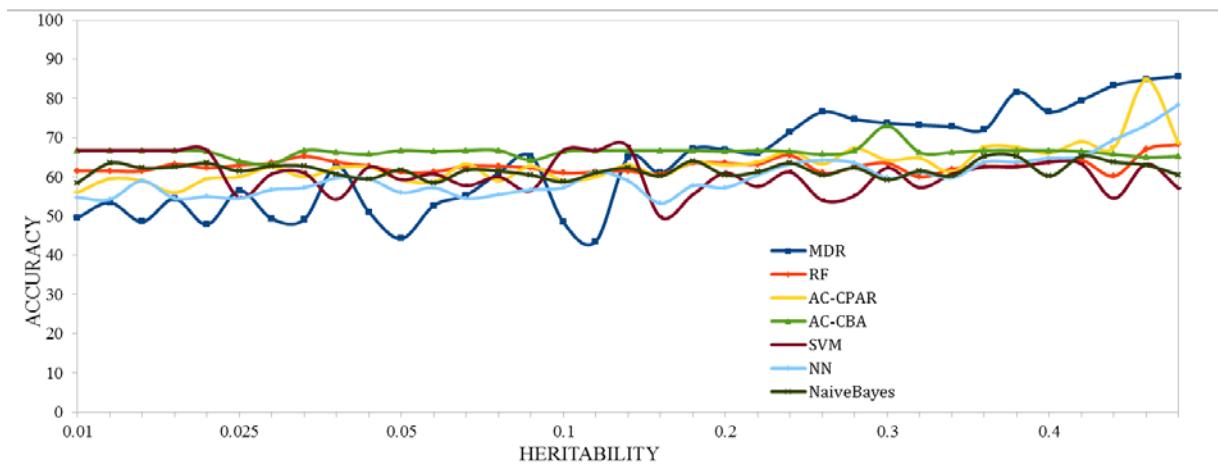


**Figure.3.8** Accuracy of 70 models with ratio 1:1 for MAF 0.4

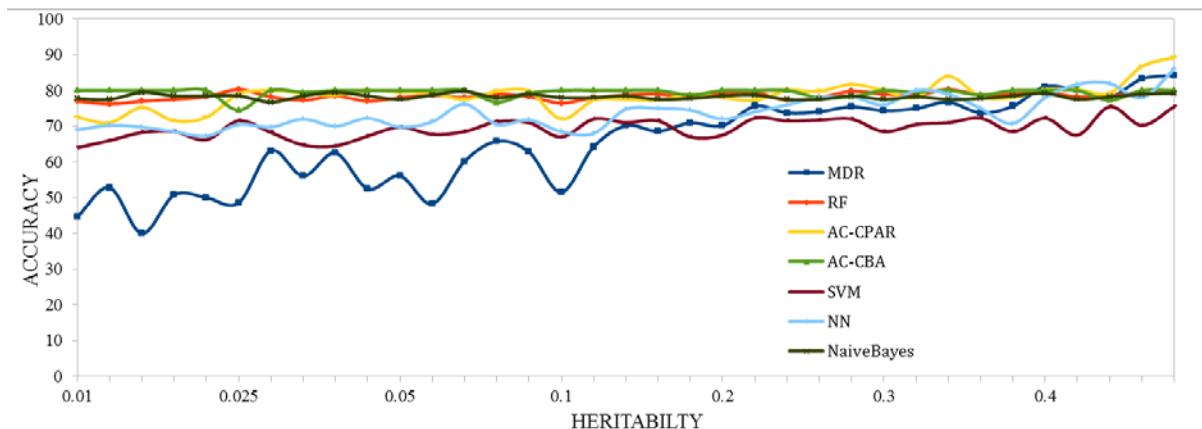


**Figure.3.9** Accuracy of 70 models with ratio 1:2 for MAF 0.2

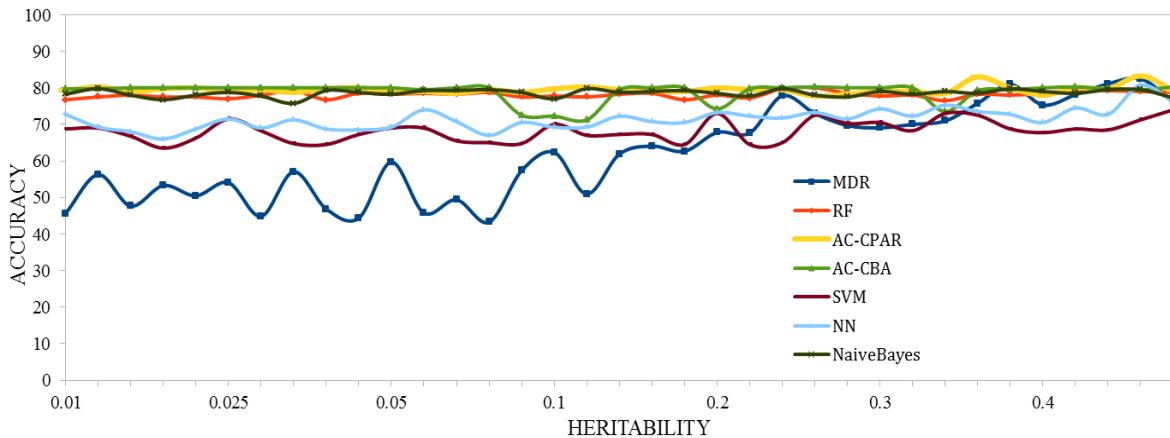
Figure 3.9 and Figure 3.10 graphically represents accuracy of AC for 1:2 ratio of sample size 400 with MAF 0.2 and 0.4 respectively. For average of MAF 0.2 and 0.4, the accuracy of AC is higher by up to 14% and 16% respectively compared to MDR for heritability values of 0.01, 0.025, 0.05 and 0.1. It is also been observed that, Naive Bayes' algorithm significantly performed better than MDR. However, on average AC was more accurate than Naive Bayes' algorithm for MAF 0.2 and 0.4 by upto 3% and 5% respectively. AC had the same accuracy as MDR for heritability 0.2, 0.3 and 0.4 for both MAF values (0.2 and 0.4).



**Figure.3.10** Accuracy of 70 models with ratio 1:2 for MAF 0.4



**Figure.3.11** Accuracy of 70 models with ratio 1:4 for MAF 0.2



**Figure.3.12** Accuracy of 70 models with ratio 1:4 for MAF 0.4

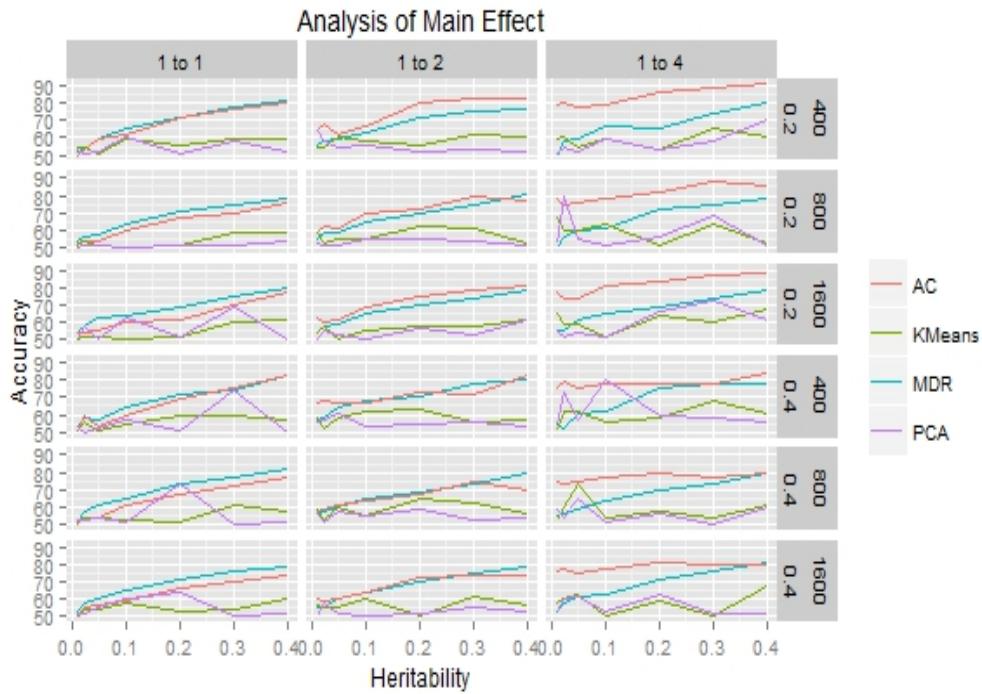
Figure 3.11 and 3.12 illustrates accuracy of AC for 1:4 ratios with MAF 0.2 and 0.4 respectively. AC predominantly outperformed in all 70 models compared with other existing approaches. These results demonstrate that the power of AC increases in imbalanced data with higher proportions of controls than cases.

### 3.5.2 Evaluation for main effect and higher-order interactions

The rule based method is evaluated to detect the prediction accuracy of interactions between SNPs that contributes to the disease. Series of datasets with samples consisting of 400, 800 and 1600 were simulated for 70 models with cases and controls of 1:1, 1:2 and 1:4 ratios for SNPs of one locus to six loci. From the preliminary evaluations, it was observed that the accuracy of AC was consistently higher in imbalanced datasets, though it showed only small improvements in balanced datasets over previous approaches. On an average of 100 datasets for each model, MDR performed well for all 6 models in balanced datasets. In this study, findings of the performance of AC are extended over single-locus to six-locus models. Further experiments are conducted to evaluate the performance of AC over some of the unsupervised algorithms, such as, KMeans and PCA, and are represented in Table 3.10 to Table 3.16.

Table 3.10 summarizes the prediction accuracy of 70 models with single functional SNP that contributes to the disease. The average prediction accuracy across 100 datasets for each of these models is evaluated to validate the method over the MDR, KMeans, and PCA. Threshold level is equal to 1 for the balanced datasets and it is 0.5 and 0.25 for

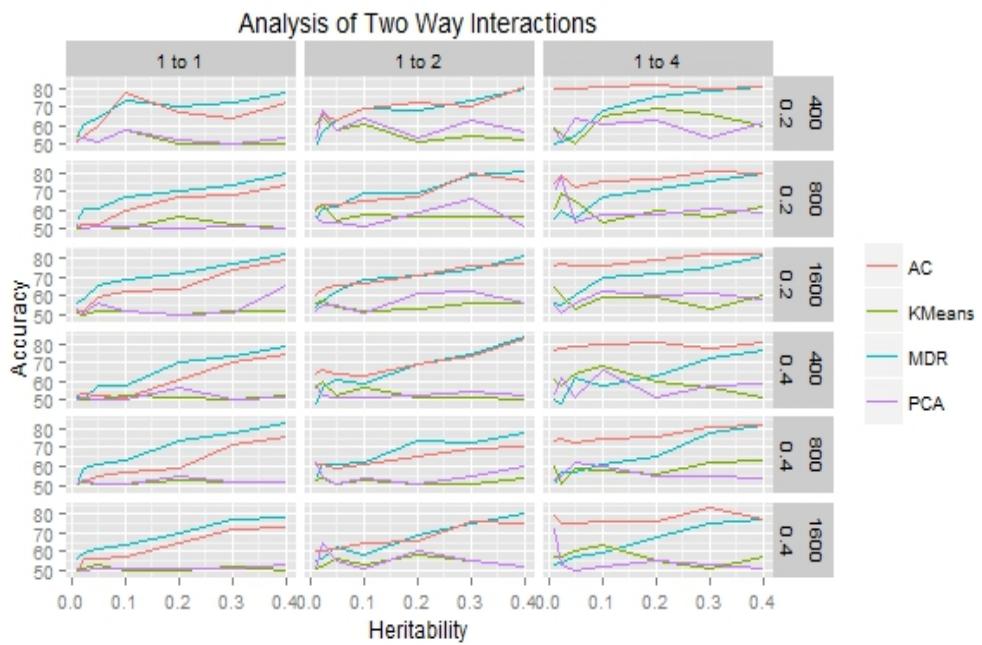
case-control ratios of 1:2 and 1:4 respectively. Figure 3.13 illustrates the accuracy of AC over MDR, KMeans, and PCA for single-locus models. Overall, the results show that AC performed better than MDR both in balanced and imbalanced datasets. In a 1:1 ratio, AC and MDR performed almost equally across all models. In 1:2 ratio, the accuracy of AC is slightly better than MDR by about 5%. In ratio 1:4, AC represents a 16% increase in accuracy over MDR.



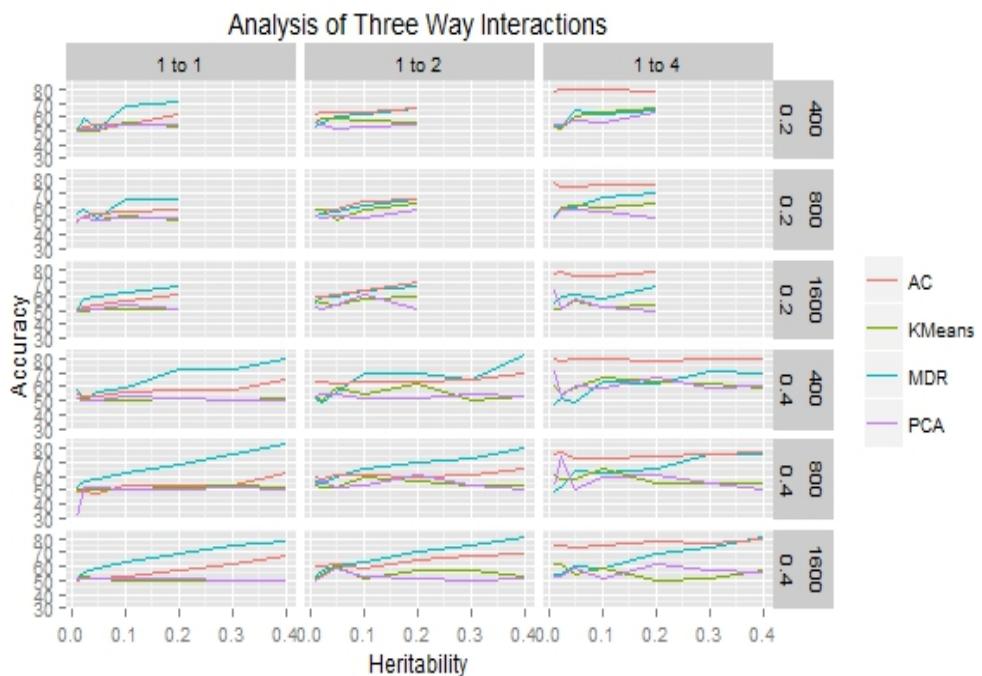
**Figure.3.13** Single-locus analysis

Table 3.11 summarizes the accuracy of 70 models with two-locus SNP interactions that contribute to the disease. The results are plotted in Figure 3.14. MDR performed better than AC in 1:1 case-control ratio. MDR and AC performed almost equally in 1:2 ratio. However, in a few models, AC showed slightly higher accuracy (1% to 4%) over MDR. In 1:4 ratio, AC performed well (11% to 16%) compared to MDR.

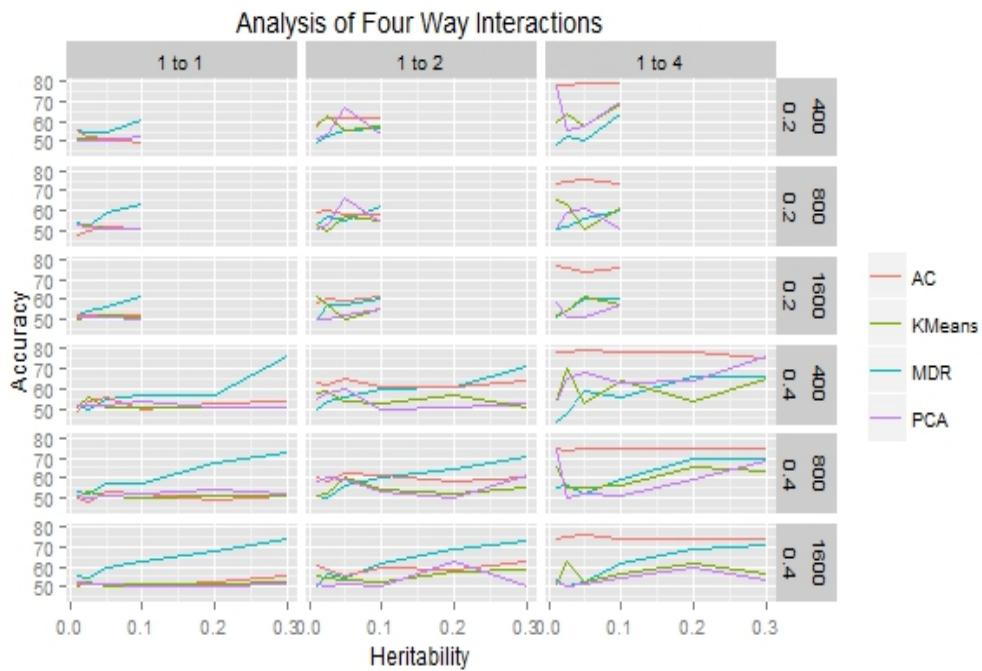
Table 3.12 summarizes the accuracy of 12 models with three-locus SNP interactions that contribute to the disease. There are three functional SNPs and seven non-functional SNPs among ten SNPs. The results demonstrated maximum accuracies for the proposed method in all 12 models. Figure 3.15 visually illustrates the accuracy of models against heritability. It represents the prediction accuracy of two algorithms against the case-



**Figure.3.14** Two-locus analysis



**Figure.3.15** Three-locus analysis

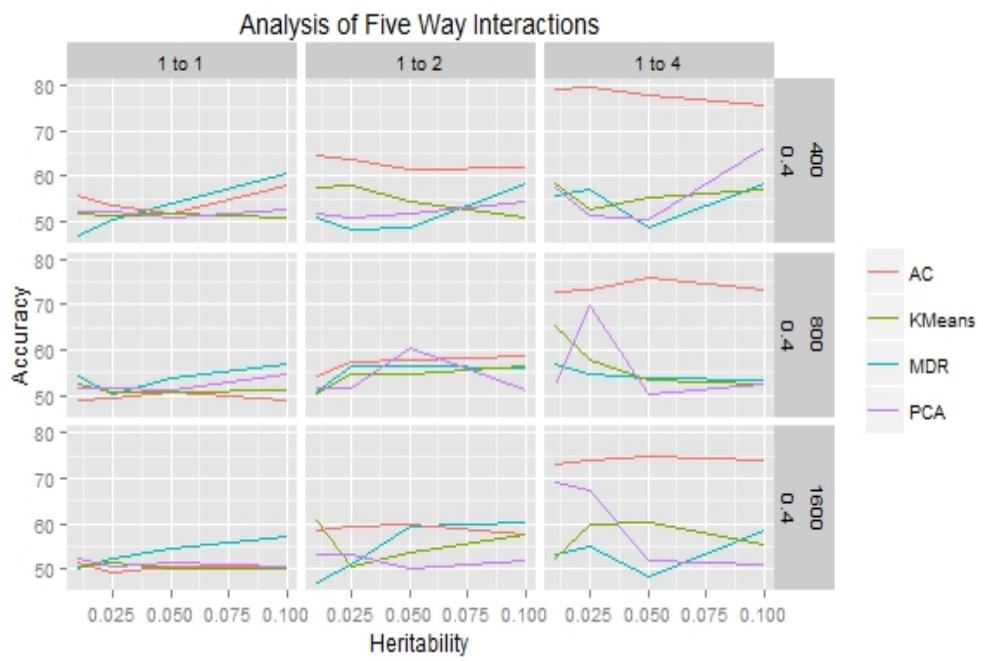


**Figure 3.16** Four-locus analysis

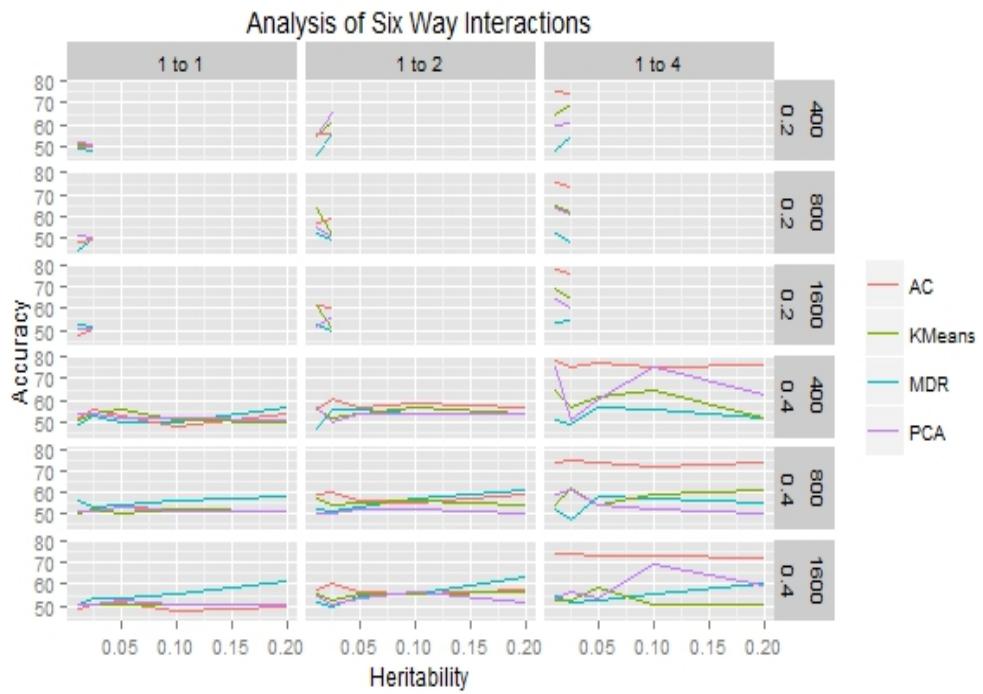
control ratio, sample size and MAF. As in two-way interaction models, MDR perform better than AC in 1:1 case-control ratios, and performed equally well with AC in 1:2 case-control ratios. However, AC predominantly performed well compared to MDR with an approximately 30% rise in prediction accuracy.

Table 3.13 summarizes the prediction accuracy of four-way interaction models. Ten models are generated with four functional SNPs and six non-functional SNPs which contribute the phenotype. In Figure 3.16, as expected AC performed poorly in balanced datasets and performed well in imbalanced datasets compared to MDR. Table 3.14 summarizes the accuracy of five loci interactions. Four models were generated with MAF = 0.4 and they are analysed by the proposed approach. Figure 3.17 illustrates the prediction accuracy of AC.

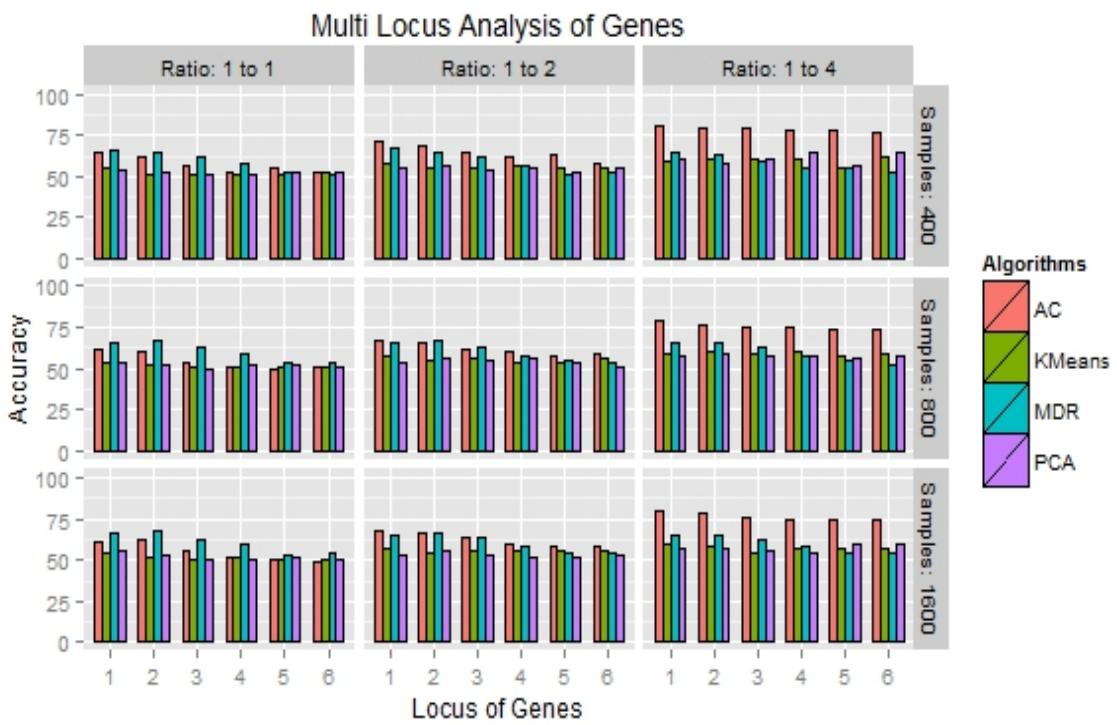
Similarly, Table 3.15 summarizes the accuracy of six way interactions. Seven models were generated with six functional SNPs and four non-functional SNPs. Figure 3.18 visualizes six-locus analysis of AC over previous approach.



**Figure.3.17** Five-locus analysis



**Figure.3.18** Six-locus analysis



**Figure.3.19** Multi-locus analysis

Table 3.16 summarizes the evaluation results from single-locus to six-locus interactions. Figure 3.19 graphically represents the multi-locus analysis of AC. The accuracy of KMeans and PCA are relatively less than MDR and AC in almost all the simulated scenarios. It is concluded that the AC approach consistently had higher prediction accuracy for imbalanced datasets. Hence, the rules generated from the model had better ability to identify the correct interaction model.

**Table 3.10:** Accuracy of single-locus models

MAF	H	400 Samples												800 Samples												1600 Samples											
		1:1 ratio				1:2 ratio				1:4 ratio				1:1 ratio				1:2 ratio				1:4 ratio				1:1 ratio				1:2 ratio				1:4 ratio			
		MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA
0.2	0.01	52	49.8	54	53	55.5	64	54.25	64	48.91	79.3	59.75	51	53	52.13	51.5	50.5	55.38	58.4	58.75	52.88	50.31	78	66.875	52.5	53.5	54.6	50.813	50.63	53.99	62.25	53.938	50.75	55.86	77.56	65.1875	55.813
0.2	0.05	59.5	59	50.75	51.5	59.2	62	60	55	58.91	78.3	55	53	57.88	53.5	51.125	51.1	58.65	61.3	54.375	51.25	59.53	76.13	59.5	54.88	62.12	55.4	51.438	50.06	58.97	61.13	50.875	52.56	61.05	73.88	59.125	53.563
0.2	0.03	53	53.8	54.75	50.5	58.3	68.5	54.5	57.8	58.59	80.8	61	55	55.62	53.13	53	51.4	59.04	62.1	52.25	51.5	56.17	74.75	59.25	79.63	57.19	54.1	51.125	57.13	57.53	60.19	58	54.81	54.8	74.25	58.8125	52
0.2	0.1	66	62.3	59.5	60.8	62.9	66.5	57.5	55.8	66.56	78.8	59.25	60	63.37	59.25	50.25	50	64.6	69.3	54.625	54.88	60.47	78.75	63.625	51	63.75	59.8	50.438	62.19	64.68	69.25	55.438	50.56	64.69	81.5	51.875	52.188
0.2	0.2	71.3	71.3	55.5	51	71.8	80.8	56.25	52.5	65.94	86	53.5	54	70.62	66.63	51.625	51.1	69.35	71.8	61.625	55.25	72.66	81.88	50.75	56.25	68.56	62	51.125	51.06	69.9	74.74	57.313	56.69	68.63	83.69	64.5	66
0.2	0.3	77.5	76.5	59	57.8	75.4	83.3	61.5	53.3	74.53	88.5	65.75	58	74.25	69.75	58.125	50.8	74.25	79.3	60.875	53.75	74.77	88.63	63	68	74.5	70.1	60.188	69.25	73.36	78.75	57.688	53.25	73.24	87.63	60.375	72.563
0.2	0.4	81.3	80.8	59.75	51.5	76.7	83	60	52	80.31	91.8	60	71	77.88	75.25	58.25	53	80.97	77.5	52.125	51	77.97	85.75	52.5	51.5	80.12	77.1	61.188	50.13	78.77	80.69	61.813	62	78.95	87.94	67.375	61.375
0.4	0.01	53.3	49.5	50.5	52.8	57.5	66.3	58.25	55.3	54.53	75.3	51	55	51.38	50.25	51.5	52.8	56.07	57.5	59	56.5	55.08	75	52.625	58	53.25	48.6	50.5	51.31	54.93	60.75	56.818	50.75	53.28	75.69	58	52.5
0.4	0.05	57.3	53	50.75	51.8	64.7	66.5	58.75	61	60.78	75.5	61.75	58	61.5	53.38	53.375	53.5	59.32	60.8	60.375	57.88	58.13	74.25	73.75	64.38	59.87	56.1	54.188	55.06	60.62	60.63	55.563	51.19	61.05	75.44	62.875	61.813
0.4	0.03	58.5	59.8	55.75	50	58.4	68.5	52	57.5	51.72	79.3	61.75	73	57.38	53.13	54.125	52.3	57.83	59	52.25	51.5	56.72	73.88	60	53.5	57.75	55.4	54.5	52.13	55.18	58.44	55.75	57.63	58.2	77	60.3125	60.313
0.4	0.1	64.3	59.3	54.75	56.5	67.7	67.3	61.5	52.8	61.87	78.3	56.25	80	64.25	61.38	52	51.6	64.65	63.4	54.625	54.88	63.44	77.38	53.375	51.25	65.38	59.5	57.438	60.25	63.48	64.56	60.75	50.13	62.73	77.38	50.25	53.125
0.4	0.2	71.8	69.8	59.75	51	70.2	72.8	63	54.3	75.63	77.5	58.25	60	73.25	66.75	51	73.4	68.44	67.5	65.25	58.38	69.69	80.13	57.75	56.63	71	66.3	53.188	64.31	70.27	72.31	50.875	51.75	71.13	80.88	58.5	63.188
0.4	0.3	74	75.8	59.75	74	77.5	71.5	55.25	55.5	77.5	78.3	68.5	58	76.88	71.75	60.875	50.3	73.58	74	62.875	52.75	73.67	76.5	53.25	50.5	76.5	70.4	54.688	50.82	75.52	73.63	61.563	54.75	76.09	80.38	50.8125	51.188
0.4	0.4	82.5	82.3	56.75	50	80.1	82.5	56.75	53.5	77.34	83.8	60.25	56	81.38	76.88	57.375	51.6	79.65	69.8	56.375	54	79.3	79.88	60.875	59.25	78.19	73.4	60.625	51.19	78.77	73.75	56.375	52.31	80.74	79.81	67.6875	51.188

**Table 3.11:** Accuracy of two-locus models

MAF	H	400 Samples												800 Samples												1600 Samples											
		1:1 ratio				1:2 ratio				1:4 ratio				1:1 ratio				1:2 ratio				1:4 ratio				1:1 ratio				1:2 ratio				1:4 ratio			
		MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA
0.2	0.01	54	51	54	52.3	48.8	60.5	60.5	51.8	49.69	79	59	58	54.5	51.5	50	52.9	55.39	61.4	59.75	57.25	55.94	75.13	59.5	70.88	55.69	52	51.1875	53.19	53.68	60.13	55.6875	52	55.74	75.81	64.6875	55.313
0.2	0.05	63.3	60	50.75	51	62.3	62.8	57.75	57.3	54.38	80	50.5	63	61	52.5	51.25	51.6	62.05	63.1	54.875	53.25	56.25	72.25	65	53.25	65.62	58.9	51.1875	55.88	62.77	65.19	53.8125	55.19	60.43	76.38	52.625	55.813
0.2	0.03	60.5	54	53.5	53.8	56.9	66.3	64.75	67.8	51.41	79.3	55.5	52	60.5	52.88	50.75	50.5	60.63	63.5	63.375	53.25	60.08	78.5	69.5	77.75	58.19	51.5	50.1875	51.13	57.3	63	57.3125	55.31	55.51	77.13	59.875	50.938
0.2	0.1	73.4	78	57.75	57.8	69.2	68.8	61	64	67.5	80.3	64.5	60	67.75	60.38	50.75	51.5	69.34	65.3	58	51.5	67.03	75.25	53.125	58.13	68.56	61.9	51.5625	51.75	68.97	66.25	52.3125	50.44	69.88	76.25	59.625	62.063
0.2	0.2	70.3	67	50.25	52	68.3	72.3	51.5	53.3	75	81.5	68.75	63	70.75	67.63	56.625	50.3	68.99	67.6	56.75	59	71.72	76.27	60.375	57.38	71.56	63.9	50.25	50.31	70.93	70.75	52.75	61.13	71.91	78.69	59.0625	60.375
0.2	0.3	72	64	50	50.5	73.2	70.5	54.25	62.5	77.97	80	66	53	73.87	68.75	52.5	51.4	78.92	79.8	56.75	66.13	75.31	80.63	57	60.75	76.63	73.5	51.5	51.13	73.79	75.69	55.8125	62.06	74.77	82.25	53.1875	61.313
0.2	0.4	77	72	50.25	53.5	79.3	80.5	52.5	56.8	80.31	80.5	59.5	62	79.87	73.63	50.875	50.5	80.96	75.4	56.375	51.63	80.08	80.25	61.75	59.38	81.94	78.9	52.125	65.81	81.58	76.63	56	56.19	81.8	82.75	60.5	58.5
0.4	0.01	51.3	53	50.25	50.5	47.3	63.8	57	57.8	49.84	76.8	60.25	53	51.25	50.38	50.625	50.8	53.61	61.4	53.125	62.13	50.55	73.25	6													

**Table 3.12:** Accuracy of three-locus models

MAF	H	400 Samples												800 Samples												1600 Samples											
		1:1 ratio				1:2 ratio				1:4 ratio				1:1 ratio				1:2 ratio				1:4 ratio				1:1 ratio				1:2 ratio				1:4 ratio			
		MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA				
0.2	0.01	49.5	50	52.25	50.8	52.7	62	55.75	56.5	52.5	79	53.25	55	55.25	48.5	50.625	50.8	53.43	58.1	58.5	53.75	53.67	76.88	52.625	52.63	51.31	49.2	51.0625	50.25	56.17	60.06	57.313	52.5	54.88	76.06	50.688	64.94
0.2	0.05	51.8	54	50.5	52.3	59.8	64	58.75	52	64.53	79.8	60.25	58	52.25	54.8	50.5	50.6	56.05	58.4	50.375	53.13	59.69	74.25	61.375	58.75	60.69	54.8	50.875	50.44	60.01	61.63	53.875	54.06	61.72	74.75	56.563	59.13
0.2	0.03	58.5	53	50	52	56.3	64	59.5	54.5	52.5	79.3	51	55	57.63	54	51.5	52.4	55.1	58.5	58	51.5	57.34	74.13	59.375	57.63	59.19	52.6	50.25	51	59.82	60.69	55.125	50.69	59.65	77.88	50.75	51.63
0.2	0.1	67.3	55	55.5	54	61.9	62.8	57.25	53	61.25	80	62.75	56	65.62	56.5	53.375	52.6	61.57	63.5	57.75	51.5	66.87	75.88	59.875	56.88	62.7	57.8	51.6875	53.44	65.12	64.81	58.688	61	58.52	75.63	51.938	53.25
0.2	0.2	70.8	61	53.5	55.3	66.6	66.8	56.5	54	65.31	77.8	66.25	64	65.13	58.8	51.25	52.1	65.43	64.9	62.625	58.25	69.61	75.5	62.5	51.5	68.12	60.9	50.8125	50.44	67.37	70.13	60	51.13	66.84	78.5	54	50.25
0.4	0.01	58	55	52	51.5	51.8	63.3	53.5	53.5	47.5	80	61	71	52.63	48.6	50.75	31.1	56.24	60	53.625	55.13	49.3	75.38	61.75	53	51.75	50.3	50.875	51.25	52.32	59.44	50.75	50.31	53.87	75.31	61.625	52.06
0.4	0.05	56.5	54	50.75	50.3	56.3	61.5	58.75	54.5	48.75	79.8	59.5	61	58.75	47.9	50.125	51.6	59.51	61.8	51.625	52.75	63.44	73	58	50.88	58.25	50.8	50.9375	50.56	60.25	60.94	60.563	59.38	59.45	74.25	53.688	59.31
0.4	0.03	50.5	53	51.25	50.3	48.8	64.3	50.25	55.3	52.34	78.5	54.5	53	56.12	48.9	51	52	55.86	58.3	52	56.75	51.64	77	58.25	74.25	55.06	50.8	52.0625	51.25	57.52	60.56	54.188	52.44	54.18	74.81	62	52.38
0.4	0.1	58.5	57	50.75	52.8	69.4	63	54.5	51.3	63.28	79.5	66.25	60	62.63	54.3	50.25	50.8	64.95	61.3	59	54.13	62.81	73.13	65.25	60	63.13	52.3	50	50.56	63.67	59.13	51.5	53.25	58.4	74.63	59.375	50.81
0.4	0.2	73	57	51	51.8	69	64.3	62	52.3	61.72	79	63	67	69.25	53.3	52.625	51.4	69.82	59.8	56	61.5	65.62	74.25	54.5	61.63	68.56	56.9	50.3125	50.88	70.37	64.56	57	50.75	69.45	77.56	50.063	61.56
0.4	0.3	72.3	58	50.5	50.8	64.7	64.8	50.75	55	71.09	79.3	61.5	59	75.37	54.4	53.125	52	73.01	61.4	53.875	53.88	75.78	76.25	55	55.5	74.75	61.1	50.4375	50.06	74.92	67.19	56.75	50.44	73.75	77.19	50.813	57.44
0.4	0.4	80.3	66	51.75	50.3	83.1	69.3	52.5	53.3	69.69	79.8	59.75	60	83.38	61.9	51.75	50.5	80.51	66.1	53	50.63	76.48	77.63	55.375	50.63	78.37	68.1	50.3125	50.31	80.69	69.25	52.563	50.94	80.55	79.13	56.813	55.19

**Table 3.13:** Accuracy of four-locus models

MAF	H	400 Samples												800 Samples												1600 Samples											
		1:1 ratio				1:2 ratio				1:4 ratio				1:1 ratio				1:2 ratio				1:4 ratio				1:1 ratio				1:2 ratio				1:4 ratio			
		MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA				
0.2	0.01	55.3	56	51.75	50.8	49.7	58.8	57.5	51.3	48	78	60.25	77	54	48.1	52.625	52.6	52.58	58.8	52.625	50.5	50.47	73.88	65	51	52.75	51.8	50.3125	51.44	49.04	58.56	61.875	50	51.21	77.13	51.75	58.75
0.2	0.05	54.5	51	50.25	50.3	56.1	61.5	56	67	50.6	79	57.75	58	59	51.8	50.9	54.84	58.1	56.625	66	56.33	75.13	50.875	61.13	56.12	51.2	51.9375	51.19	56.92	59.75	50.125	52.5	60.43	74.31	61.75	50.75	
0.2	0.03	55	53	51.5	50.5	52.2	62.3	62.75	53.3	52.2	78.3	64	55	52.25	49.6	52.875	51.8	56.62	60.6	50	53.38	51.8	74.63	63	59.38	54.19	52	51.375	50.94	57.71	60.63	57.6875	50.38	54.57	75.63	54.0625	51.375
0.2	0.1	60.5	50	53	52.8	56.5	62.3	57.25	54	63.4	79.8	69.5	70	63	50.8	50.75	50.6	61.76	58	55.5	54.25	60.55	73.38	61	50.38	61.19	52.5	50.8125	50.44	60.71	61.94	55	55	60.51	75.81	57.188	
0.4	0.01	51.8	49	50.5	52	49.5	63	58	54.3	43.6	78.3	54	54	53.37	50.1	50	51	51.08	60.1	50.875	58.13	55.31	75	65.125	73.13	55.44	51	50.6875	52.06	50.02	60.81	55.375	51.5	52.93	74.31	50.4375	52.375
0.4	0.05	54.8	56	50.5	51.8	56.1	64.8	53.75	60.3	59.1	79.5	52.25	68	57.38	53.4	51	51.3	56.24	61.9	59.75	59.5	52.03	74.5	55.375	51.5	60	50.6	50.25	50.88	54.1	55.88	53.75	50.81	52.19	76.06	52.5625	51.438
0.4	0.03	50	53	55.75	51.5	53.7	61.8	58.75	57.8	47.2	78	70.5	65	51.88	48.1	53.5	50.1	49.58	58.6	52.25	59.75	55.7	74.13	54.875	50	54.81	52.7	52	51.13	56.16	58.13	54.3125	50.13	50.59	75.25	62.625	50.125
0.4	0.1	57	49	51	53.3	60	61	52.25	50	55.9	78.3	63.75	63	56.75	52	50.25	51.6	60.54	61	54.125	53.25	58.91	74.63	56.125	51	62.19	50.4	51.75	50.38	61.74	59.25	51.8125	50.13	61.95	73.69	56.75	54.375
0.4	0.2	57.3	53	50.25	50.3	60.8	61.3	57.25	50.8	65.8	78	53.75	64	67.12	49.4	51.25	53.6	64.39	58.1	52	50.38	69.3	74.63	65.875	58.75	67.37	52.1	50.875	50.06	68.54	58.5	57.0625	62.31	68.87	74.25	61.5	59.813
0.4	0.3	76.3	53	50.25	50.8	71.1	63.8	50.75	52.5	65.6	75.5	65.25	77	72.75	51	50.625	52.3	70.58	60.4	55.375	61	69.61	75.13	63.625	68	73.75	55.3	52.3125	51.06	72.81	62.75	59	50.5	71.09	74.44	56.875	53.188

**Table 3.14:** Accuracy of five-locus models

MAF	H	400 Samples												800 Samples												1600 Samples											
		1:1 ratio				1:2 ratio				1:4 ratio				1:1 ratio				1:2 ratio				1:4 ratio				1:1 ratio				1:2 ratio				1:4 ratio			
		MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA				
0.4	0.01	47	56	51.75	52	50.7	64.5	57.25	51.5	55.62	79.3	58.25	58	54	48.75	52.625	51.6	50.71	54.4	50.375	51.75	56.95	72.88	65.25	52.25	50.31	51.4	50.6875	52.25	46.79	58.31	60.813	53	53.2	73.31	52.125	69.25
0.4	0.05	54	52	51.75	51	48.4	61.5	54.5	51.5	48.59	77.8	55.25	51	53.87	50.63	50.5	51.1	56.42	57.8	54.625	60.63	53.83	76	53.125	50.13	54.62	50.4	50.125	51.44	59.31	59.94	53.625	50.06	48.55	74.69	60.125	51.75
0.4	0.03	50.5	54	51.25	52	48.2	63.5	57.75	50.8	57.19	79.8	52.5	51	50.37	49.25	50.5	51.4	56.51	57.3	54.75	51.38	54.69	73.13	57.875	69.63	52.13	49.1	51.3125	50.69	51.01	59.5	50.688	53.38	55.12	74.19	59.625	67.188
0.4	0.1	60.5	58	51	52.5	58.4	61.8	50.75	54.3	58.44	75.5	57	66	57	48.88	51.125	54.8	56.14	58.6	56.5	51.13	53.12	73.38	52.375	52.5	57.06	50.4	50.3125	50.56	60.43	57.63	57.563	51.81	58.4	74.06	55.25	51

**Table 3.15:** Accuracy of Six-locus models

MAF	H	400 Samples												800 Samples												1600 Samples											
		1:1 ratio				1:2 ratio				1:4 ratio				1:1 ratio				1:2 ratio				1:4 ratio				1:1 ratio				1:2 ratio				1:4 ratio			
		MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA				
0.2	0.01	48.5	50	50.5	52.3	46.2	55.8	55.25	55	47.66	75	64.25	60	44.37	48.75	51.625	51.5	52.48	57.5	63.5	55	51.8	75.88	64.75	64	52.31	48.1	50.5	50.81	52.7	61.75	61.5	51.25	53.4	78	68.9375	64
0.2	0.03	47.8	50	51.25	51	55.9	56.3	61.5	65.5	55.16	74.5	69	61	49.88	49.5	50.125	50.6	49.49	58.9	51.25	50	48.05	73.5	61.625	61.38	51.19	50.6	50.375	50.31	49.37	60.5	50.25	56.5	54.77	76.18	64.0625	59.875
0.4	0.01	49.3	52	50.75	53.8	47.5	56.3	55.5	56.5	50.63	78.5	64.5	76	56.25	51.25	51.25	51	52.01	59.1	56.875	50.25	52.58	74	53.875	59.13	50.94	48.6	50.8125	50.38	51.67	57.38	55.125	54.56	54.8	74.31	52.6875	53.813
0.4	0.05	50	53	55.5	51.5	56	56.5	54.25	54	56.41	77.5	61.25	61	53.87	53.13	50.375	53	53.6	56.6	55.25	52.25	57.89	73.25	54.375	54.38	53.19	51.8	51.125	52.63	54.2	56.63	55.25	53.19	52.97	73.31	58.0625	54
0.4	0.03	53.3	56	53.5	54	55.9	60.5	51.75	50.3	48.91	75.3	56.75	51	52.88	51.63	52.125	51.1	50.98	60.3	53.875	50.25	47.03	74.75	62.125	61	53.25	50.7	50.5625	51	50.03	60	53.0625	50.38	51.48	74.31	52.5	56.063
0.4	0.1	49.8	48	51.25	51.8	54.3	58.5	56.5	54.3	55.62	75.3	64.5	76	55.75	52.5	52.25	51.5	56.69	55.4	56.375	52	56.72	72	58.625	52	56	47.6	50.3125	50.56	55.98	55.94	55.125	56.06	55.55	73.06	50.6875	69.438
0.4	0.2	56.5	54	50.25	50.5	53.9	56.8	54	54	51.72	76	51.75	62	58.38	51	51.5	50.9	61.48	59.5	54.375	50.13	54.84	74.13	60.875	50.38	61.12	49.9	50.6875	50.5	63.2	57.38	56.5	51.56	60.08	72.56	51.0625	59.438

**Table 3.16:** Average balanced accuracy of single-locus to six-locus models

Sample Size	Ratio	Balanced Accuracy																							
		Single Locus				Two Locus				Three Locus				Four Locus				Five Locus				Six Locus			
		MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA
400	1:1	65.9	64.46	55.804	54.4	65	61.48	51.643	52.2	62.2	55.9	51.646	51.8	57.23	52.15	51.475	51.4	53	54.7	51.438	51.9	50.71	51.9	51.857	52.1
	1:2	66.85	71.66	57.82	55.70	65	68.8	55.768	56	61.7	64.15	55.833	53.8	56.57	62.03	56.425	55.1	51.4	62.8	55.063	52	52.81	57.2	55.536	55.6
	1:4	65.22	80.79	59.43	60.00	63.2	79.39	60.09	58	59.2	79.29	59.92	60	55.14	78.25	61.1	65	55	78.1	55.75	56	52.3	76	61.71	64
	1:1	65.62	61.65	53.87	53.10	66.8	60.41	51.55	51.6	62.8	53.47	51.41	49.8	58.75	50.43	51.39	51.6	53.8	49.4	51.19	52.2	53.05	51.1	51.16	51.4
800	1:2	65.84	66.53	57.53	54.03	66.9	66.02	55.14	55.54	62.6	60.99	55.53	54.41	57.82	59.56	53.91	56.6	54.9	57	54.06	53.72	53.82	58.2	55.93	51.41
	1:4	64.85	78.63	59.08	57.66	65.5	76.54	59.571	59.21	62.7	75.27	58.656	56.94	58	74.5	60.088	57.4	54.6	73.8	57.156	56.13	52.7	73.9	59.464	57.46
	1:1	65.83	61.64	54.39	55.39	67.4	62	51.085	52.65	62.7	55.45	50.802	50.87	59.78	51.94	51.231	51	53.5	50.4	50.609	51.23	54	49.6	50.625	50.88
	1:2	65.43	67.93	56.63	53.51	66	67.12	54.446	55.77	64	63.2	55.693	53.07	58.78	59.62	55.6	52.3	54.4	58.8	55.672	52.06	53.88	58.5	55.259	53.36
1600	1:4	65.75	79.50	59.69	57.63	65.2	77.94	57.955	56.5	62.6	76.31	54.859	55.66	58.43	75.09	56.55	53.9	53.8	74.1	56.781	59.8	54.72	74.5	56.857	59.52
	1:1	65.83	61.64	54.39	55.39	67.4	62	51.085	52.65	62.7	55.45	50.802	50.87	59.78	51.94	51.231	51	53.5	50.4	50.609	51.23	54	49.6	50.625	50.88

### 3.5.3 Evaluation on Real data

The approach is further applied to sporadic breast cancer data to confirm the success rate of identifying the interactions between SNPs. AC and MDR methods are evaluated for each number of loci from one to nine. The five-locus model is identified by MDR as the best model with the highest prediction accuracy, and highest cross validation consistency (CVC). The best prediction model identified by MDR analysis includes the interaction between polymorphisms of COMT, GSTM1, CYP1A1m1, CYP1B1-codon 48, and CYP1B1-codon 432. The model had a maximum prediction accuracy of 53.41 and had a maximum CVC of ten. Statistical significance is determined by permutation testing under the null hypothesis of no association with the disease. The  $\chi^2$  value of the model is 95.4553, whose p value is less than 0.05 ( $p < 0.05$ ). Hence, the identified five-locus model is statistically significant. It is suggested that the interactions between five SNPs that occurs in four genes contributes to the association of the disease. The breast cancer data is also analysed using KMeans and PCA. The accuracy of the predicted models by KMeans and PCA are 52.1951 and 50.7371 respectively. However, the classification accuracy of both KMeans and PCA are relatively less than MDR.

AC identified six-locus model as the best model with the highest prediction accuracy, and highest CVC. The best prediction model identified by AC analysis includes the interaction between polymorphisms of Cyp1B1-119, Cyp1B1-432, Cyp1A1m1, GSTM1, COMT, and GSTT1. The identified model had highest prediction accuracy of 55.8537 with the highest CVC of ten. The accuracy of the predicted model by the implemented method is about 3% higher than the model identified by MDR analysis. Statistical significance of the model is evaluated by chi square test and whose *p value* is less than 0.05.

**Table 3.17:** Evaluation of sporadic breast cancer

Algorithms	Best Model	Number of Loci	Prediction Accuracy	CVC
MDR	Cyp1A1m1, Cyp1B1-48, Cyp1B1-432, COMT, GSTM1	5	53.41	10/10
AC	Cyp1B1-119, Cyp1B1-432, Cyp1A1m1, GSTM1,	6	55.8537	10/10

	COMT, GSTT1			
--	-------------	--	--	--

Table 3.17 illustrates the evaluation results of sporadic breast cancer data over MDR and implemented AC method. Hence, it is evident that the interactions between these six SNPs that occurred in five genes contributed to manifestation of sporadic breast cancer. The implemented method had better prediction ability compared to the previous approach MDR on real data.

## 3.6 Chapter Summary

In this chapter, an associative classifier was implemented for detecting SNP interactions in balanced and unbalanced data. Initial studies were performed to study the performance of the method on two-locus simulated datasets. The approach performed significantly better than the existing approaches in imbalanced datasets. However, the experimental results showed only small improvement in classification accuracy for balanced datasets. Further studies investigated the performance of the method over single-locus to six-locus interactions. The method was further successfully applied over sporadic breast cancer data. The results showed that the six-locus interaction model was responsible for the disease. These results were reported in terms of rules hence their interpretation was straightforward. In the next chapter, a MDR based AC is proposed by improving the performance of the method in high-dimensional genomic data.

# Chapter 4

## Multifactor Dimensionality Reduction based Associative classification

In the previous chapter, association rule mining and classification were integrated to extract all possible rules including hidden rules extracted to detect interactions between SNPs at different loci. AC builds an accurate and efficient classifier during the training phase of the model generation, by improving the ability of discovering true causal interactions responsible for a disease. A number of experimental results were demonstrated over balanced and imbalanced simulated datasets. AC performed significantly better than the previous traditional approaches in imbalanced datasets. However, the performance of the MDR was encouraging for all balanced datasets compared with other approaches. This motivated the research to be progressed by combining MDR and AC (MDRAC) in this chapter.

---

This chapter is based on following publications:

- S. Uppu, A. Krishna, and R. P. Gopalan, "A Multifactor Dimensionality Reduction Based Associative Classification for Detecting SNP Interactions," in *Neural Information Processing*, pp. 328-336, 2015: © 2015 Springer, "The original publication is available at [https://link.springer.com/chapter/10.1007/978-3-319-26532-2\\_36](https://link.springer.com/chapter/10.1007/978-3-319-26532-2_36) ."
- S. Uppu and A. Krishna, "Evaluation of associative classification-based multifactor dimensionality reduction in the presence of noise," *Network Modeling Analysis in Health Informatics and Bioinformatics*, vol. 5, pp. 1-9, 2016: © 2016 Springer, "The original publication is available at <https://link.springer.com/article/10.1007/s13721-016-0114-9> ."
- S. Uppu, A. Krishna, and R. P. Gopalan, "Combining associative classification with multifactor dimensionality reduction for predicting higher-order SNP interactions in case-control studies," in *International Journal of Computational Intelligence in Bioinformatics and Systems Biology, Inderscience publications*, accepted on 22/10/2017 (accepted - Under production).

The proposed hybrid MDRAC improves the performance of the method for both balanced and imbalanced datasets. The proposed method is evaluated for various simulated scenarios, and analyzed over one-locus to six-locus SNP interaction models. The performance of the method is studied in depth by adjusting threshold levels, and varying model parameters. The experimental results demonstrated improved accuracy over previous approaches by reducing classification errors. The proposed approach identified some of the interesting interacting SNPs at various locations that are not exposed by traditional approaches. However, the performance of MDRAC in the presence of noise due to genotyping error, missing data, phenocopy, and genetic heterogeneity is unknown.

In this chapter, MDRAC is further evaluated to determine the power of the method in the presence of noise. Several experiments were conducted on simulated datasets to demonstrate the performance of MDRAC in the presence of noise. On average, the results showed improved performance of MDRAC over other approaches. Further, the approach was successfully demonstrated over real world data application. A five-locus, and a three-locus interaction models were identified for the manifestation of sporadic breast cancer, and hypertension in humans.

This chapter is organized by presenting the overview of the proposed method in Section 4.1. Section 4.2 presents simulated datasets and real datasets used in this chapter to evaluate the method. Section 4.3 includes the data analysis implemented in the method. Section 4.4 discusses the evaluation of the method over simulated and real datasets. Finally, performance of the method in the presence of noise is evaluated and discussed in Section 4.5.

## 4.1 Integrating AC into MDR (MDRAC)

MDR is a non-parametric model used to detect interactions between multi-locus genes by reducing high-dimensional data to a single dimension using the constructive induction method [17]. The detailed study of MDR is reviewed in Chapter 2, Section 2.3.1. MDR exhaustively searches for all possible interactions associated with a complex human disease. AC combines associative rule-mining and traditional rule-based classification to build models [283], and was implemented in previous chapter. AC is integrated into MDR by proposing a hybrid approach MDRAC. That is, the method introduces the associative classification into MDR framework to formulate the proposed method. MDRAC is a model-free, non-parametric method that exhaustively searches all the possible  $n$ -locus SNP interactions associated to a disease. Figure 4.1 illustrates the workflow representation of the proposed method to detect higher-order SNP interactions (based on [17, 283]).

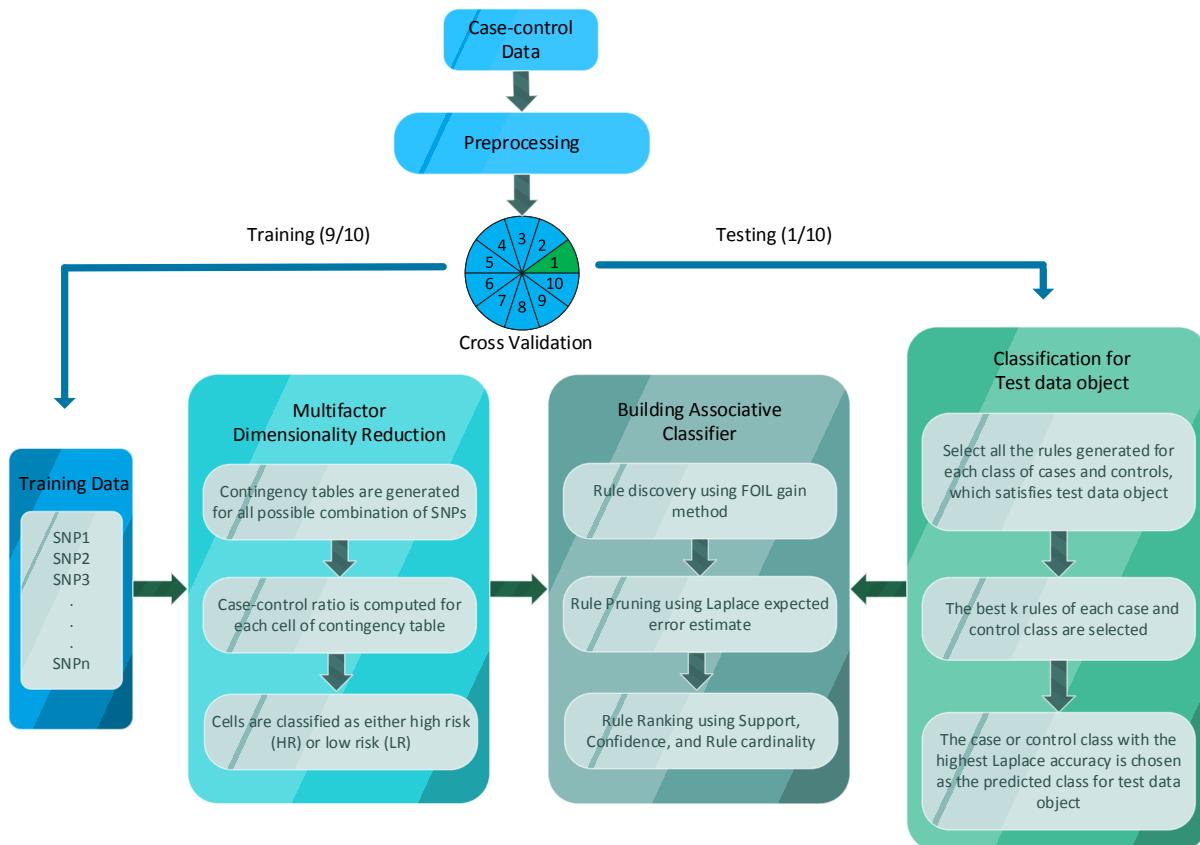


Figure 4.1 Workflow representation of the proposed MDRAC method.

MDRAC implements a series of seven steps to identify SNP-SNP interactions responsible for a complex disease. The steps (adopted from [17]) involved in implementing MDRAC method for a two-locus model is illustrated in Figure 4.2:

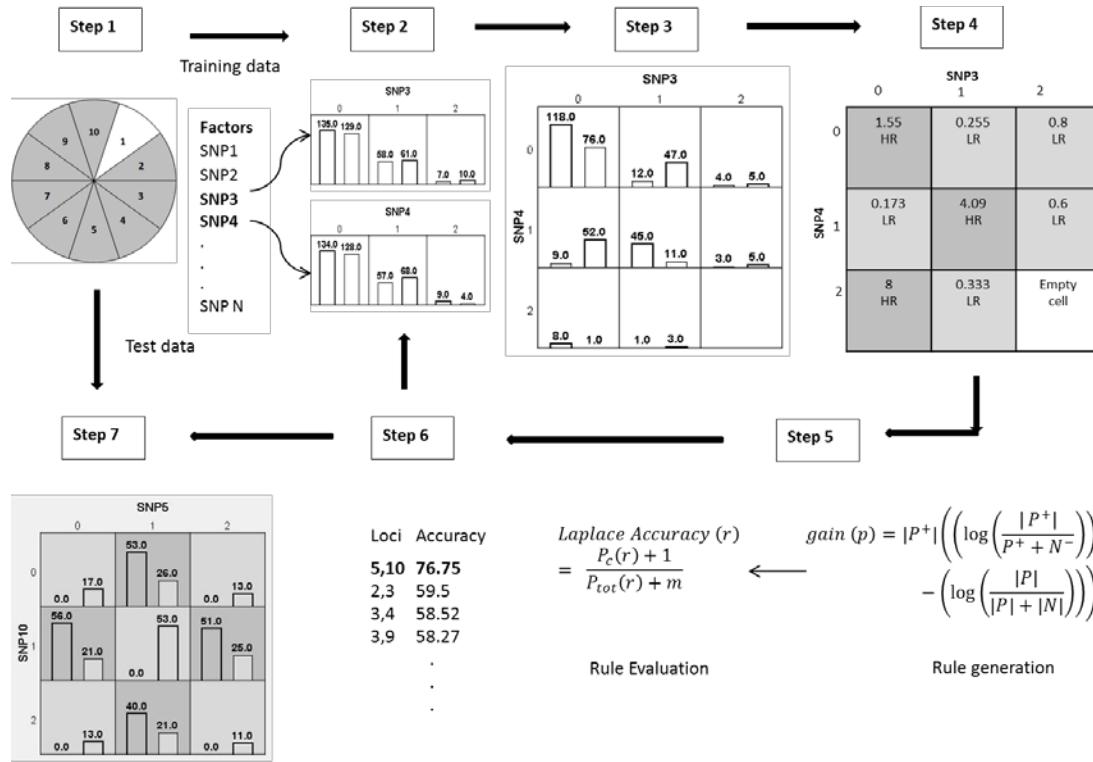


Figure.4.2      Summary of general steps involved in MDRAC.

The steps are summarized as follows:

**Step 1:** Ten-fold cross validation is performed on SNP data with sample size  $S$ . The data is divided into ten equal parts, in which nine parts are used for training, and remaining tenth part is used for testing. The cases and the controls of the class variables are represented numerically as 1 and 0 respectively.

**Step 2:** All possible combinations of SNPs in  $m$ -loci are enumerated. Due to duplication of DNA, a SNP is bi-allelic in nature with alleles A and a. The three possible genotypes of an SNP are common homozygous (AA), heterozygous (Aa/aA), and variant homozygous (aa). Numerically, AA, Aa/aA, and aa are represented by 0, 1, and 2 respectively.

In step 3, the number of cases and controls are counted for  $3^n$  genotype combinations for  $n$  loci in  $n$  dimensional space. For example, there will be  $3^2 = 9$  genotypes for two-locus interactions. Similarly, there will be  $3^3 = 27$  genotypes for three-locus interactions. That is, a  $3 \times 3$  contingency table is generated with 9 multifactor cells for two-locus interactions. Number of cases and controls of the subjects are placed in their corresponding cells in the contingency table.

Step 4: In each cell, the ratio of the number of cases  $c_1$  to the number of controls  $c_0$  is computed  $r = \frac{c_1}{c_0}$ . Their corresponding values are compared with the threshold value  $T$  for classifying the cells in the  $n$  dimensional space. The cells are classified as high risk if the value of the ratio is higher than  $T$ , and low risk if the value of the ratio is lower than  $T$ . The cells that are left blank represent the absence of genotype combinations in the datasets. The number of cases and controls are equal in balanced datasets, such that  $T = 1$ . The number of cases and controls are not equal in imbalanced datasets, such that  $T \neq 1$ . Hence, the threshold value is adjusted in imbalanced datasets for segregating the cells either as high risk (case-control ratio is higher than threshold value) or low risk (case-control ratio is lower than threshold value). The genotype combinations that are not represented in the dataset are left blank is termed as empty cells.

Step 5 uses AC to test for association with a disease by generating rules, implemented in the previous chapter for the classification. The rules are generated by using predictive rule mining (PRM) in which a weighted first order inductive learner (FOIL) gain method is used to improve the prediction [283]. The rules are generated in a greedy fashion by measuring FOIL gain  $F_G$  for each attribute  $A$  given by:

$$F_G = G_1' \left( \log \frac{G_1'}{G_1' + G_0'} - \log \frac{G_1}{G_1 + G_0} \right) \quad (4.1)$$

Where  $G_1$  , and  $G_0$  are high risk and low risk levels of a rule  $r$  in training dataset.  $G_1'$  and  $G_0'$  are high risk and low risk levels after best attribute gain  $p$  is added to the new rule  $r$ . The association rule  $r$  is represented as  $S \rightarrow Y$ , where  $S$ , and  $Y$  are the antecedent, and the consequent respectively. In the weighted version of FOIL, weights of  $G_1$  and  $G_0$  are decreased by multiplying a factor for producing more number of rules to improve the

accuracy, and efficiency of the algorithm. Further, the accuracy of the rules are evaluated using the Laplace expected error estimation. The Laplace expected accuracy  $L_A$  of a rule is obtained as:

$$L_A = \frac{G_c + 1}{G_t + k} \quad (4.2)$$

where  $G_c$  is the number of cases belonging to class  $c$ ,  $G_t$  is the total number of cases satisfying rule  $r$ , and  $k$  is the number of classes in the dataset.  $L_A$  is estimated for each rule. The best rules for the groups  $G_0$  and  $G_1$  are selected based on the highest Laplace expected accuracy.

Step 6: For each possible combination of SNPs, steps 2 to 4 are repeated. The rules are ranked further on the basis of support, confidence, and cardinality [278]. Confidence is the conditional probability of  $Y$  for a given  $X$ , and support is the percentage of transactions that contain both  $X$  and  $Y$ . Initially, the rules are sorted based on confidence. If the rules have identical confidence, they are sorted based on support. If the rules have the same support, they are sorted based on the least number of SNPs as antecedent. The rules are ranked randomly if the cardinality is the same. Finally, the classifier is built using generated rules, and the overall classification accuracy is calculated. The best model with the highest classification accuracy is chosen to predict the rules on the test data.

Step 7: steps 1 to 5 are repeated to avoid over fitting of the data in all the ten cross validation intervals. In all ten cross validations, the classification accuracy is averaged for each  $n$ -locus models. The overall best  $n$ -locus model with the highest CVC and the highest prediction accuracy is chosen. CVC is the number of times the  $n$ -locus model is chosen as the best model during the 10-fold cross validation. The balanced accuracy (the mean of sensitivity and specificity)  $B_a$  is calculated when cases and controls are not equal [108].

$$B_a = \frac{S_e + S_p}{2} \quad (4.3)$$

where  $S_e, S_p$  are the sensitivity and specificity respectively.

$$S_e = \frac{TP}{TP + FN} \quad (4.4)$$

$$S_p = \frac{TN}{TN + TP} \quad (4.5)$$

Where,  $TP$ ,  $TN$ , and  $FN$  are the true-positive, true-negative, and false-negative rates respectively. The  $TP$  rate is the proportion of positive cases that are classified correctly. The  $TN$  rate is the proportion of negative cases that are classified correctly. The  $FN$  rate is the proportion of positive cases that are classified incorrectly. The model with the least number of SNPs is selected if more than one model has the same prediction accuracy, and CVC [109].

## 4.2 Datasets

### 4.2.1 Simulated Datasets in the absence of noise

The datasets generated in Section 3.2.2 of the previous chapter are used to evaluate the proposed method (MDRAC) in the absence of noise. Datasets are simulated for 70 two-locus epistasis models with different heritability (H), and minor allele frequency (MAF). For each model, 100 datasets are simulated with different case-control ratios (such as, 1:1, 1:2, and 1:4), and sample size (such as, 400, 800, and 1600). In total, in this chapter, MRDAC is evaluated on 54,900 simulated datasets in absence of noise for one-locus to six-locus interactions.

### 4.2.2 Simulated Datasets in the presence of noise

The common sources of noise in genetic epidemiology studies is due to genotyping error, missing data, phenocopy, and genetic heterogeneity [36]. The datasets are generated in the presence of noise for six two-locus epistasis models that are explained in Section 3.2.1 of previous chapter [36] using GAMETES tool [221]. Genotyping error (GE) is generated by biasing 5% of genotypes towards one allele using directed-error model [296]. Missing data (MS) is generated by randomly removing 5% of genotypes. Phenocopy (PC) is generated such that 50% of cases had consistent low-risk genotypes corresponding to the penetrance function of the epistasis model. Genetic heterogeneity (GH) is generated such that 50% of

cases had one high-risk genotype combination (SNP5, SNP10) and other 50% of cases had other high-risk genotype combination (SNP3, SNP4). Six two-locus epistasis models with different penetrance values are simulated for ten SNPs with two functional SNPs (SNP5 and SNP10), and eight independent non-functional SNPs. Case-control datasets are simulated with 200 cases and 200 controls in accordance to Hardy-Weinberg proportions. Ten datasets are simulated in the absence of noise, and ten datasets are simulated for each type of noise (5% genotyping error, 5% missing data, 50% phenocopy, and 50% genetic heterogeneity). Further, ten datasets are simulated for each combinations of noise type to evaluate their combined effect on MDRAC. In total, 960 datasets are generated for six two-locus epistasis models.

### 4.2.3 Real Datasets

MDRAC is evaluated on three real datasets. The first dataset is obtained from the whole genome association studies published in SNPassoc package [297, 298]. The data comprise of 157 samples with 110 cases and 47 controls. There are 134 missing values. The data frame contains identifier, case or control status, sex, arterial blood pressure, protein levels, and 35 SNPs observations for each sample. The second dataset is obtained from Vanderbilt University Medical School [17]. The sporadic breast cancer data is explained in Section 3.3 of the previous chapter. It is also used to evaluate the proposed MDRAC in this study.

Hypertension is a complex disease, which is caused by the interplay of different factors acting together and/or independently. The data comprises of 443 outpatient samples, 313 hypertensives and 130 normotensives, collected from the National Taiwan University Hospital during July 1995 to June 2002 among the Taiwanese [299]. DNA extraction and genotyping were performed by a dye-terminator cycle, and an automatic sequencing method. The eight SNPs in four genes, rs4762, and rs699 at AGT, rs5050, rs5051, rs11568020, and rs5049 at AGT 5', rs4646994 at ACE, and rs5186 at AT1-R, that could increase the risk of hypertension is analysed in the study [300].

## 4.3 Data Analysis

The structure and the size of datasets play an important role in designing complex genetic

models. Balanced and imbalanced datasets are simulated based on cases and controls. The number of cases and controls are equal in balanced datasets, whereas they are not equal for imbalanced datasets. That is, the balanced datasets are analysed by considering the threshold value to be equal to one, and as not equal to one for the imbalanced datasets. The threshold value is defined as the ratio of number of cases to controls, which determines the disease risk of a multi-locus genotype combination. Imbalanced datasets can be analysed using oversampling or under-sampling or by adjusting the threshold value of MDR. Oversampling randomly re-samples the underrepresented class of samples by equalizing the number of cases and controls. Under-sampling randomly removes samples from an overrepresented class by equalizing the number of cases and controls in the datasets. Even though this technique is used in the literature [108], there can be false associations due to the samples that are oversampled or under sampled. This could also provide a false sense of high classification power. Hence, the imbalanced datasets are analysed by adjusting threshold values  $T$ . In this paper, the  $T$  value for 1:2, and 1:4 case-control ratios are set to 0.5 and 0.25 respectively. The simulated datasets include a small amount of missing data which may not have an impact on the power of MDR [301]. However, data imputation techniques (for large amount of missing data) are included in the data manipulation module of MDR software, publicly available from [www.epistasis.org](http://www.epistasis.org) to download.

Once datasets are analysed, configuration parameters are established for both MDR and AC. The configuration parameters for MDR are random seed, number of loci and  $m$  fold cross validation. The random seed is a random number used for shuffling randomly, which reduces the bias due to cross validation. The number of loci is selected from one to nine factors. That is, it selects the best models of one to nine factors and their association with a disease. The ten-locus model was not analysed due to the existence of only one form of this model, whose consistency is always ten. The configuration parameters of AC, the support, confidence, and length of antecedent of a rule, are set to 65%, 80%, and 9 respectively. The best 30 rules are used for the prediction, such that interactions between weaker SNPs are not neglected.

Once the configuration parameters are determined, the datasets (in the presence and in the absence of noise) are analysed using MDR, and the proposed MDRAC method with ten-fold

cross validation (CV). Ten-fold or five-fold CVs are the most successful internal model validation methods used for detecting SNP-SNP interaction models [118]. The data is equally split into  $m$  subsets without losing any data. One subset is considered as testing data and  $m - 1$  splits are used as training data. The algorithm runs on training data for each split by excluding different splits for testing data. That is, models are built ten times excluding one 10th of the data each time. The model is assessed by the remaining one tenth of the data. The best model is selected for each subset of the data. Finally, the overall best model is selected with the highest CVC, and lowest prediction error. It is repeated for each combination of  $n$  locus, such that the algorithm runs  $m$  times. Finally, the hypothesis test is performed over the best models of each locus to evaluate its statistical significance.

Prior to the analysis of sporadic breast cancer and hyper tension data, MDR and the proposed approach were evaluated using simulated datasets. The findings are confirmed over breast cancer and hypertension datasets with the same set of configuration parameters for MDR, and AC. However, the threshold values for breast cancer, and hypertension are set to 1.0147 (207/204), and 2.4077 (313/130). An exhaustive search was performed for all possible one to nine-locus models for the breast cancer data, and one to seven-locus models for the hypertension data. The final results are statistically evaluated with a 1,000 fold permutation test, whose p-values are compared with 0.05 in determining the statistical significance of the findings.

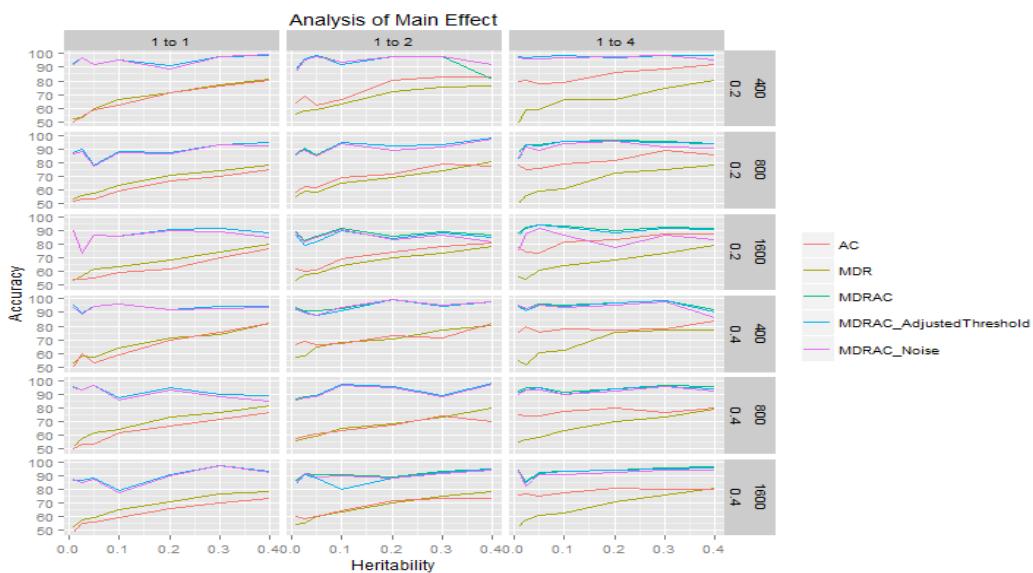
## 4.4 Evaluation on the proposed method

A number of experiments are carried out over several simulated scenarios, and the real datasets by evaluating the accuracy of MDRAC over some of the existing approaches. The experiments are conducted using weka [126], and wekacg [302] tools. The approach generates association rules based on cases, and controls that are statistically significant. Class label for the test example is predicted by using the best rules obtained from training. The accuracy of the proposed approach is improved by reducing the false positive errors. Hence, the results obtained from the proposed approach identified the higher-order interactions in the absence of main effects better than the existing approaches. The

preliminary results are illustrated on all the simulated scenarios, and the findings are confirmed on real datasets in the following sections by revealing the genotype-phenotype relationships.

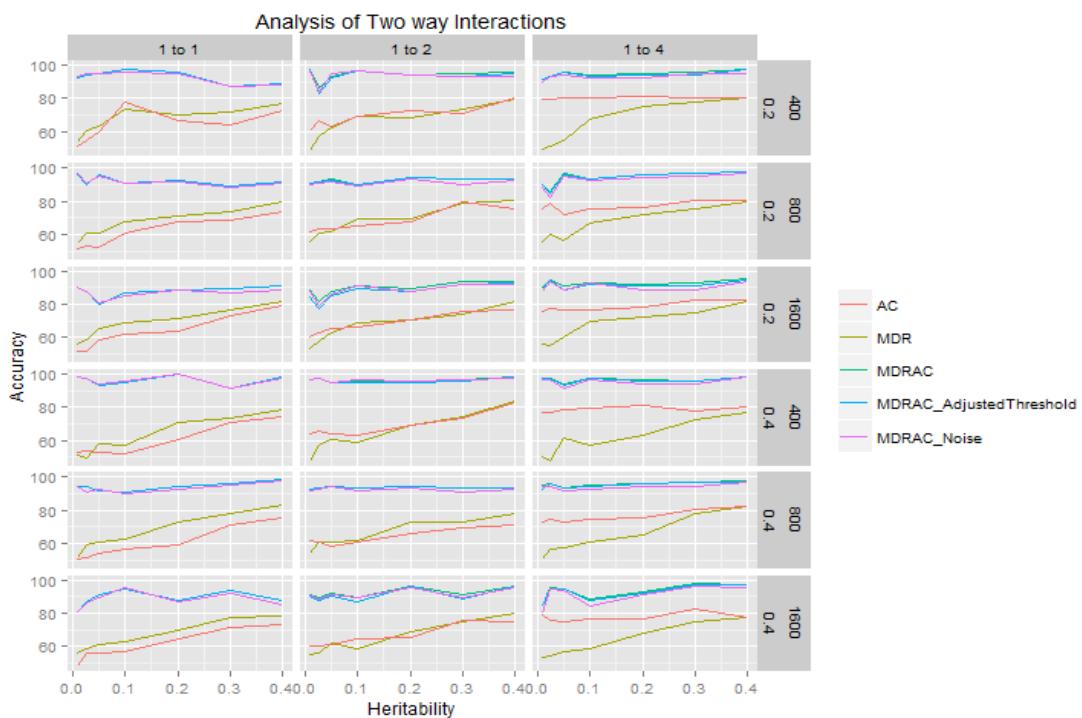
#### 4.4.1 Analysis of simulated data

The proposed method is evaluated for one to six loci interaction models on a series of simulated datasets consisting of 70 different models with 1:1, 1:2, and 1:4 case-control ratios along with 400, 800, and 1600 samples [108]. The accuracy of MDRAC is evaluated for each simulated scenario, as illustrated from Table 4.1 to Table 4.6. For each simulated scenario, 100 datasets are generated, and validated on MDR, AC, and MDRAC. The method is further evaluated in this study by adjusting threshold levels to improve the power of MDRAC for detecting associated SNPs. Table 4.1 illustrated the prediction accuracy of single-locus models that may have high risk of contributing to a disease. The MDRAC is evaluated for single-locus models by adjusting the threshold values. The threshold value is set to 1 for balanced datasets, and 0.5, and 0.25 for 1:2, and 1:4 case-control ratios (imbalanced datasets) respectively. The method is also evaluated by adding noise to the dataset in the preprocessing.



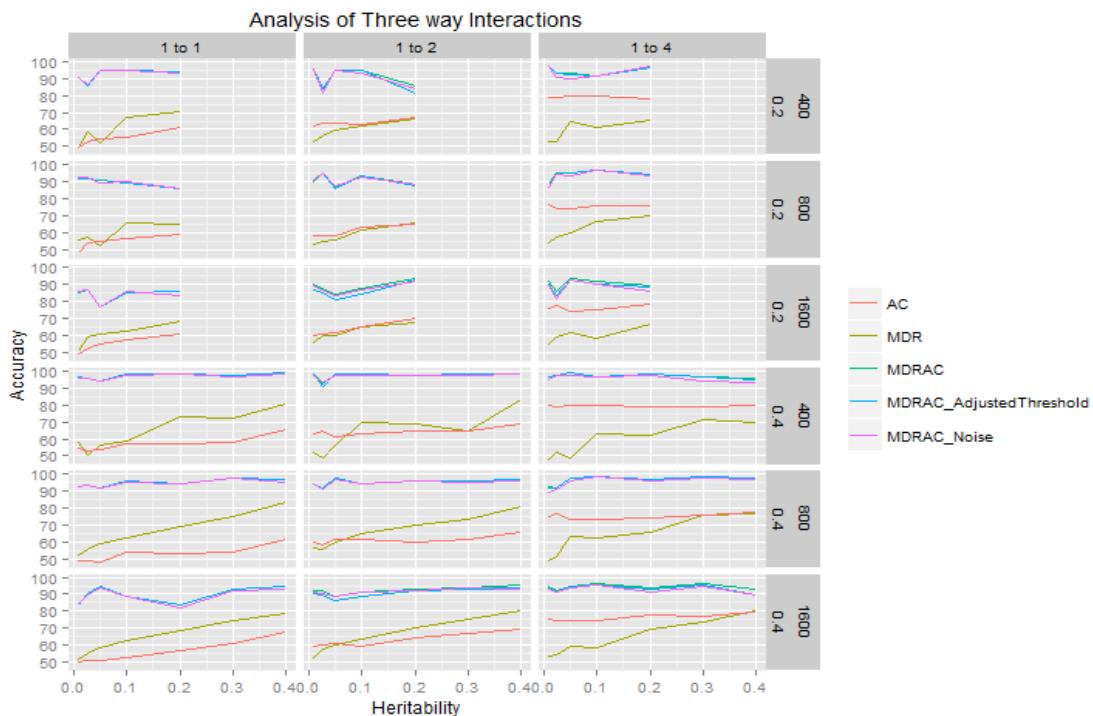
**Figure 4.3** Single-locus analysis for ratios 1:1, 1:2, and 1:4, sample size 400, 800, and 1600, and MAF 0.2, and 0.4.

Figure 4.3 exhibits the accuracy of MDRAC for single-locus analysis by adjusting the threshold values in the presence and absence of noise, and compared with MDR and AC. On average, the MDRAC performed better than the previous algorithms (such as, MDR and AC) for balanced and imbalanced simulated datasets. In general, AC and MDR performed equally well across all simulated models for all balanced datasets. The accuracy of AC is slightly higher than MDR (5% and 16%) for imbalanced datasets (1:2, and 1:4 case-control ratios). The accuracy of MDRAC is improved by 23% to 31% for 1:1, 1:2, and 1:4 case-control ratios, when compared with the original MDR. The performance of MDRAC is also observed by adjusting the threshold values, and adding noise to the datasets. The accuracy of MDRAC by adjusting threshold values is almost equal for both balanced and imbalanced datasets. Further, the performance of the models are also not much affected by introducing 10% noise to the simulated datasets. The accuracy of the models are reduced by less than 1% for all the samples of 1:1, 1:2, and 1:4 case-control ratios.



**Figure 4.4** Two-locus analysis for ratios 1:1, 1:2, and 1:4, sample size 400, 800, and 1600, and MAF 0.2, and 0.4.

Table 4.2 visually illustrates the accuracy of two-locus models for both balanced and imbalanced datasets. The experimental results of all two-locus simulated models are represented in Figure 4.4. MDR performed slightly better than AC for 1:1 ratio, and equally well for 1:2 ratios with the variation of 1% to 4% in accuracy values. However, AC performed 11% to 16% better than MDR for 1:4 ratios. It is also observed that MDRAC performed better than MDR and AC (maximum of 27% increase in accuracy) in all simulated models with the threshold value of 1. As expected, the accuracy of 1:2, and 1:4 ratio models are almost equal, when threshold values are adjusted to 0.5, and 0.25. The performances of the models are affected by less than 1% fall in accuracy by adding noise to the datasets.

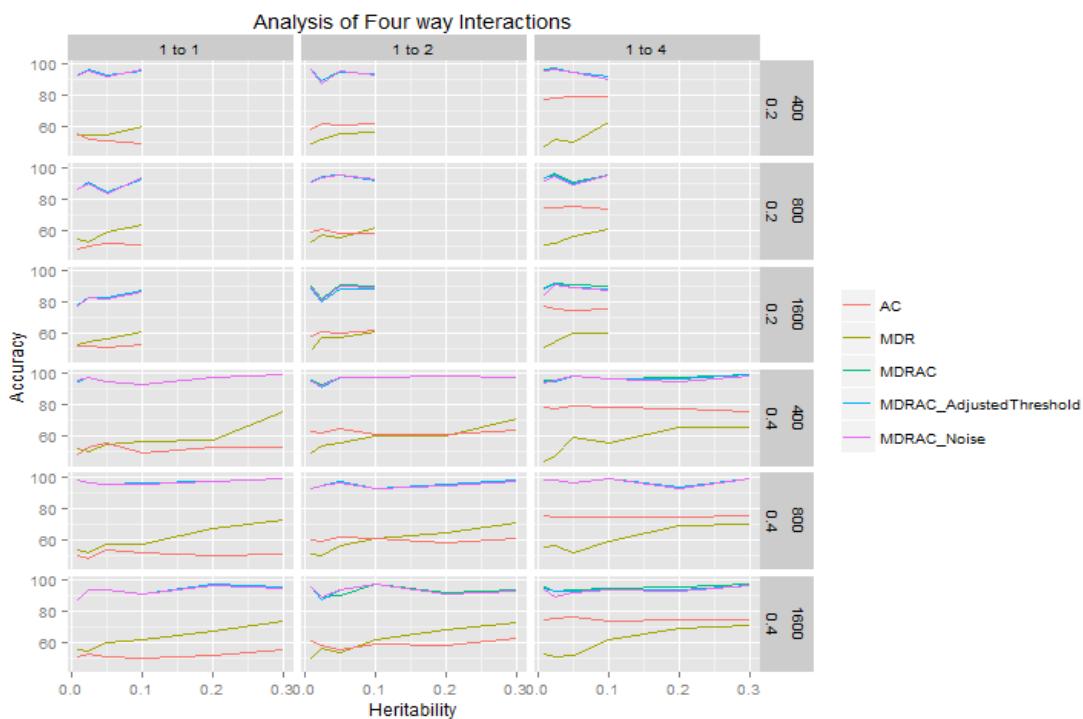


**Figure 4.5** Three-locus analysis for ratios 1:1, 1:2, and 1:4, sample size 400, 800, and 1600, and MAF 0.2, and 0.4.

Table 4.3 summarizes the accuracy of three-locus interaction models that may contribute to a disease. Figure 4.5 plots the accuracy of both balanced and imbalanced models in the presence of noise, and by adjusting threshold values. MDRAC performed well against MDR and AC with 30% rise in accuracy. However, MDRAC performed equally well by adjusting

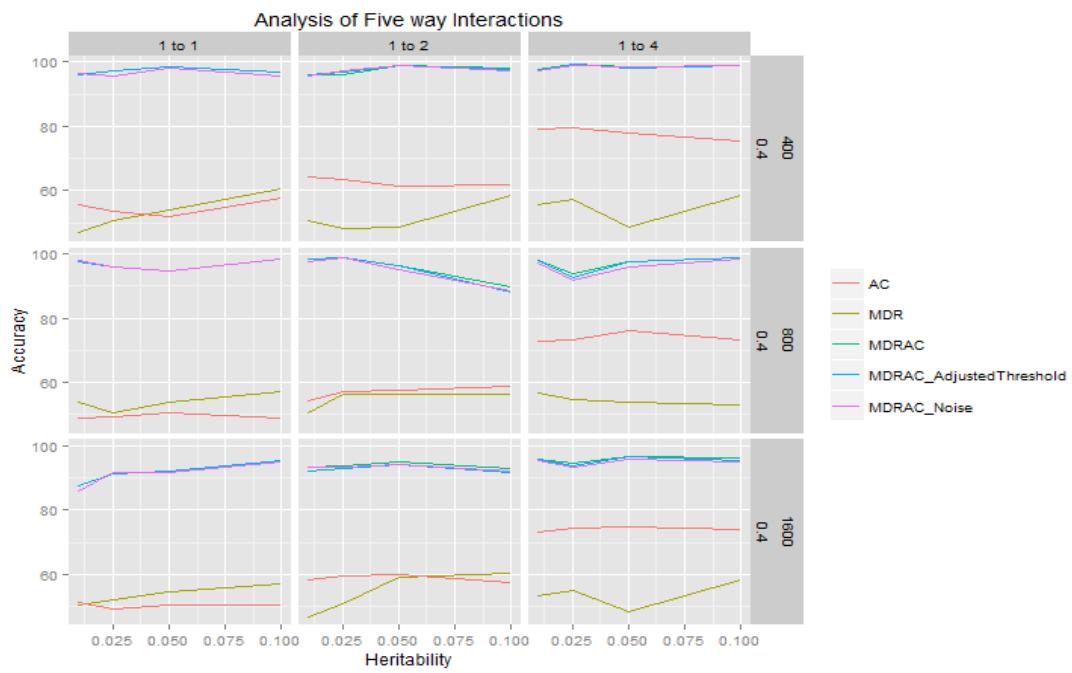
threshold values, and adding noise to the datasets (with less than 2% of fall in accuracy for 1:4 ratios).

Table 4.4 summarizes the accuracy of four-locus SNP-SNP interaction models for balanced and imbalanced simulated datasets. Figure 4.6 visually illustrates the performance of MDRAC in the presence and absence of noise. Further, it illustrates the performance of MDRAC by adjusting threshold values, and comparing with MDR and AC. The results demonstrate an improved prediction accuracy of 89% to 96% by reducing prediction errors. As expected, all the four-locus models are consistently stable after adjusting threshold values and adding noise to the datasets.

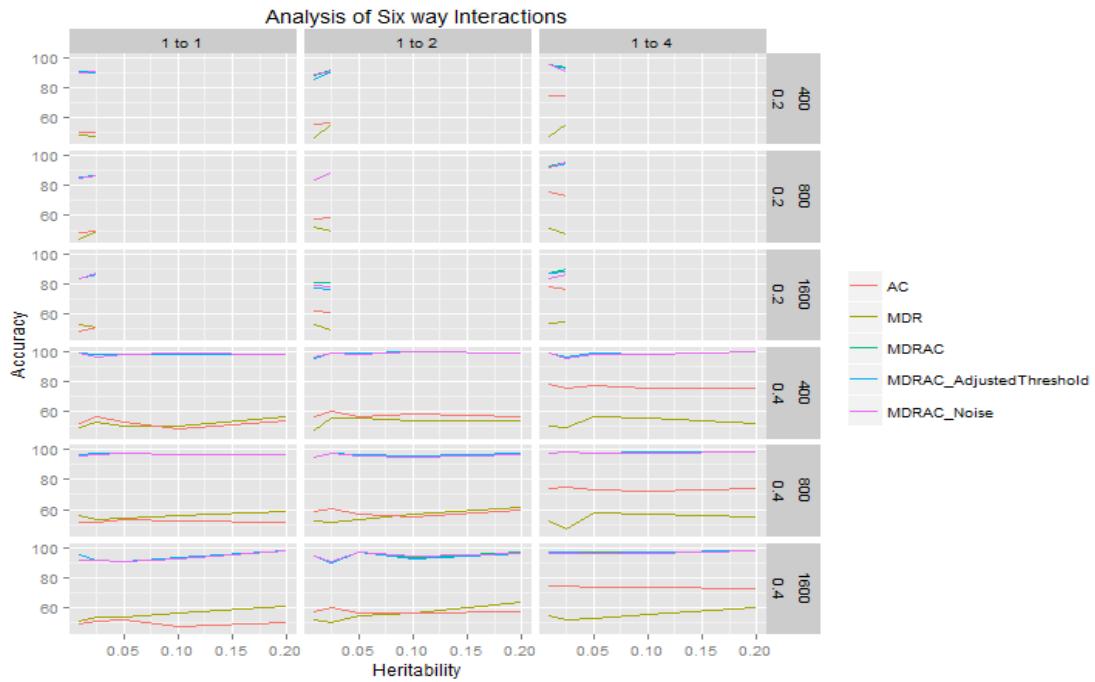


**Figure 4.6** Four-locus analysis for ratios 1:1, 1:2, and 1:4, sample size 400, 800, and 1600, and MAF 0.2, and 0.4.

Similarly, Table 4.5 and Table 4.6 summarize the accuracy of five-locus, and six-locus interaction models. Figure 4.7 and Figure 4.8 visually illustrate the performance of MDRAC for five and six loci models.

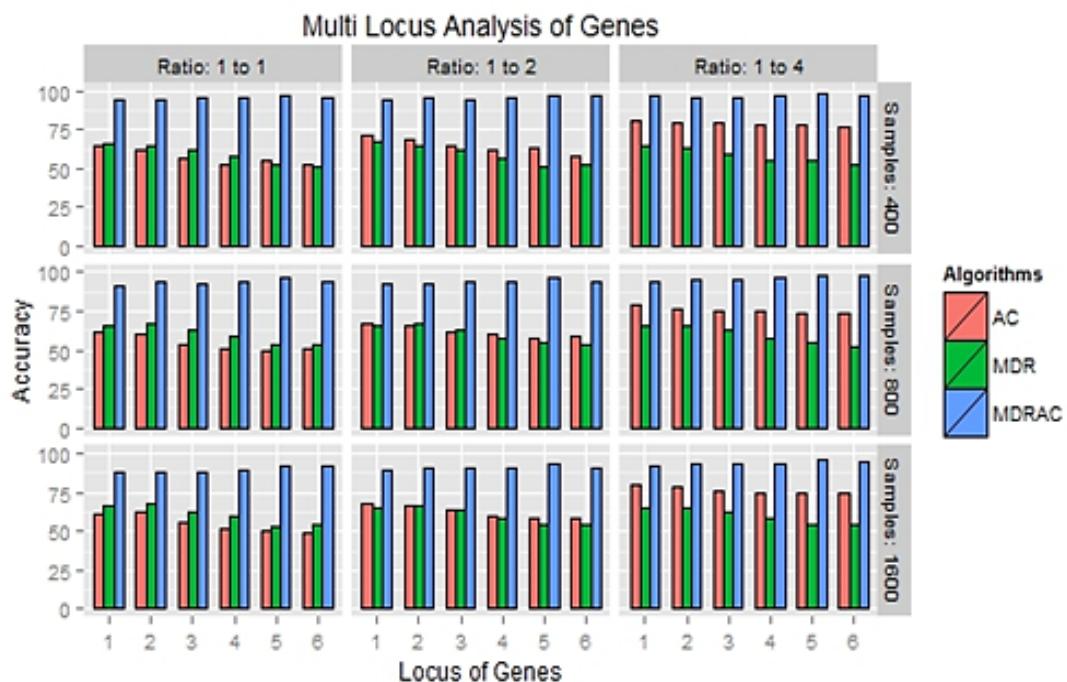


**Figure.4.7** Five-locus analysis for ratios 1:1, 1:2, and 1:4, sample size 400, 800, and 1600, and MAF 0.2, and 0.4.



**Figure.4.8** Six-locus analysis for ratios 1:1, 1:2, and 1:4, sample size 400, 800, and 1600, and MAF 0.2, and 0.4.

The average balanced accuracy of MDRAC on simulated datasets for one-locus to six-locus models are summarized, and illustrated below in Table 4.7 and Figure 4.9 respectively. From the observations, it is concluded that the MDRAC performed consistently better than MDR and AC, with improved prediction accuracy for all simulated scenarios by reducing prediction errors.



**Figure 4.8** Summary of multi-locus analysis for ratios 1:1, 1:2, and 1:4, sample size 400, 800, and 1600.

**Table 4.1:** Evaluation for single-locus models

		400 Samples												800 Samples												1600 Samples																			
		1:1 ratio				1:2 ratio				1:4 ratio				1:1 ratio				1:2 ratio				1:4 ratio				1:1 ratio				1:2 ratio				1:4 ratio											
MAF	H	MDR	AC	MDRAC	Noise	MDR	AC	MDRAC	Adjusted	MDRAC	Noise	MDR	AC	MDRAC	Adjusted	MDRAC	Noise	MDR	AC	MDRAC	Adjusted	MDRAC	Noise	MDR	AC	MDRAC	Adjusted	MDRAC	Noise	MDR	AC	MDRAC	Adjusted	MDRAC	Noise										
0.2	0.01	52	49.75	92.5	91.75	55.51	64	89	88.25	87.25	48.91	79.25	97.75	97.75	97	53	52.125	87.75	86.375	55.38	58.375	86.5	85.375	86.25	50.31	78	87	82.25	83.125	53.5	54.625	90.375	53.99	62.25	89.6875	86.5625	88.75	55.86	77.563	88.375	86.9375	75.1875			
0.2	0.05	59.5	59	92.25	92	59.16	62	98.25	98.25	97.5	58.91	78.25	97.5	97.5	96	57.88	53.5	78.125	77.75	58.65	61.25	85.75	84.75	84.625	59.53	76.13	93.125	92.5	89.375	62.12	55.375	86.8125	86.6875	58.97	61.125	86	82.125	85.3125	61.05	73.875	94.3125	94	91.9375		
0.2	0.025	53	53.75	97	97	58.28	68.5	95.75	95.75	95	58.59	80.75	97.25	97	96.25	55.62	53.125	89.75	88.5	59.04	62.125	90.375	89.625	89.375	56.17	74.75	92.875	92.75	91.7	57.19	54.125	73.5625	73.6875	57.53	60.188	82.9375	79.1875	81.6875	54.8	74.25	92.375	91.8125	87.9375		
0.2	0.1	66	62.25	95	95	62.9	66.5	93.75	92.25	93.75	66.56	78.75	98.25	98.25	96.75	63.37	59.25	88.5	87.5	64.6	69.25	94.625	94.625	94	60.47	78.75	95.75	95.75	94	63.75	59.813	86.0625	86.3125	64.68	69.25	91.375	89.875	90.9375	64.69	81.5	93.3125	92.3125	87.1875		
0.2	0.2	71.25	71.25	91.25	89	71.81	80.75	98	98	65.94	86	97.25	97.25	97.5	70.62	66.625	87.625	86.625	69.35	71.75	92.25	92.125	88.75	72.66	81.88	96.75	95.75	96	68.56	62	90.75	90.375	69.9	74.738	86.0625	84.625	83.4375	68.63	83.688	90.125	88.625	77.875			
0.2	0.3	77.5	76.5	98	98	75.38	83.25	97.75	97.75	97.75	74.53	88.5	98.5	98.5	98.5	74.25	69.75	93.375	93.125	74.25	79.25	93.5	93.125	91.625	74.77	88.63	95.5	94.625	91.75	74.5	70.125	91.4375	89.625	73.36	78.75	89.3125	88.375	86.5	73.24	87.625	92.25	91.75	86.875		
0.2	0.4	81.25	80.75	99	99	92.25	76.69	83	81.25	91.75	91.75	80.31	91.75	98.25	98.25	98.25	92.25	92.25	80.97	77.5	98.25	98.25	98.25	92.25	92.75	77.97	85.75	93.75	93.75	93.75	90.5	80.12	77.125	88.125	85.5	78.77	80.688	86.625	84.875	82.025	79.95	87.938	91.625	90.6875	83.5625
0.4	0.05	53.25	49.5	95	93.75	57.47	66.25	93.5	92.25	93.25	54.53	75.25	94.5	94.5	95.25	51.38	50.25	95.75	95	56.07	57.5	86.375	85.75	85.375	55.08	75	92.625	91.875	89.75	53.25	48.625	87.125	88	54.93	87.45	85.3125	53.28	75.688	94.0625	93.3125	93.1875				
0.4	0.05	57.25	53	94.25	94.75	64.73	66.5	91	88	87.75	60.78	75.5	96.25	95.5	95.5	61.5	53.375	96.625	96.625	59.32	60.75	89.375	88.75	87.875	58.13	74.25	95.25	95	93.5	59.87	56.063	88.625	87.875	60.62	60.625	91.0625	88.75	89.625	61.05	75.438	92.4375	92	91.3125		
0.4	0.025	58.5	59.75	89.5	89	58.37	91.25	90.25	89.75	51.72	79.25	93.25	91.25	93	57.38	53.125	93	93	57.83	59	88.375	87.125	87.625	56.72	73.88	94.75	93.75	93.25	57.75	55.438	86.625	85.0625	55.18	58.438	92	90.9375	92.1875	58.2	77	85.9375					
0.4	0.1	64.25	59.25	96.5	96.5	67.74	67.25	92.75	91.5	93.5	61.87	78.25	95	94.25	93.5	64.25	61.375	87.625	85.875	64.65	63.375	97.125	97.125	96.625	63.44	77.38	91.5	89.75	89.625	65.38	59.5	79.3125	77.9375	63.48	64.563	91.25	80.0625	89.9375	62.73	77.375	93.625	93.1875	91.125		
0.4	0.2	71.75	69.75	92.25	92.25	70.19	72.75	99.25	99.25	99.5	75.63	77.75	97.25	97.25	95.75	73.25	66.75	94.5	93.375	68.44	67.5	95.375	95.375	94.75	69.69	80.13	94.25	93.625	92.375	71	66.313	91.0625	89.9375	70.27	72.313	89.6875	88.625	88.4375	71.13	80.875	94.625	94.0625	92.375		
0.4	0.3	74	75.75	94.25	93.25	77.5	71.5	95.5	94.5	95	77.5	78.25	99	99	98.25	76.88	71.75	89.625	88.125	73.58	74	89.375	89.25	88.125	73.67	76.5	95.875	96	76.5	70.438	97.625	97.4375	75.52	73.625	93.4375	92.75	92.125	76.09	80.375	96	94.875	94			
0.4	0.4	82.5	82.25	94.25	93.5	80.14	82.5	98.25	98.25	98	77.34	83.75	92	90	86.5	81.38	76.875	89.125	84.75	79.65	97.875	97.875	97.375	79.3	79.88	95.375	94.25	92.45	92	78.19	73.375	93.8125	92.9375	78.77	73.75	94.875	94.875	94.875	94.875	94.875	94.875	94.875	94.875	94.875	94.875

**Table 4.2:** Evaluation for two-locus models

		400 Samples												800 Samples												1600 Samples																	
		1:1 ratio				1:2 ratio				1:4 ratio				1:1 ratio				1:2 ratio				1:4 ratio				1:1 ratio				1:2 ratio				1:4 ratio									
MAF	H	MDR	AC	MDRAC	Noise	MDR	AC	MDRAC	Adjusted	MDRAC	Noise	MDR	AC	MDRAC	Adjusted	MDRAC	Noise	MDR	AC	MDRAC	Adjusted	MDRAC	Noise	MDR	AC	MDRAC	Adjusted	MDRAC	Noise	MDR	AC	MDRAC	Adjusted	MDRAC	Noise								
0.2	0.01	54	51	92.5	92.75	48.81	60.5	97	96.75	49.69	79	91.75	91.75	89.5	54.5	51.5	96.25	96.5	55.38	61.375	90.625	89.875	55.94	75.13	90.25	90	88.625	55.69	52	90.5	90.5	53.68	60.125	88.1875	84.6875	87.5	55.74	75.813	90.5	89	87.9375		
0.2	0.05	63.25	59.5	95.25	95.25	62.31	62.75	92.75	92.25	94.5	54.38	80	96	95.75	94	61	52.5	96.25	95.375	62.025	63.125	93.625	92.875	91.875	56.25	72.25	96.5	96.375	95.5	65.62	58.938	79.5625	80.6875	62.77	65.188	87.4375	85.5	85.5625	60.43	76.375	91	88.625	88.625
0.2	0.025	60.5	54.25	94	95	56.9	66.25	83	84	51.41	79.25	93.25	93	93.25	60.5	52.875	90.125	90.625	60.63	63.5	91.75	91.75	91	60.08	78.5	86	84.625	82.125	58.19	51.8	88.625	88.125	57.3	63	81.375	77	79.3125	55.51	77.125	94.5	94.5	93.5625	
0.2	0.1	73.44	77.5	97.75	96	69.2	68.75	96.5	96.5	96.25	67.5	80.25	94	98.25	92.25	67.75	60.375	90.625	91.25	69.34	65.25	90.25	90.125	89	67.03	75.25	93.625	93.5	92.75	68.656	61.938	85.125	86.97	66.25	91.375	89.625	91.3125	69.88	76.25	92.125	92.5625		
0.2	0.2	70.25	66.5	95.75	95.25	68.26	72.25	94	94	94.25	75	81.5	94.5	94.25	92	70.75	67.625	92.125	91.875																								

**Table 4.3:** Evaluation for three-locus models

	MAF	H	400 Samples												800 Samples												1600 Samples																	
			1:1 ratio			1:2 ratio			1:4 ratio			1:1 ratio			1:2 ratio			1:4 ratio			1:1 ratio			1:2 ratio			1:4 ratio			1:1 ratio			1:2 ratio			1:4 ratio								
			MDR	AC	MDRAC	Noise	MDR	AC	MDRAC	Adjusted Threshold	MDR	AC	MDRAC	Noise	MDR	AC	MDRAC	Adjusted Threshold	MDR	AC	MDRAC	Noise	MDR	AC	MDRAC	Adjusted Threshold	MDR	AC	MDRAC	Noise	MDR	AC	MDRAC	Adjusted Threshold	MDR	AC	MDRAC	Noise						
	0.2	0.01	49.5	49.5	90.5	52.71	62	95.75	95.75	95.5	52.5	79	97.5	97.5	97.25	55.25	48.5	91.75	92.625	53.43	58.125	90.375	90.75	53.67	76.88	88.375	87	85.75	51.31	49.188	85.4375	85.875	56.17	60.063	90.1875	86.9375	89.1875	54.88	76.063	91.6875	90.75	89.8125		
	0.2	0.05	51.75	54.25	94.75	94.5	59.84	64	94.5	94.5	95	64.53	79.75	92.75	92.5	90	52.25	54.75	90.625	88.875	56.05	58.375	86.75	85.625	87.75	59.69	74.25	95.25	94.875	93	60.69	54.75	77.125	60.01	61.625	84.1875	80.8125	83.875	61.72	74.75	93.9375	92.875	92.5	
	0.2	0.025	58.5	52.5	85.5	86.5	56.28	64	84	83.5	81	52.5	79.25	93	93	90.5	57.63	54	92.125	92.375	55.1	58.5	95.25	95.25	95.375	57.34	74.13	95.25	94.625	94.5	59.19	52.625	86.9375	59.82	60.688	87.875	85.1875	86.875	59.65	77.875	85.8125	83.625	81.8125	
	0.2	0.1	67.25	55.25	94.75	94.5	61.9	62.75	95	95	93.5	61.25	80	91.75	91.75	91.25	65.62	56.5	89.375	89.875	61.57	63.5	93	93	92.375	66.87	75.88	96.875	96.75	96.625	62.7	57.75	85.4375	86.1875	65.12	64.813	88.125	84.8125	87.0625	58.52	75.625	91.75	90.8875	90
	0.2	0.2	70.75	61.25	94.25	93.5	66.61	66.75	85.75	81.75	83.5	65.31	77.75	96.75	96.75	97	65.13	58.75	85.75	85.5	65.43	64.875	88	87.625	88.5	69.61	75.5	94.375	94.125	93.125	68.12	60.938	86.125	83.375	67.37	70.125	93.5625	93.0625	92.3125	66.84	78.5	89.875	88.75	86.3125
	0.4	0.01	58	54.5	96.5	95.75	51.8	62.25	98.25	98.25	97.75	47.5	80	96.5	96.5	94.75	52.63	48.625	93	92.625	56.24	60	94.5	94	94.625	49.3	75.38	92.5	91.5	89.625	51.75	50.25	83.5	84.3125	52.32	59.438	92.125	90.6875	91.3125	53.87	75.313	94.375	94.125	93.0625
	0.4	0.05	56.5	53.5	93.75	94.25	56.28	61.5	98.25	98.25	97.75	48.75	79.75	99	99	97.75	58.75	47.875	91.375	92	59.51	61.75	97.375	97.375	96.75	63.44	73	97.75	97.75	95.625	58.25	50.813	94.375	93.8125	60.25	60.938	89.125	86.0625	88.6875	59.45	74.25	94.75	94.1875	93.5625
	0.4	0.025	50.5	52.5	96	95.75	48.78	64.25	92.5	90.25	92.75	52.34	78.5	97.75	97.75	97.5	56.12	48.875	93.125	93.375	55.86	58.25	91.375	91.375	90.75	51.64	77	91.875	91.5	91	55.06	50.813	90.25	89.5	57.52	60.563	92	89.5625	90.3125	54.18	74.813	92.25	91.0625	90.8125
	0.4	0.1	58.5	56.75	98	97.75	69.43	63	98.25	98.25	97.5	63.28	79.5	96.75	96.75	96.25	62.63	54.25	96.375	95.5	64.95	61.25	93.875	93.875	94	62.81	73.13	98.75	98.75	98.125	63.13	52.25	89	88.5625	63.67	59.125	91.25	88.875	91.3125	58.4	74.625	95.875	95.75	95.125
	0.4	0.2	73	57	98	98	69.04	64.25	97.25	97.25	97.25	61.72	79	98	98	97	69.25	53.25	94.25	94.5	69.82	59.75	95.75	96.75	96.52	74.25	96.875	96.75	95.875	68.56	56.875	83.6875	82.375	70.37	64.563	92.625	91.9375	92	69.45	77.563	93.5625	92.8125	91.375	
	0.4	0.3	72.25	58	97.5	96.5	64.72	64.75	98	98	97	71.09	79.25	96.75	96.25	93.75	75.37	54.375	97.875	97.875	73.01	61.375	96.375	95.375	95.375	75.78	76.25	98.375	98.125	97.75	74.75	61.063	93.1875	92.4375	74.92	67.188	93.875	92.8125	93.375	73.75	77.188	95.9375	95.125	94.5625
	0.4	0.4	80.25	65.75	98.75	98	83.11	69.25	98.25	98.25	98	69.69	79.75	95.5	94.75	93	83.38	61.875	96.85	95	80.51	66.125	96.625	96.625	96	76.48	77.63	97.75	97.75	97.65	78.37	68.063	94.5	93.3125	80.69	69.25	95.25	94.125	93.25	80.55	79.125	92.75	89.875	89.6875

**Table 4.4:** Evaluation for four-locus models

	MAF	H	400 Samples												1600 Samples												1600 Samples																		
			1:1 ratio			1:2 ratio			1:4 ratio			1:1 ratio			1:2 ratio			1:4 ratio			1:1 ratio			1:2 ratio			1:4 ratio			1:1 ratio			1:2 ratio			1:4 ratio									
			MDR	AC	MDRAC	Noise	MDR	AC	MDRAC	Adjusted Threshold	MDR	AC	MDRAC	Noise	MDR	AC	MDRAC	Adjusted Threshold	MDR	AC	MDRAC	Noise	MDR	AC	MDRAC	Adjusted Threshold	MDR	AC	MDRAC	Adjusted Threshold	MDR	AC	MDRAC	Adjusted Threshold	MDR	AC	MDRAC	Adjusted Threshold	MDR	AC	MDRAC	Adjusted Threshold			
	0.2	0.01	55.25	55.5	93.25	93.25	49.73	58.75	97	97	96.75	47.97	78	96.75	95.5	96	54	48.125	85.875	85.875	52.58	58.75	91.125	90.75	90.75	50.47	73.88	93.5	93	91.375	52.75	51.75	78	77	49.04	58.563	90.1875	89.9375	89.125	51.21	77.125	88.75	87.875	85.0625	
	0.2	0.05	54.5	51.25	92.75	92.25	56.1	61.5	95.75	94.5	95.5	50.63	79	95.25	95	94.5	59	51.75	84.625	83.375	54.84	58.125	95.375	95.375	95	56.33	75.13	90.875	89.5	89	56.12	51.188	83	82.1875	56.92	59.75	90.6875	89.375	60.43	74.313	91.1875	89.4375	89		
	0.2	0.025	55	52.5	96.5	96.25	52.15	62.25	89.75	89.75	88	52.19	78.25	98	98	96.75	52.25	49.625	90.25	90	56.62	60.625	94.625	94.625	93.625	51.8	74.63	96	95.5	94.625	54.19	52	83	82.6875	57.71	60.625	79.6875	80.6875	54.57	75.625	92.3125	92.25	91.375		
	0.2	0.1	60.5	49.75	96	96.5	56.46	62.25	93.5	93.75	93.5	63.44	79.75	92.25	92	90	63	50.75	92.75	93	61.76	58	92	92	92.125	60.55	73.38	95.5	95.5	95.125	61.19	52.5	87.125	86.8125	60.71	61.938	90.375	88.894375	60.51	75.813	90.1875	88.25	87.125		
	0.4	0.01	51.75	48.5	94.75	95.75	49.53	63	95.75	95.5	95	43.59	78.25	95.5	95.25	94.25	53.37	50.125	98	97.875	57.38	53.375	95.25	94.875	56.24	61.875	97	97	96.25	52.03	74.5	94.625	94.375	60	50.563	93.75	54.1	55.875	90.0625	89.36875	94	52.19	74.313	95.1875	95.375
	0.4	0.05	54.75	55.5	95.25	95	56.1	64.75	97.75	97.75	98	59.06	79.5	99	99	98.75	57.38	53.375	95.25	94.875	56.24	61.875	97	97	96.25	52.03	74.5	94.625	94.375	60	50.563	93.75	54.1	55.875	90.062										

**Table 4.5:** Evaluation for five-locus models

		400 Samples												800 Samples												1600 Samples																	
		1:1 ratio				1:2 ratio				1:4 ratio				1:1 ratio				1:2 ratio				1:4 ratio				1:1 ratio				1:2 ratio				1:4 ratio									
MAF	H	MDR	AC	MDRAC	Noise	MDR	AC	MDRAC	Adjusted	MDRAC	Noise	MDR	AC	MDRAC	Adjusted	MDRAC	Noise	MDR	AC	MDRAC	Adjusted	MDRAC	Noise	MDR	AC	MDRAC	Adjusted	MDRAC	Noise	MDR	AC	MDRAC	Adjusted	MDRAC	Noise								
0.4	0.01	47	55.75	96.25	96.5	50.67	64.5	96.25	96.25	95.75	55.62	79.25	97.75	97.5	97.5	54	48.75	97.75	98.125	50.71	54.375	98.375	98.375	56.95	72.88	98.125	98.125	97	50.31	51.438	87.375	86	46.79	58.313	93.375	92	93.375	53.2	73.313	96	95.625	95.25	
0.4	0.05	54	51.75	98.5	98	48.42	61.5	99	99	48.59	77.75	98.5	98.25	98.5	53.87	50.625	94.75	94.625	56.42	57.75	96.5	96.5	53.83	76	97.5	97.5	96.125	54.62	50.438	92.125	91.5	59.31	59.938	94.8125	94.3125	94.25	48.55	74.688	96.6875	96.4375	96		
0.4	0.025	50.5	53.5	97.25	95.75	48.23	63.5	96	97	97.5	57.19	79.75	99.25	99	99	50.37	49.25	96.125	95.875	56.51	57.25	99	99	98.75	54.69	73.13	94	92.5	51.23	49.125	91.25	91.6875	51.01	59.5	93.625	92.8125	93.375	55.12	74.188	94.5	93.75	93.3125	
0.4	0.1	60.5	57.75	96.75	95.75	58.37	61.75	98	97.75	97.5	58.44	75.5	98.75	99	57	48.875	98.5	98.25	56.14	58.625	89.625	88.125	53.12	73.38	99	98.75	98.375	57.06	50.438	95.5	94.8125	60.43	57.625	93	91.875	92.25	58.4	74.063	96.1875	95.5625	94.9375		

**Table 4.6:** Evaluation for six-locus models

		400 Samples												800 Samples												1600 Samples																	
		1:1 ratio				1:2 ratio				1:4 ratio				1:1 ratio				1:2 ratio				1:4 ratio				1:1 ratio				1:2 ratio				1:4 ratio									
MAF	H	MDR	AC	MDRAC	Noise	MDR	AC	MDRAC	Adjusted	MDRAC	Noise	MDR	AC	MDRAC	Adjusted	MDRAC	Noise	MDR	AC	MDRAC	Adjusted	MDRAC	Noise	MDR	AC	MDRAC	Adjusted	MDRAC	Noise	MDR	AC	MDRAC	Adjusted	MDRAC	Noise								
0.2	0.01	48.5	50.25	91	90	46.17	55.75	88.5	85.75	88.75	47.66	75	95.5	95.25	95.25	44.37	48.75	85.375	85.25	52.49	57.5	84.375	83.625	84.125	51.8	75.08	92.875	91.75	91.75	52.31	48.063	83	83	52.7	61.75	81.125	77.5	79	53.4	78	87.0625	86.75	83.3125
0.2	0.025	47.75	49.75	89.75	90.75	55.93	56.25	91.75	91.25	91.25	55.16	74.5	93.25	93	91.75	49.88	49.5	86.375	86.5	49.49	58.875	88.5	88.5	88.75	48.05	73.5	95.5	95	95.375	51.19	50.563	86.5625	86.6875	49.37	60.5	80.25	76.4975	78.1975	54.77	76.182	89.3125	88.375	86.25
0.4	0.01	49.25	51.75	99	99.25	47.51	56.25	96.75	95.5	96	50.63	78.5	99.5	99.25	96.25	51.25	96.75	96.125	52.01	59.125	94.5	94.625	52.58	74	98	97.625	97.625	50.94	48.625	95.375	91.8125	51.67	57.375	94.5625	94.125	94.5625	54.8	74.313	96.875	96.875	95.8125		
0.4	0.05	50	53	98	98	55.95	56.5	98.75	98.75	98.5	56.41	77.5	98.75	98.25	53.87	53.125	97.5	97.5	53.6	56.625	96.875	96.875	96	57.89	73.25	97.5	97.75	97.25	53.19	51.75	91.125	90.9375	54.2	56.625	97.125	96.75	97.125	52.97	73.313	96.75	96.25	96.125	
0.4	0.025	53.25	56.25	98	96.75	55.94	60.5	99	99	98.75	48.91	75.25	96.75	95.5	52.88	51.625	97.5	96.625	50.98	60.25	98	98	98	97.75	47.03	74.75	98.75	98.75	98.125	53.25	50.688	92	91.75	50.03	60	91	89.5625	90.3125	51.48	74.313	97.3125	97.125	96.125
0.4	0.1	49.75	48.25	98.5	98.75	54.25	58.5	99.75	99.75	99.75	55.62	75.25	98.5	98.5	55.75	52.5	96.625	96.375	56.69	55.375	96.125	96.125	95.25	56.72	72	98.625	98.625	98.75	56	47.563	93.0325	92.625	55.98	55.938	93.5625	92.5	94.125	55.55	73.063	97.1875	97.1875	96	
0.4	0.2	56.5	54.25	98.25	98.25	53.89	56.75	99	99	99	51.72	76	100	100	99.75	58.38	51	97	96.5	61.48	59.5	97.625	97.625	94.625	54.84	74.13	98.75	98.75	98.875	61.12	49.938	98.125	97.5	63.2	57.375	96.75	96.375	95.875	60.08	72.563	98.3125	98	97.5

**Table 4.7:** Average balanced accuracy of one-locus to six-locus models

Sample Size	Ratio	Balanced Accuracy																																			
		Single Locus				Two Locus				Three Locus				Four Locus				Five Locus				Six Locus				MDR	AC	KMeans	PCA								
MAF	H	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA	MDR	AC	KMeans	PCA				
400	1:1	65.9	64.46	55.804	54.4	65	61.48	51.643	52.2	62.2	55.9	51.646	51.8	57.23	52.15	51.475	51.4	53	54.7	51.438	51.9	50.71	51.9	51.857	52.1												
	1:2	66.85	71.66	57.82	55.70	65	68.8	55.768	56	61.7	64.15	55.833	53.8	56.57	62.03	56.425	55.1	51.4	62.8	55.063	52	52.81	57.2	55.536	55.6												
	1:4	65.22	80.79	59.43	60.00	63.2	79.39	60.09	58	59.2	79.29	59.92	60	55.14	78.25	61.1	65	55	78.1	55.75	56	52.3	76	61.71	64												
800	1:1	65.62	61.65	53.87	53.10	66.8	60.41	51.55	51.6	62.8	53.47	51.41	49.8	58.75	50.43	51.39	51.6	53.8	49.4	51.19	52.2	53.05	51.1	51.16	51.4												
	1:2	65.84	66.53	57.53	54.03	66.9	66.02	55.14	55.54	62.6	60.99	55.53	54.41	57.82	59.56	53.91	56.6	54.9	57	54.06	53.72	53.82	58.2	55.93	51.41												
	1:4	64.85	78.63	59.08	57.66	65.5	76.54	59.571	59.21	62.7	75.27	58.656	56.94	58	74.5	60.088	57.4	54.6	73.8	57.156	56.13	52.7	73.9	59.464	57.46												
1600	1:1	65.83	61.64	54.39	55.39	67.4	62	51.085																													

#### **4.4.2 Analysis of real data**

The performance of the proposed method is further evaluated over sporadic breast cancer, and hypertension datasets. Table 4.8 and Table 4.9 show the accuracy of MDRAC over cross validation. The threshold values of sporadic breast cancer and hypertension datasets are set to one, and adjusted to 1.0147 and 2.4077 respectively. The proposed method is evaluated for one-locus to nine-locus models on sporadic breast cancer data, and one-locus to seven-locus models for hypertension data. The five-locus interaction model (Cyp1B1-119, Cyp1B1-432, COMT, Cyp1A1m1, and Cyp1B1-453) is identified as the best model for breast cancer with a prediction accuracy of 75.6098%. The three-locus interaction model (rs5051, rs699, and rs5049) with a prediction accuracy of 72.912% is identified as the best model for hypertension. The observations confirmed that the performance of MDRAC is equally well performed by adjusting threshold values as that on simulated datasets.

Further, the performance of MDRAC is evaluated by adding 10%, 20%, and 30% of noise into the datasets. It is observed that the accuracy of MDRAC slightly dropped as the percentage of noise is increased. The accuracy dropped ranging from 1% to 3% on adding up to 30% of noise into breast cancer data. Similarly, the accuracy dropped by 3%, 5%, and 10% by adding 10%, 20%, and 30% of noise respectively in hypertension data. The method is further evaluated by splitting the data with 80% for training to identify the best model, and 20% for validation to evaluate the accuracy. This is performed to identify the more accurate models. The results are tabulated in Table 4.10 and Table 4.11 respectively for breast cancer and hypertension data. The accuracy for MDRAC is improved by 3%. However, the accuracy dropped by 3%, and 7% for models with the adjusted threshold values, and the addition of noise into breast cancer data. Further, the accuracy dropped by 2%, 1%, and 5% for the models with and without adjusting the threshold values, and the addition of noise into hypertension data.

Figures from 4.9 to 4.17 are illustrated using MDR tool [17, 109]. Figure 4.9 and Figure 4.10 shows the graphical representation of cell values of best model obtained from MDR analysis over sporadic breast cancer, and hypertension data respectively. MDR analysis identifies the five-locus genotype combinations associated for manifestation of both breast cancer and hypertension.

**Table 4.8:** Experimental results of sporadic breast cancer data on cross-validation

Algorithms	Best Model	No. of Loci	Accuracy	CVC
MDRAC	Cyp1B1-119, Cyp1B1-432, COMT, Cyp1A1m1, Cyp1B1-453	5	75.6098	10/10
MDRAC with adjusted threshold	Cyp1B1-119,CypIBI-453,CypIBI-432,Cyp1B1-48,CypIAIm2,CypIA1m1,COMT,GSTT1	8	75.122	10/10
MDRAC in presence of noise (10%)	COMT, CypIBI-453, CypIA1m1,CypIBI-432,CypIAIm4	5	74.3902	10/10
MDRAC with adjusted threshold in presence of noise (10%)	Cyp1B1-119,Cyp1B1-48,CypIAIm4,CypIBI-432,CypIA1m1,COMT,CypIBI-453,GSTM1,GSTT1	9	75.3659	10/10
MDRAC in presence of noise (20%)	CypIAIm4,COMT,CypIBI-453,CypIA1m1,Cyp1B1-48,CypIBI-432	6	72.6829	10/10
MDRAC with adjusted threshold in presence of noise (20%)	CypIBI-453,CypIAIm4,COMT,GSTM1,GSTT1	5	76.8293	10/10
MDRAC in presence of noise (30%)	CypIAIm4,Cyp1B1-119,GSTT1,COMT,CypIBI-453,CypIBI-432	6	75.8537	10/10
MDRAC with adjusted threshold in presence of noise (30%)	CypIA1m1,CypIAIm4,CypIBI-432,Cyp1B1-119,COMT,GSTT1	6	73.9024	10/10

**Table 4.9:** Experimental results of hypertension data on cross-validation

Algorithms	Best Model	No. of Loci	Accuracy	CVC
MDRAC	rs5051,rs699,rs5049	3	72.912	10/10
MDRAC with adjusted threshold	rs699,rs5186	2	73.1377	10/10
MDRAC in presence of noise (10%)	rs5050,rs5186,rs4646994,rs5051	4	69.9774	10/10
MDRAC in presence of noise with adjusted threshold (10%)	rs5050,rs4646994,rs5186	3	70.2032	10/10
MDRAC in presence of noise (20%)	rs4646994,rs5186,rs11568020,rs5051	4	67.9458	10/10
MDRAC in presence of noise with adjusted threshold (20%)	rs4762 ,rs5050,rs11568020,rs5049,rs5051	5	68.3979	10/10

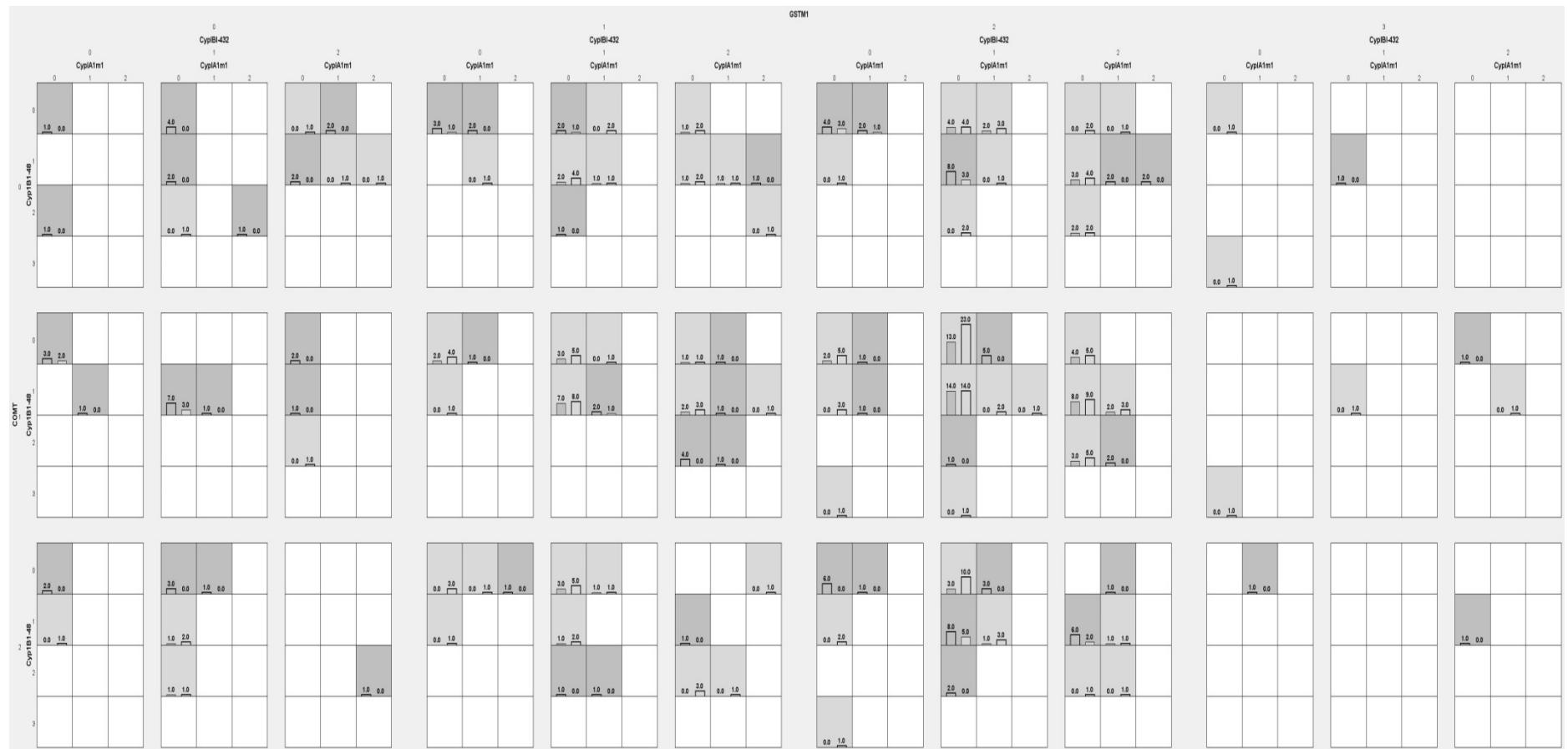
MDRAC in presence of noise (30%)	rs4762,rs4646994,rs5186,rs11568020,rs5051	5	63.4312	10/10
MDRAC with adjusted threshold in presence of noise (30%)	rs4646994,rs4762,rs5051,rs11568020	4	64.1084	10/10

**Table 4.10:** Experimental results of sporadic breast cancer data

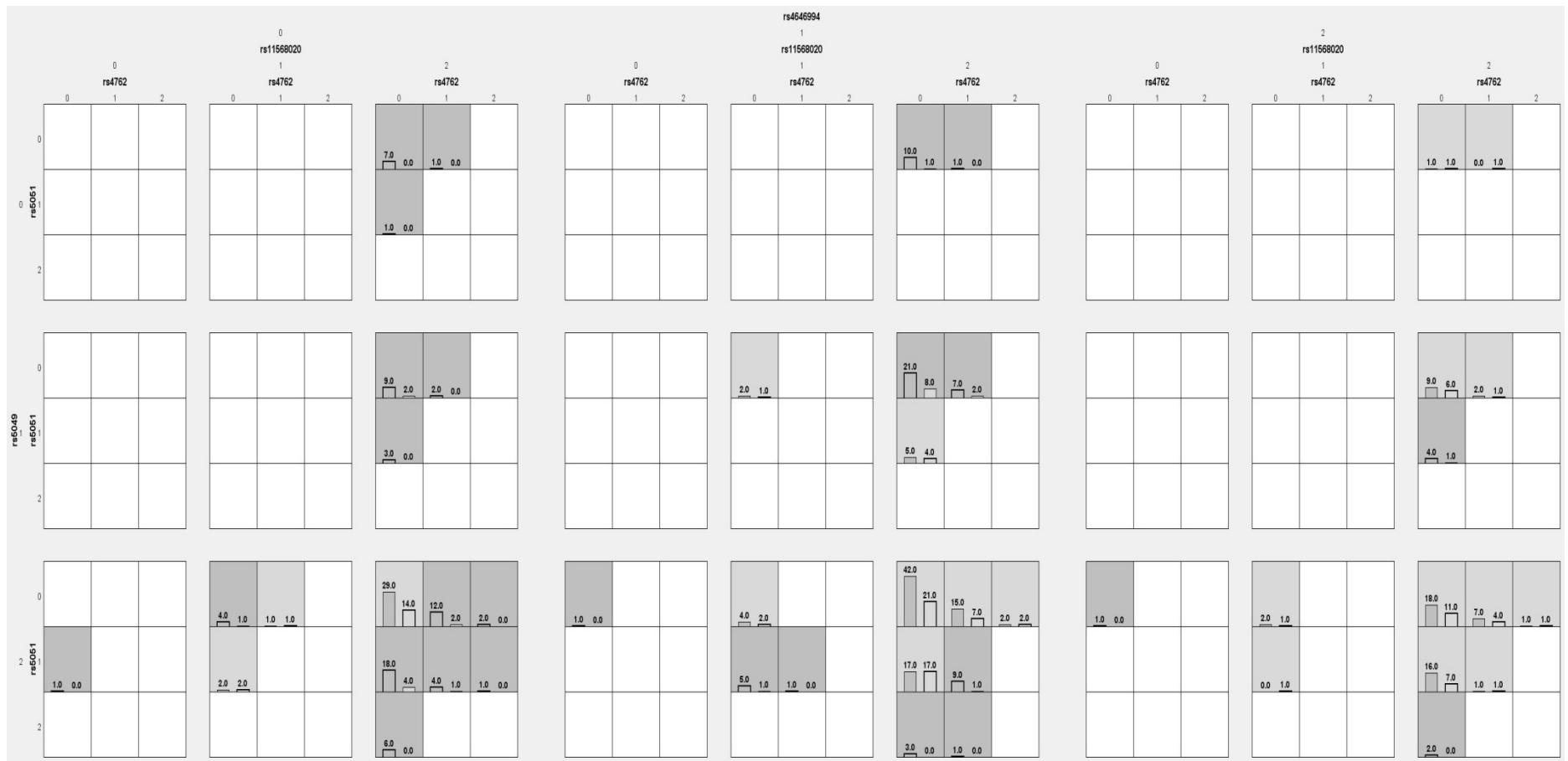
Algorithms	Best Model	No. of Loci	Accuracy
MDRAC	Cyp1B1-119,CypIBI-432,COMT,CypIA1m1,CypIBI-453	5	78.0488
MDRAC with adjusted threshold	Cyp1B1-119,CypIBI-453,CypIBI-432,Cyp1B1-48,CypIA1m2,CypIA1m1,COMT,GSTT1	8	71.9512
MDRAC in presence of noise	COMT, CypIBI-453, CypIA1m1,CypIBI-432,CypIA1m4	5	67.0732
MDRAC with adjusted threshold in presence of noise	Cyp1B1-119,Cyp1B1-48,CypIA1m4,CypIBI-432,CypIA1m1,COMT,CypIBI-453,GSTM1,GSTT1	9	74.3902

**Table 4.11:** Experimental results of hypertension data

Algorithms	Best Model	No. of Loci	Accuracy
MDRAC	rs5051,rs699,rs5049	3	70.7865
MDRAC with adjusted threshold	rs699,rs5186	2	71.9101
MDRAC in presence of noise	rs5050,rs5186,rs4646994,rs5051	4	65.1685
MDRAC in presence of noise with adjusted threshold	rs5050,rs4646994,rs5186	3	64.0449

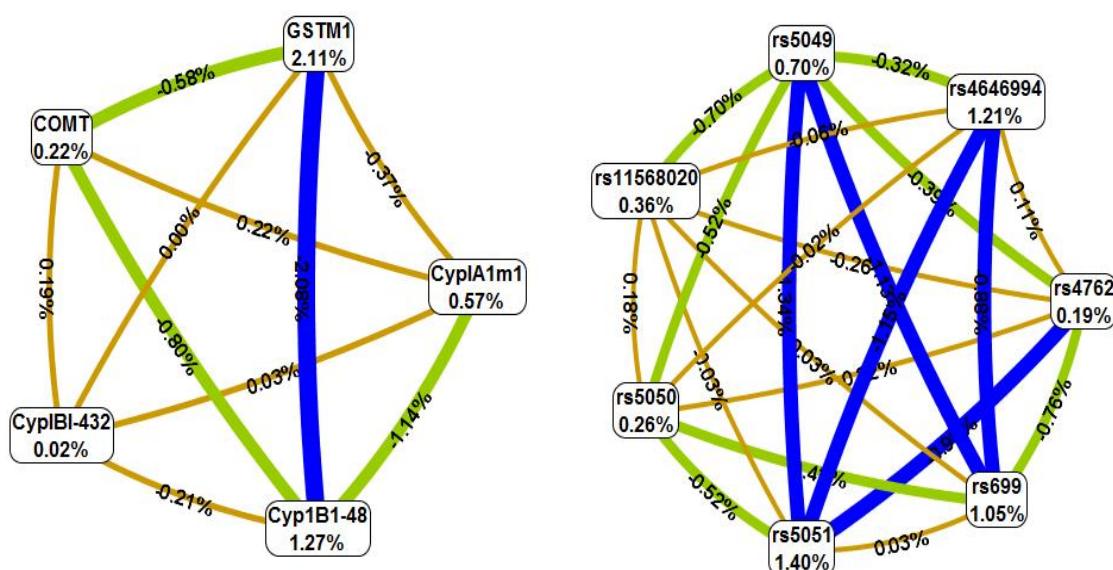


**Figure 4.9** Graphical representation of cell values of five-locus genotype combinations of MDR analysis over sporadic breast cancer data.



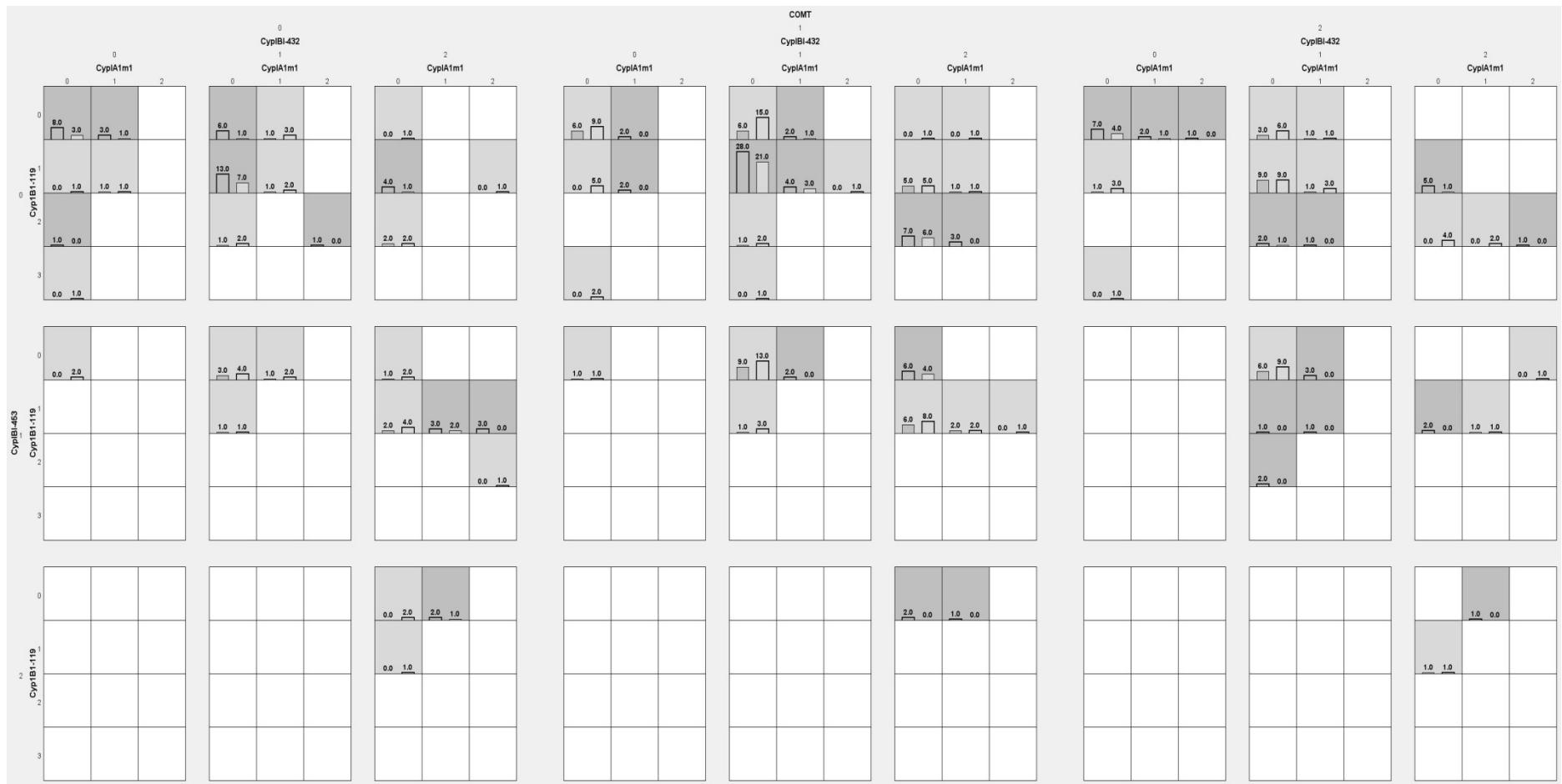
**Figure 4.10** Graphical representation of cell values of five-locus genotype combinations of MDR analysis over hypertension data.

The best five-locus model identified for sporadic breast cancer disease is Cyp1A1m1, Cyp1B1-48, Cyp1B1-432, COMT, and GSTM1. Similarly, the best model identified by MDR analysis for hypertension in human is rs4762, rs5051, rs11568020, rs5049, and rs4646994. High risk cells are represented in dark gray colour, low risk cells in light gray colour, and empty cells in white colour for each genotype combination. The patterns of each cell differ in each different multi-locus dimension. This provides evidence of interactions between polymorphisms of different genes. That is, a SNP at a locus may influence the association of a disease by depending on the genotypes at the other four loci.

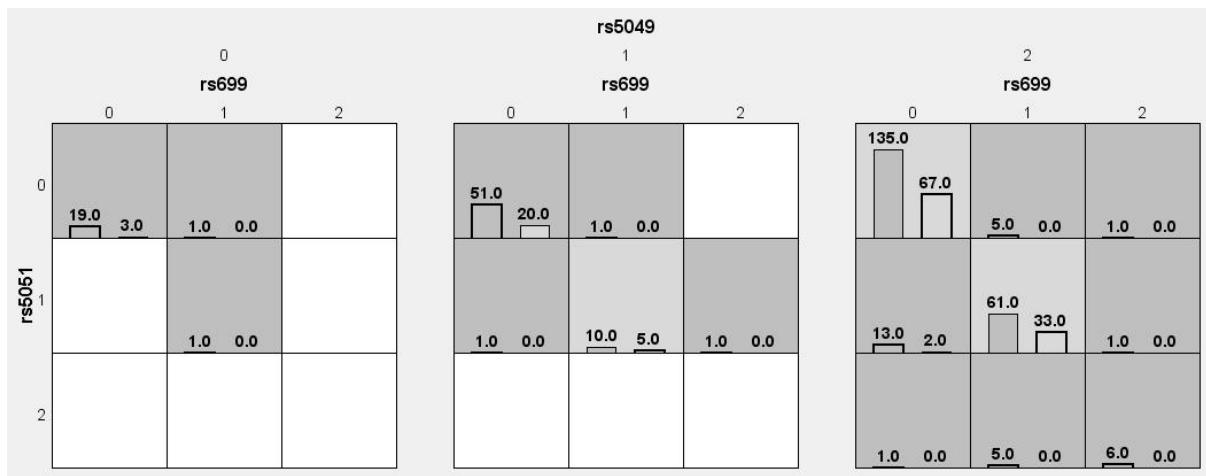


**Figure 4.11** Entropy circular graph of five-locus genotype combinations of MDR analysis over sporadic breast cancer data and hypertension data.

The circle graphical representation of interactions between five SNPs, and seven SNPs for MDR analysis on breast cancer, and hypertension data is illustrated in Figure 4.11 with an entropy graph. These graphical representations are described in percentage of entropy in case-control based samples. The interactions between SNPs are represented by a line with a positive percentage of entropy. Redundancy is represented with a negative percentage of entropy. The positive values on the nodes represent the independent main effects of SNPs. Tan lines indicate independence, blue lines represents a strong correlation, and green lines represents a weak correlation between the SNPs.

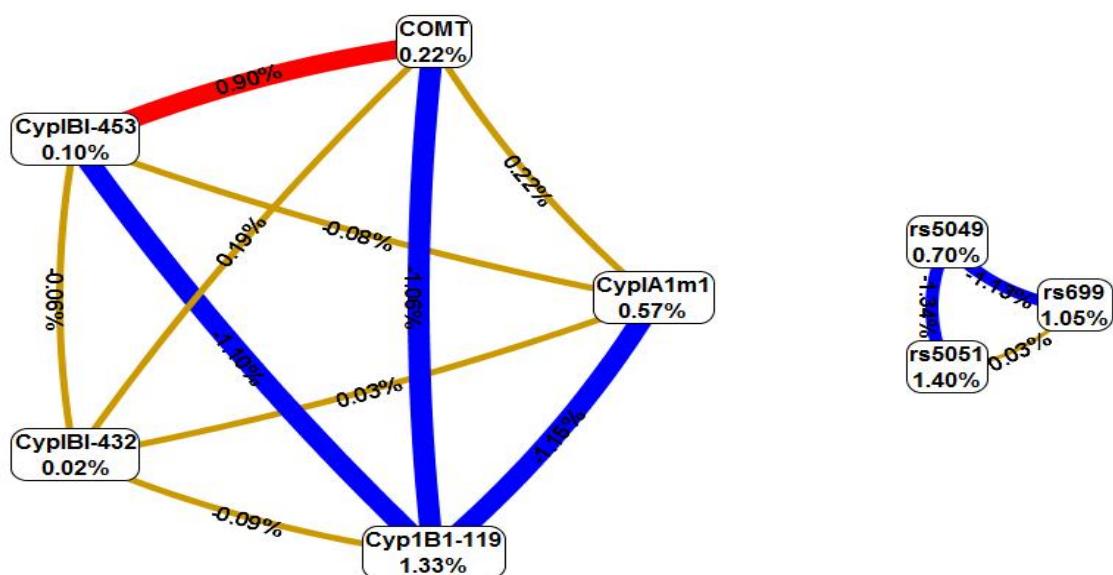


**Figure 4.12** Graphical representation of cell values of five-locus genotype combinations of MDRAC analysis over sporadic breast cancer data.



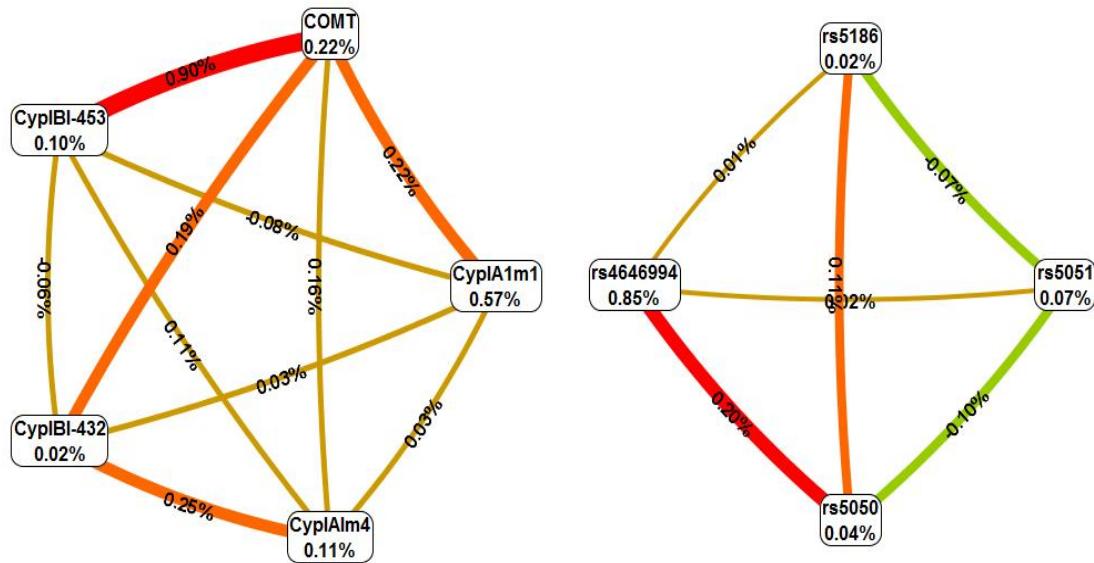
**Figure 4.13** Graphical representation of cell values of three-locus genotype combinations of MDRAC analysis over hypertension data.

Figure 4.12 and Figure 4.13 illustrates the graphical representation of cell values of the best models (based on [17]) identified by MDRAC over sporadic breast cancer, and hypertension data. The five-locus genotype combinations of MDRAC analysis shows the high risk (dark gray shaded), low risk (light gray shaded), and empty cells (white) with the corresponding distribution of cases (left bars), and controls (right bars). It is observed that each SNP at a locus interacts with other two SNPs at different loci. The circular graphical entropy illustration of five-locus and three-locus interaction models obtained from MDRAC analysis on sporadic breast cancer, and hypertension data are visualized in Figure 4.14. The red line represents weak epistasis between SNPs.

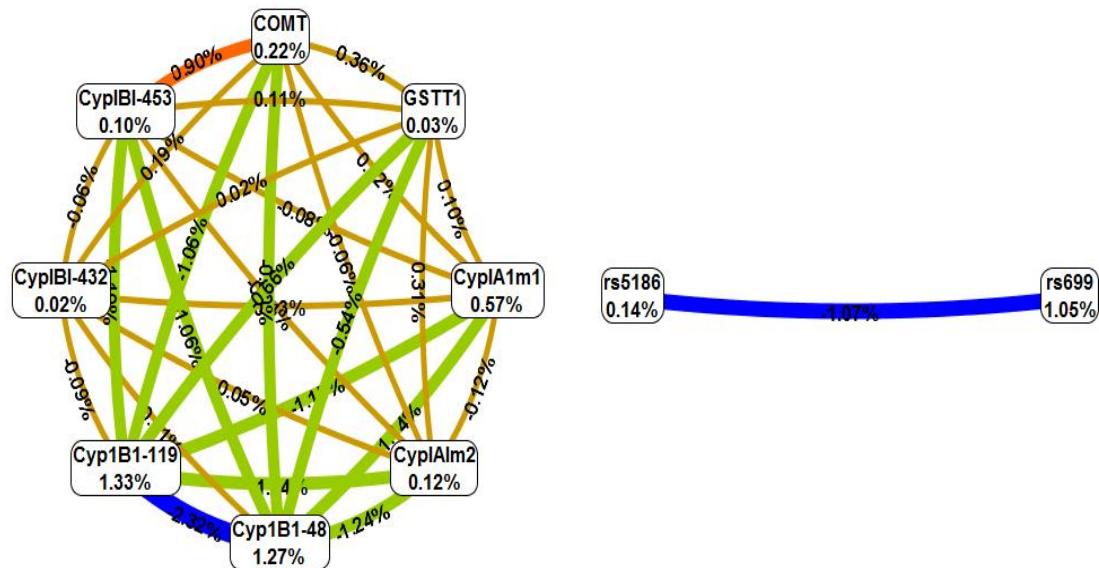


**Figure 4.14** Entropy graph of MDRAC analysis over sporadic breast cancer and hypertension data.

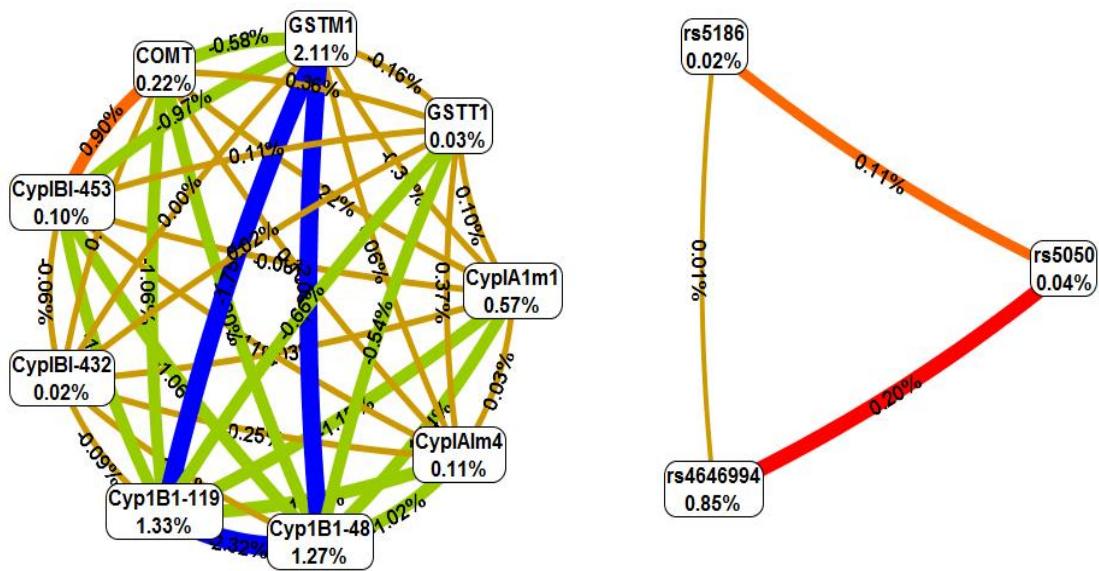
Figure 4.15 is the visual representation of MDRAC analysis by adding noise into breast cancer, and hypertension datasets. Orange lines denote a positive moderate information gain in the presence of main effect. Similarly, Figure 4.16 and Figure 4.17 are the entropy graphical representations of the MDRAC analysis by adjusting threshold values in the absence, and the presence of noise on sporadic breast cancer and hypertension data respectively.



**Figure 4.15** Entropy graph of MDRAC analysis in the presence of noise over breast cancer and hypertension data.



**Figure 4.16** Entropy graph of MDRAC analysis by adjusting threshold value over breast cancer and hypertension data.



**Figure 4.17** Entropy graph of MDRAC analysis by adjusting threshold value in the presence of noise over breast cancer and hypertension data.

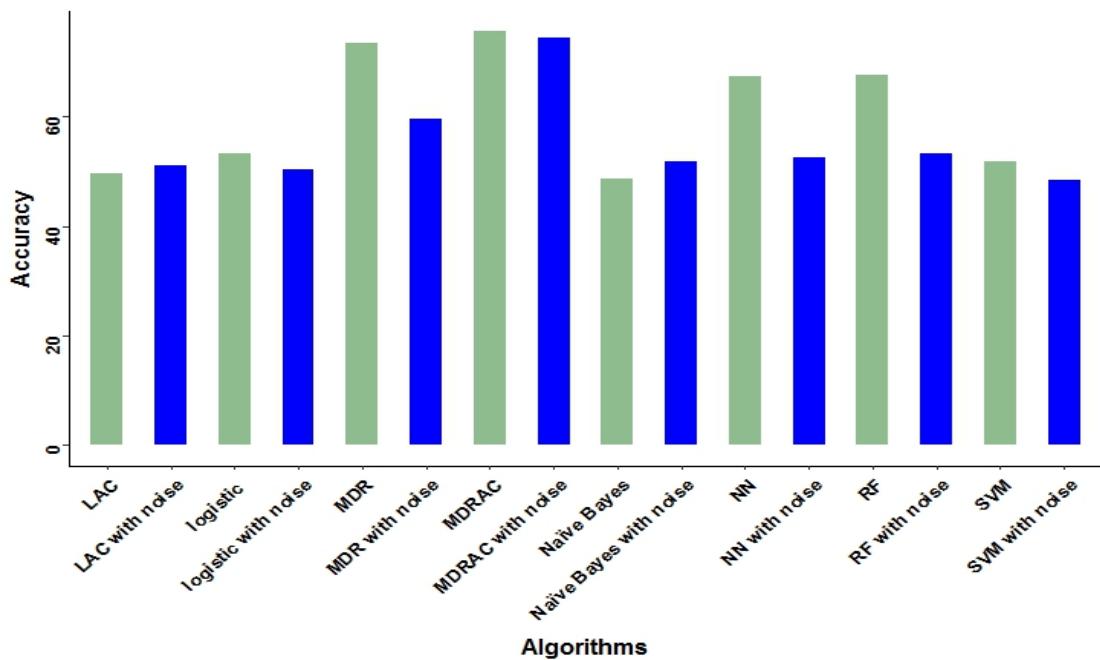
Table 4.12 summarizes the accuracy of MDRAC over the existing approaches, such as, logistic, MDR, Naïve Bayes, RF, LAC, SVM, and NN, in the absence and the presence of noise. It is observed that the best model identified by MDRAC analysis for both sporadic breast cancer, and hypertension datasets has the highest accuracy over other approaches. Compared to the other approaches, MDRAC also performed well with the highest accuracy of 73%, and 70% in the presence of noise.

**Table 4.12:** Accuracy of MDRAC over previous algorithms

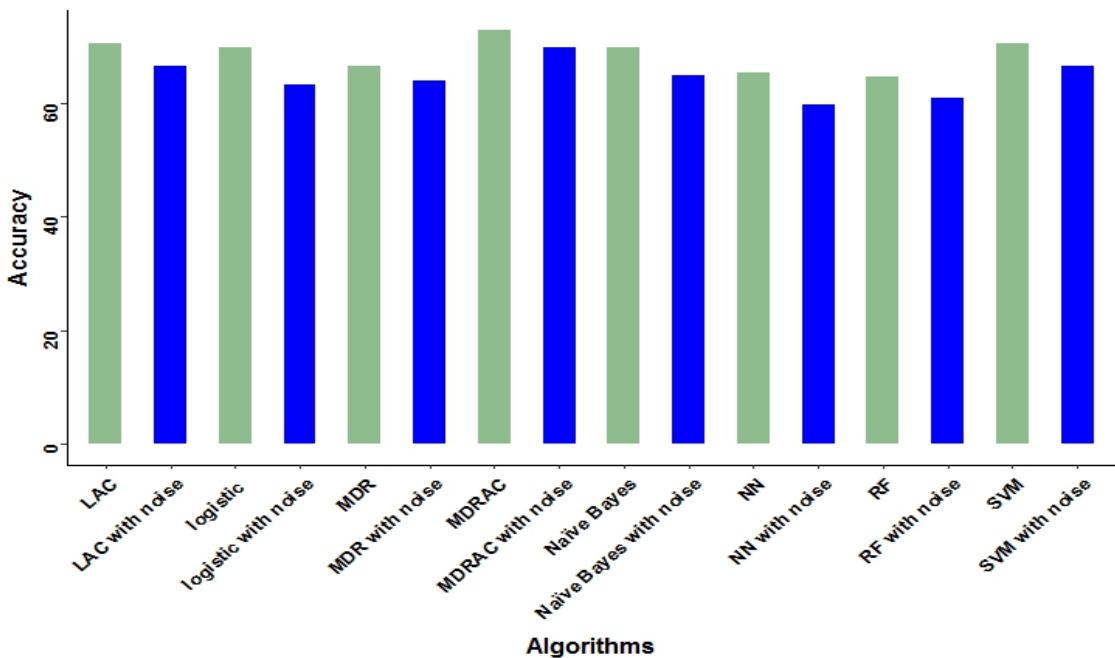
Algorithms	Breast cancer	Hypertension
MDR	73.55	66.71
MDRAC	75.6098	72.912
RF	67.561	64.7856
NN	67.317	65.4628
SVM	51.7073	70.6546
logistic	53.1707	69.9774
Naïve Bayes	48.5366	69.9774
LAC	49.5122	70.6546
MDR with noise	59.5	64.09
MDRAC with noise	74.3902	69.9774
RF with noise	53.1707	60.9481

NN with noise	52.439	59.8194
SVM with noise	48.2927	66.5914
logistic with noise	50.2439	63.4312
Naïve Bayes with noise	51.7073	65.0113
LAC with noise	50.9756	66.5914

The corresponding bar charts are illustrated in Figure 4.18 and Figure 4.19 in the absence and the presence of noise for breast cancer, and hypertension data analyses respectively. The statistical significance of the best models is evaluated by 1,000 fold permutation test, whose p-value is less than 0.05 ( $p < 0.05$ ). Hence, it is evident that the interactions between SNPs identified by MDRAC analysis have better prediction ability than other existing approaches.



**Figure 4.18** Accuracy of MDRAC compared with other approaches over sporadic breast cancer data.



**Figure 4.19** Accuracy of MDRAC compared with other approaches over hypertension data.

#### 4.4.3 Discussion

Studies for identifying SNP interactions associated with the complex diseases in humans have become increasingly significant. However, restricted regions in human genome with less number of SNPs are focused in these studies due to the computational limitations, and biomolecular complexities [114]. Flexible approaches, and software tools for detecting and interpreting epistasis play an important role in exposing the architecture behind the disease manifestation [19]. Currently, MDR is the most popular method that reduces multi-factor dimensionality to a single dimension. It combines attribute selection and a constructive method to form an iterative process, so that MDR can also be used for interleaving or preprocessing [108]. Interleaving is a process in which, the newly constructed attributes are updated to the existing dataset. It is more useful in the studies that involves more than 300,000 SNPs [302]. Traditionally, it is proven that AC generates more relevant rules that are associated with a disease by improving the classification accuracy [276]. Since AC uses  $k$  rules to determine a test example, it has much higher accuracy than the Naïve Bayes classifier that is traditionally used in MDR [280]. This motivated the proposed hybrid MDRAC to detect the highly associated interactions between SNPs for the detection and characterization of various factors involved in the interaction effects responsible for diseases. The approach

generates a single dimensional attribute with high or low risk factors using MDR, and CPAR is used for the classification. The experimental results of the proposed hybrid approach show the improved accuracy of the method by bridging the gap between statistical, and biological epistasis.

The applicability of the proposed approach was first demonstrated by using simulated datasets. GAMETES is a fast and flexible tool used to generate complex  $n$ -locus simulated models with random architecture [221]. One-locus to six-locus models were generated for different penetrance values with various heritability, and minor allele frequencies. However, it has a limited ability for simulating models with higher heritability values. For certain values of prevalence and heritability, penetrance tables with low probabilities are generated [221]. In instances such as, when heritability  $H \geq 0.1$ , the models in five and six loci models are not able to generate for more than 100,000 iterations. The models are also not able to simulate for  $H = 1$ , and prevalence  $p = 0.25$ . In contrast,  $n$ - locus models are generated for all values of  $H \leq 1$ , and MAF = 0.5. Hence, only 12, 10, 5, and 7 models are generated for three, four, five, six, and seven loci models. In spite of the above mentioned limitations, the GAMETES tool is still used in this research to simulate one to six loci models for its ability to generate pure, and strictly epistasis models [108, 221]. Moreover, it considers complex interacting multi-locus effects in generating the epistasis model. The simulated results significantly improved the prediction accuracy by reducing the classification errors both in balanced and imbalanced case-control data. On successful implementation of the approach over simulated data, it is applied over sporadic breast cancer, and hypertension data.

As reported by Ritchie in his research [17], the major known risk factor of breast cancer is associated with cumulative exposure to estrogen. Enhanced cell proliferation increases cell division and may cause mutation. The mutated polymorphisms interact in enzymatic pathways responsible for the metabolism of estrogen in breast tissues that confer a high risk of breast cancer. A single gene in the pathway of estrogen metabolism is responsible for the cause of familial breast cancer. Familial genes such as BRCA1 and BRCA2 cause less than 20% susceptibility of the disease [17]. These results have been inconsistent with and in contradiction to some of the studies, which have found interactions between SNPs of estrogen metabolism that can increase the risk of sporadic breast cancer [303]. MDR and the proposed approach were applied to ten SNPs that

occurred in five genes COMT, CYP1A1, CYP1B1, GSTM1, and GSTT1 [17]. The ability of these enzymes is to detoxify intracellular products of oxidative reactions. These enzymes metabolize estrogen while metabolizing xenobiotic (harmful exogenous compounds found in food and environmental pollutants) [303]. The reduced ability of detoxifying xenobiotic and the burden of toxic estrogen metabolites increases the risk of breast cancer in women. None of these five estrogen metabolism genes contribute to breast cancer risk on their own. However, the interactions between these genes increase the risk of breast cancer from a 2.7 to 13 fold indicator of risk, which is remarkably strong [17, 303].

In MDR analysis, the five-locus interaction model between Cyp1A1m1, Cyp1B1-48, Cyp1B1-432, COMT and GSTM1 has been identified for the association of sporadic breast cancer. The analysis of MDR based AC method on breast cancer data identified the five-locus interaction model Cyp1A1m1, Cyp1B1-119, Cyp1B1-432, Cyp1B1-453, and COMT as associated with the disease. The model detected two novel SNPs (Cyp1B1-119 and Cyp1B1-432) susceptible to higher order gene-gene interactions in sporadic breast cancer. These SNPs have not been reported in previous studies such as MDR. The results of MDR exhibit a strong association between GSTM1 and Cyp1B1-48. However, the results of the proposed approach showed that the SNP Cyp1B1-119 was strongly associated with COMT, Cyp1A1m1, and Cyp1B1-453.

The proposed approach provided stronger information regarding the likely SNP-SNP interactions responsible for the disease. The results also demonstrated that the SNPs at a locus with weak main effects also showed strong interaction effects with other SNPs at different loci. For example, in the breast cancer data analysis over MDRAC, Cyp1B1-453 has a strong association with Cyp1B1-119 and a weak association with Cyp1A1m1 and Cyp1B1-432 [41]. Cyp1B1-119 has a strong association with Cyp1B1-453, COMT, Cy1A1m1 and a weak association with Cyp1B1-119. COMT is strongly associated with Cyp1B1-119 and weakly associated with Cyp1B1-432 and Cyp1A1m1. Cyp1A1m1 is strongly associated with Cyp1B1-119 and weakly associated with Cyp1B1-453, COMT, and Cyp1B1-432. Cyp1B1-432 has weak interactions with Cyp1B1-453 and Cyp1B1-119, which leads to a strong interaction between Cyp1B1-453 and Cyp1B1-119. The rules generated from the proposed method demonstrate various interesting, and non-linear relationships between SNPs. Hence, the results obtained from these studies are

shown to be more accurate than previous approaches. The high-dimensional computational problem remains the same with the new approach. The proposed approach exhaustively searches all the combinations and can be computationally intensive when more than ten SNPs have to be evaluated. The major problem of this approach is to calculate the prediction errors of the empty cells. The empty cells generated during training data were more than the cells produced by MDR. These cells were left out during the estimation of prediction errors. Hence, the proposed approach has no clear prediction ability for the empty cells.

## **4.5 Evaluation and discussion of MDRAC in the presence of genotyping error, missing data, phenocopy, and genetic heterogeneity.**

The performance of MDRAC in the presence genotyping error (GE), missing data (MS), phenocopy (PC), and genetic heterogeneity (GH) is unknown. Hence, several experiments are further evaluated over simulated datasets, and a real dataset to evaluate the power of MDRAC over MDR. This study in this section is based on the previous study performed over MDR in the presence of noise [36]. Table 4.13 summarises the accuracy of both methods in the presence of GE, MS, PC, GH, and their combinations respectively. The average prediction accuracy across 160 datasets for each epistasis models is evaluated to validate MDRAC over the original MDR method. The results are also evaluated for both methods in absence of noise. Overall, the results show that MDRAC performed better than MDR in all simulated datasets. The classification accuracy of MDRAC is about 33% to 40% higher than MDR in first four epistasis models (Model 1 to Model 4), and about 22% higher than MDR in last two epistasis models (Model 5 and Model 6). This indicates that MDRAC effectively eliminates the prediction errors by improving the prediction accuracy.

The power of both models are summarised in Table 4.14. The power of MDRAC is the estimation of number of times the method correctly identifies the functional SNPs. The power across 160 datasets for each epistasis models is validated on MDRAC, and compared with MDR method. The power is also determined for both methods without noise. The power of both the methods is high (almost 90 to 100%) in the presence of 5%

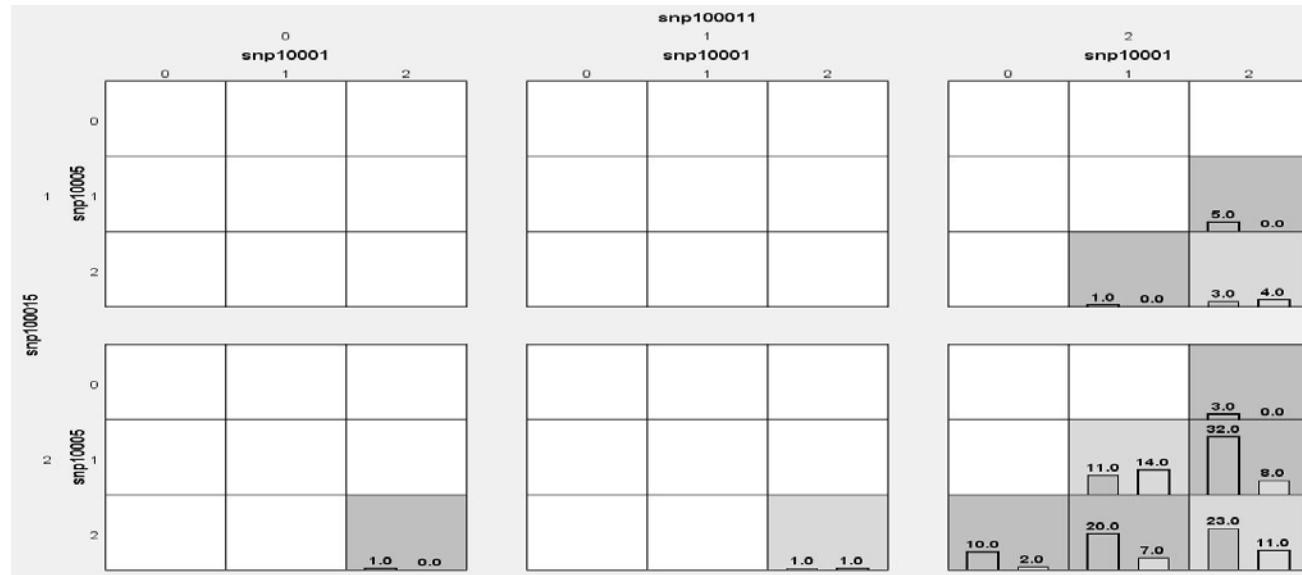
**Table 4.13:** Accuracy of MDRAC to detect two-locus SNP interactions

Sources of Noise	Model 1		Model 2		Model 3		Model 4		Model 5		Model 6	
	MDR	MDRAC										
No Error	75.95	99.475	82.325	99.65	64.8	96.625	63.3	96.55	57.875	75.25	59.375	75.5
GE	76.477	99.6	82.875	99.5	63.9	96.475	64.325	95.4	57.2	76.35	60.2	76.35
GE-GH	62.188	99.449	63.7	99.225	54.225	96.475	56.175	95.975	53.85	74.525	51.5	75.25
GE-GH-MS	62.619	99.654	65.447	99.28931	55.106	96.73684	56.262	96.31579	54.028	74.68422	52.078	75.36842
GE-GH-PC	51.6	99.55	53.8	99.475	52.05	95.525	50.25	96.4	49.825	74.625	49.3	74.175
GE-GH-PC-MS	50.289	99.57857	54.157	99.52631	51.973	96.60523	50.947	96.02631	51.052	74.42105	49.975	74.84211
GE-MS	76.764	99.57893	82.843	99.57892	63.553	96.55265	64.158	95.60528	57.501	76.23686	58.185	77.26314
GE-PC	62.05	99.725	65.35	99.5	52.825	96.375	52.375	96.15	52.6	75.225	52.725	75.7
GE-PC-MS	61.868	99.55262	63.184	99.55263	52.421	96.42782	51.079	96.89473	51.158	75.63158	51.159	76.07893
GH	61.3	99.625	63.7	99.225	55.158	96.375	56.175	95.975	53.85	74.525	51.8	75.725
GH-MS	60.79	99.63157	64.447	99.21052	54.947	96.60524	57.501	96.15789	53.028	74.57895	52.027	75.15789
GH-PC	52.925	99.475	53.95	99.525	48.35	95.85	49.675	96.875	51.225	75.725	51.8	75.975
GH-PC-MS	52.765	99.3421	51.919	99.55261	50.527	96.28947	49.92	96.44738	50.342	75.65788	50	75.81579
MS	76.211	99.55264	82.921	99.57893	63.657	96.42105	64.21	95.78946	56.634	76.18421	60.579	76.28947
PC	61.275	99.575	66.275	99.425	54.125	96.55	50.575	95.675	49.475	74.325	48.8	74.725
PC-MS	62.05	99.60524	64.658	99.55262	51.896	96.44736	51.185	96.42105	48.606	75.55263	50.869	75.28946

**Table 4.14:** Power of MDRAC to detect two-locus SNP interactions

Sources of Noise	Model 1		Model 2		Model 3		Model 4		Model 5		Model 6	
	MDR	MDRAC										
No Error	10	10	10	10	10	10	9	10	10	10	10	10
GE	10	10	10	10	10	10	10	10	9	9	10	10
GE-GH	5	8	7	9	5	6	6	6	1	3	4	5
GE-GH-MS	2	6	7	9	4	8	7	7	1	1	4	6

GE-GH-PC	4	7	4	7	0	0	0	1	0	1	0	0
GE-GH-PC-MS	2	6	1	5	0	1	2	3	0	0	0	0
GE-MS	10	10	10	10	10	10	10	10	8	9	9	9
GE-PC	10	10	10	10	8	8	6	6	6	7	5	5
GE-PC-MS	10	10	10	10	4	5	5	5	3	4	5	5
GH	3	8	7	9	5	6	6	6	1	3	2	4
GH-MS	2	8	7	10	3	4	5	7	0	2	5	5
GH-PC	3	4	3	4	2	3	1	3	1	2	0	0
GH-PC-MS	2	4	3	3	1	2	1	4	1	3	1	2
MS	10	10	10	10	10	10	10	10	9	9	9	9
PC	10	10	10	10	5	7	6	6	1	3	3	5
PC-MS	10	10	10	10	5	7	7	8	0	1	1	5



**Figure 4.20** Graphical representation of cell values of four-locus genotype combinations of MDRAC analysis over whole genome association data.

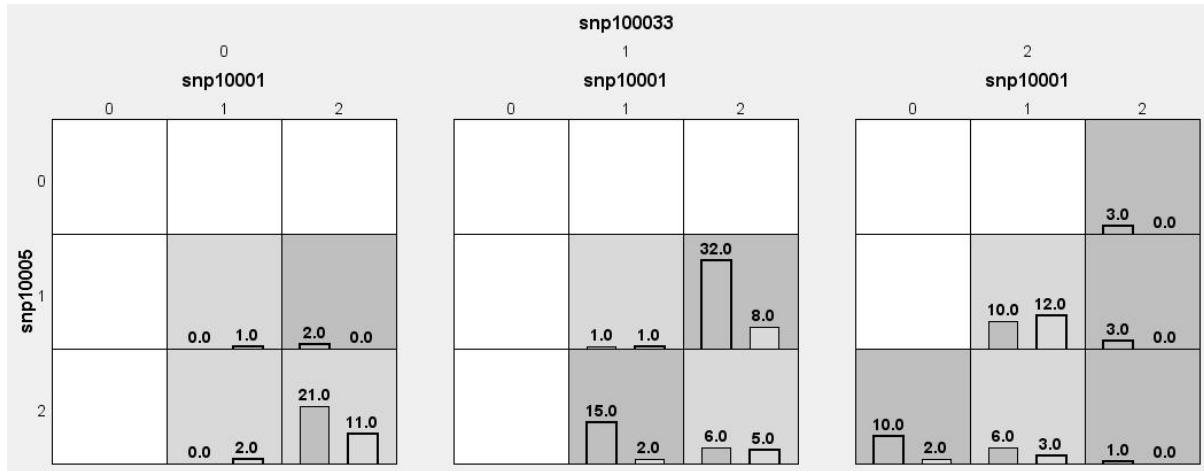
missing data, and 5% genotyping error. The accuracy of MDRAC in detecting the correct functional SNPs is almost 100% in all six epistasis models in absence of noise. The performance of combination of genotyping error and missing data varies between 80% to 100%. The power is 100% for model 1 and model 2 in the presence of 50% phenocopy in the datasets. The power decreases by 30%, 40%, 70%, and 50% for models 3 – 6 respectively. In the presence of 50% genetic heterogeneity, power falls to 80%, 90%, 60%, 60%, 30%, and 40% for models 1 – 6 respectively. The greatest impact of power in all the models is due to the presence of combined effect of phenocopy and genetic heterogeneity. The power fell to 0% in model 6. The combined effect of phenocopy with other sources of noise (missing data and genotyping) is 100% for model 1 and model 2. However, power is affected for other models. The combined effect of genetic heterogeneity with other sources of noise (missing data and genotyping) has less impact on power, except in model 2. The combination of all four sources of noise has greatest impact on power. None of the datasets of model 5 and model 6 identified functional SNPs. Overall results suggested that genetic heterogeneity, and their combined effects with other sources of noise has greatest impact on power. However, MDRAC performed comparatively better than MDR for all six epistasis models in the presence of noise.

**Table 4.15:** Best model predicted for the data obtained from the whole genome association study.

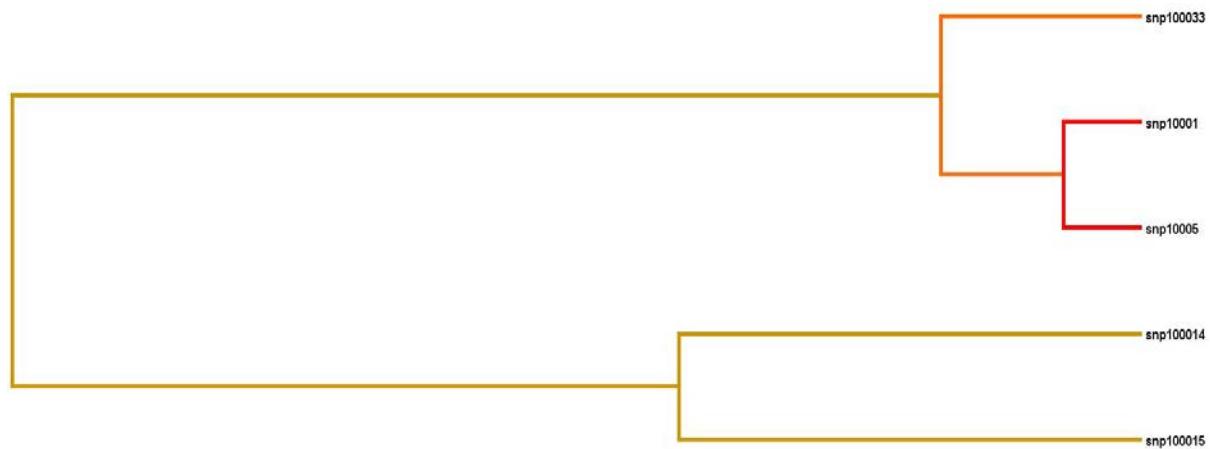
Methods	Best Model	Accuracy	CVC
MDR	snp10001,.snp10005,snp100033	57.53	8/10
MDRAC	snp100011, snp100015, snp10001, snp10005	75.1592	10/10

The prediction accuracy of MDR and MDRAC methods are summarised in Table 4.15. Both the methods are evaluated for two-locus to nine-locus interaction models on a real dataset, obtained from the whole genome association study [298]. Three-locus model is identified by MDR with highest accuracy of 57.53% and highest CVC of 8/10. MDRAC identified four-locus model as the best model with highest prediction accuracy (75.16%) and CVC (10/10). The accuracy of the model is about 18% higher than the model identified by MDR method. The identified four-locus model is statistically significant, whose p value is less than 0.05 ( $p<0.05$ ). Hence, it is suggested that the

interaction between four SNPs contribute to the association of the disease. That is, SNP at a particular locus may depend on the genotype of the other three loci to influence the disease. Figure 4.20 and Figure 4.21 illustrates the graphical cell value representation (using MDR tool [17, 109]) of MDRAC and MDR analysis over data obtained from whole genome association study published in SNPAssoc [297, 298].



**Figure 4.21** Graphical representation of cell values of three-locus genotype combinations of MDR analysis over whole genome association data.



**Figure 4.22** Dendrogram diagram of three-locus interaction model of MDR analysis over whole genome association data.

The dendrogram representation of interactions between SNPs is illustrated (using MDR tool [17, 109]) in Figure 4.22, and Figure 4.23. Figure 4.22 illustrates the dendrogram diagram of MDR analysis and Figure 4.23 illustrates the dendrogram diagram of MDRAC analysis. The SNPs with smaller distance have stronger interaction effect [17, 302]. Red and Orange coloured lines represent synergistic relationship between SNPs. Yellow

indicates independent effect. Green represents redundancy or correlations between SNPs.



**Figure.4.23** Dendrogram diagram of three-locus interaction model of MDRAC analysis over whole genome association data

This study demonstrated the improved accuracy and excellent power to detect SNP interactions in the presence of common sources of noise due to GE, MS, PC, and GH. The results showed the power of MDRAC is 90% – 100% in the presence of 5% genotyping error and 5% missing data, or their combinations. The power of MDRAC is affected in the presence of 50% phenocopy for models 3 – 6, but remains 100% for models 1 and 2. This is due to the simplest nature of these epistasis models (model 1 and 2) that do not exhibit independent main effects [36]. Model 1 is an example of XOR function, which has four high-risk cells [304]. Each single genotype is related to high-risk in the presence of two genotypes from the other locus. Model 2 has three high-risk cells diagonally. Each single genotype is related to high risk in the presence of single genotype from the other locus [293]. The power of MDRAC is greatly affected (70% fall for model 5) in the presence of 50% genetic heterogeneity for all epistasis models. It is also observed that the power of MDRAC is reduced for model 5 and 6 due to low minor allele frequency ( $p=0.1$ ) compared to the other epistasis models.

The power of MDRAC is hampered by phenocopy, a well known issue in the investigation of complex diseases. It is a variation of a phenotype that resembles as a genetic factor, but it is due to an environmental cause rather than a genetic inheritance [305]. Any methodological changes in the method cannot have the impact on power of MDRAC in the presence of phenocopy. The power of MDRAC can be improved by identifying appropriate environmental factors responsible for the phenocopy [36]. These

environmental factors are included in the analysis. Identifying the appropriate environmental factors to include in the analysis is challenging in genetics. However, a previous study evaluated the performance of five methods (TEAM [21], BOOST [22], SNPRuler [191], AntEpiSeeker [92], and epiMODE [183]) by detecting power with/without noise. The experimental results showed that AntEpiseeker, BOOST, and SNPRuler are robust to phenocopy.

Genetic heterogeneity is a phenomenon in which several genes are associated with the same phenotype [306]. It can be either allelic or locus. Allelic heterogeneity of the disease is caused due to the various mutants within the same locus. Locus heterogeneity of the disease is caused due to the various mutants in different loci. The power of MDRAC in the presence of genetic heterogeneity is relatively low compared with other sources of noise. However, model 1 and model 2 relatively exhibited greater power than the other models due to the simplicity of the models [36]. That is, Model 1 and model 2 have four and three high-risk genotype combinations respectively. Hence, the experimental results demonstrated that the power of MDRAC in the presence of genetic heterogeneity is reduced, when the number of high-risk genotype combinations cells increased in the models. Additionally, the power of MDRAC in the presence of genetic heterogeneity is reduced as the rare allele frequency is decreased [36]. Model 5 and model 6 with a rare allele frequency 0.1 has the largest drop of the power. Some of the previous studies explored clustering analysis [307], recursive partitioning [308], and incorporating odds ratio (OR) function [113] to deal with genetic heterogeneity [36]. Many complex diseases such as Alzheimer disease, type 2 diabetes, coronary artery disease, breast cancer, and asthma have been demonstrated to exhibit genetic heterogeneity [309]. Hence, methods that will deal with genetic heterogeneity will play a vital role in understanding the genetic bases of complex disease.

## 4.6 Chapter Summary

In this chapter, a MDRAC method was implemented for detecting higher-order SNP interactions. The proposed method was evaluated on various balanced and imbalanced simulated datasets for two-locus to six-locus models. These simulated studies demonstrated improved accuracy for both balanced and imbalanced datasets by predicting the known interactions between SNPs at different loci. Further, MDRAC was

successfully evaluated over real datasets to identify disease causing interactions. The studies were further conducted to evaluate the power of MDRAC in the presence of GE, MS, PC, and GH. The results demonstrated improved power over the previous MDR method in all the epistasis models. However, the power of MDRAC was reduced in the presence of phenocopy and genetic heterogeneity or their combination with other sources of noise. Hence, in the next chapter, deep learning strategies are investigated to address this research problem.

# Chapter 5

## Towards Deep Learning Models for SNP Interaction studies

In the previous chapter, MDRAC was proposed for detecting higher-order SNP interactions in both balanced and imbalanced datasets. The study discovered important SNP interactions for better understanding of underlying biological mechanisms of sporadic breast cancer and hypertension data. Further, the performance of the proposed method was evaluated in the presence of noise due to GE, MS, PC, and GH. The experimental studies improved the prediction accuracy compared over the previous methods in the presence of MS, and GE. However, the power of MDRAC was reduced in the presence of GH and PC, and their combinations with other sources of noise. These new clues show that yet there is no breakthrough in producing replicable results. Hence, the research has been progressed by exploring the application of deep learning neural networks to address the current research problem in this chapter.

The chapter is organised as follows: Section 5.1 establishes the basics of deep neural networks with an example. The section further estimates the loss function. The proposed deep learning method is trained and presented in Section 5.2. The method is evaluated and discussed in Section 5.3 and Section 5.4. Once the results have been achieved, the sections further investigate the scoring metrics of the best models and compares with the previous methods.

---

This chapter is based on the following publications:

- S. Uppu, A. Krishna, and R. P. Gopalan, "Towards Deep Learning in genome-wide Association Interaction studies", The 20<sup>th</sup> Pacific Asia Conference on Information Systems (*PACIS*), page. 20, 2016.
- S. Uppu, A. Krishna, and R. P. Gopalan, "A Deep Learning Approach to Detect SNP Interactions," *Journal of Software (JSW)*, volume. 11, pp. 965-975, 2016.

## 5.1 Deep Neural Networks

In the current era of genetic epidemiology, conventional machine learning techniques are increasingly used to reveal underlying architecture behind complex diseases. However, none of the models have truly solved the problem of detecting or classifying the patterns in the genomic data. Deep learning is an emerging field that allows systems to learn the data by portraying in hierarchical abstractions. They allow the computational models to identify the representations required for the classification using general-purpose learning procedures [310]. These deep structured learning models provides stability, generalization, and scalability to big data by providing high prediction accuracy in a number of diverse problems [311]. Among these, deep learning has been a breakthrough in image recognition [312, 313], and speech recognition [314]. It has also produced promising results in language translation [315], reconstructing brain circuits [316], question answering [317], and natural language understanding [318]. Many researchers believe that these methods will have tremendous success in many other domains such as bioinformatics [310, 319]. This motivated the new exploration of training a deep learning method to detect two-locus interactions between SNPs.

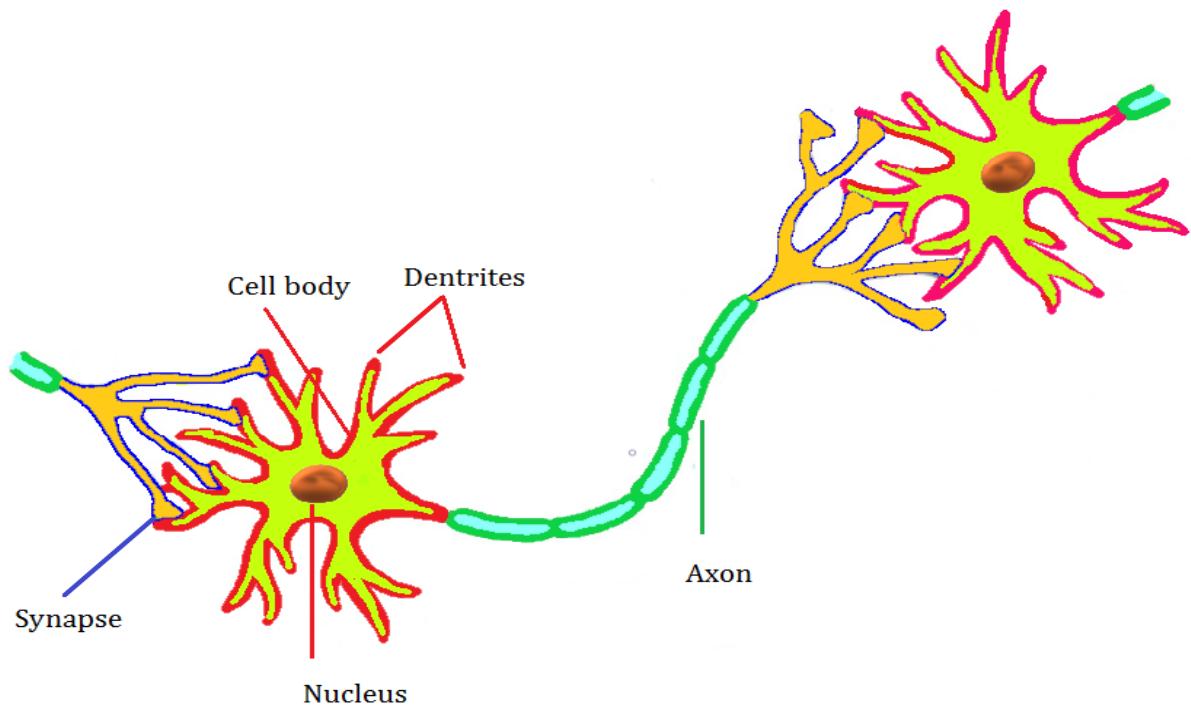


Figure 5.1 Biological structure of a neuron (adopted from [320]).

The basic unit of the deep neural networks (DNNs) are neurons. The neurons are biologically inspired from human brain. Figure 5.1 shows the communication between two neurons in a human brain.

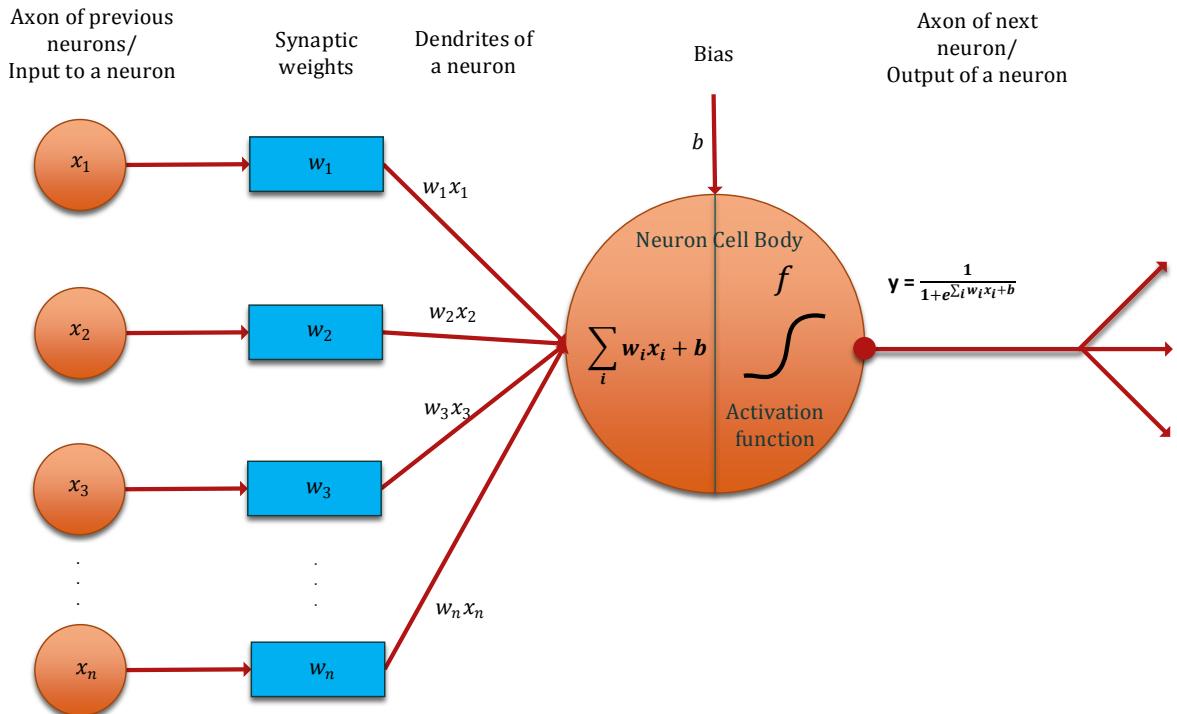


Figure 5.2 Structure of a neuron in a deep neural networks (adopted from [320]).

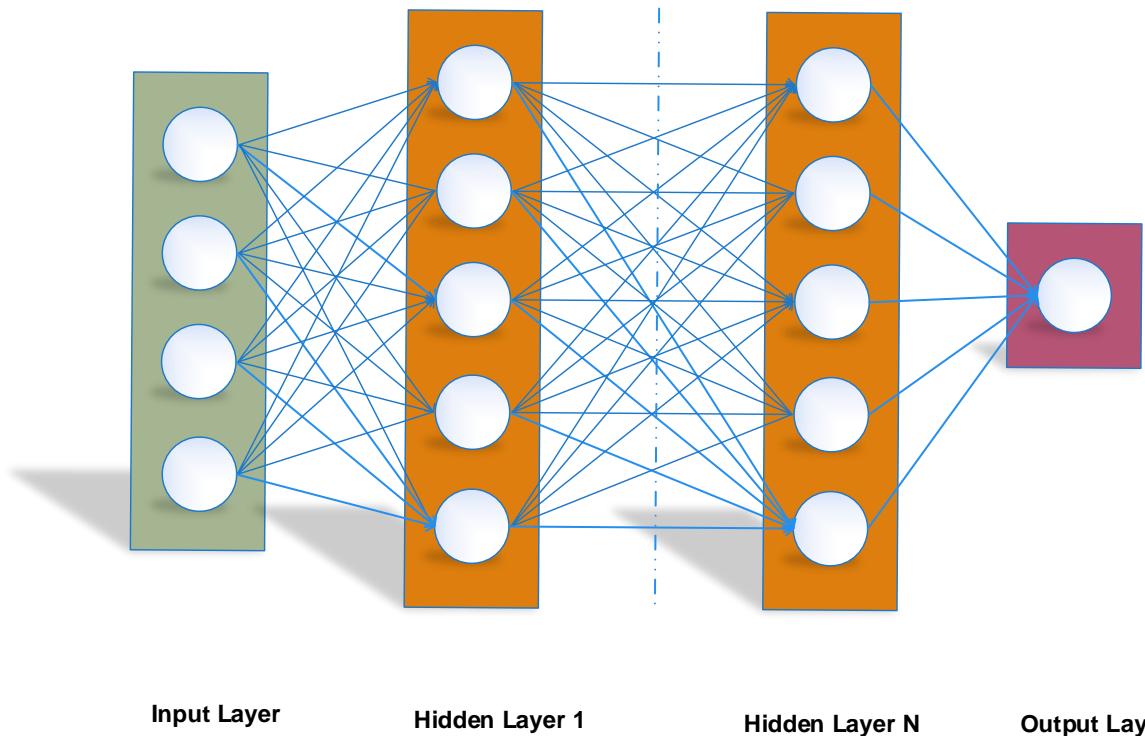


Figure 5.3 An example of a deep neural network: ©2018 IEEE.

The basic unit of a DNN is a neuron/computational unit. Figure 5.2 is an illustration of modelling biological neuron to a computational unit of a DNN. Neurons are interconnected in multiple layers. That is, NNs are stacked together in several layers for deep learning. Figure 5.3 is an illustrative example of DNN with an input layer  $s$ , hidden layers  $N$ , and an output layer  $y$  (adopted from [321]).

### **Example: Multilayered feed forward neural network**

Consider Figure 5.4 is an example of a four layered feed forward neural network adopted from [310] [321]. It comprises of single input and output layers along with three hidden layers. The output of the input layer forms the input to the hidden layers, and the output of third hidden layer is fed into the output layer as an input. The feedforward and backward propagation of the illustrated network is explained in detail in the following sections (Section 5.1.1- Section 5.1.4) based on [321-323].

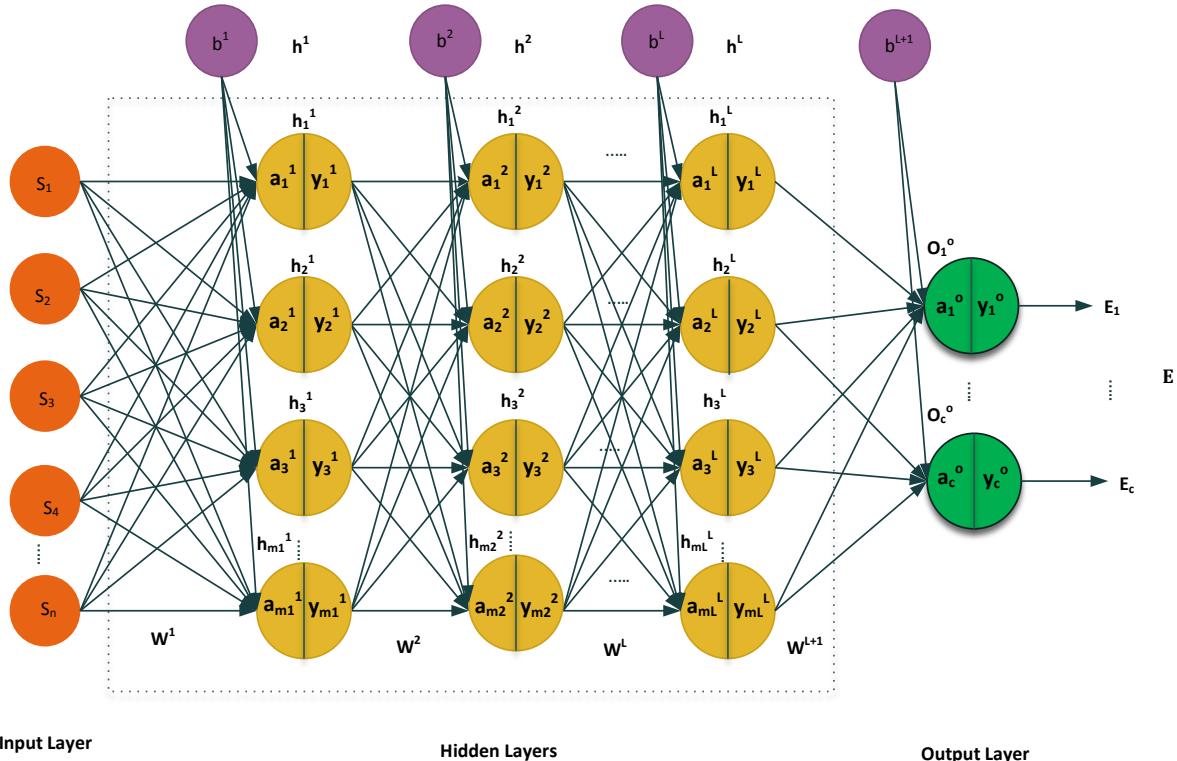


Figure 5.4 Basic structure of a four-layer feedforward network.

#### **5.1.1 Forward propagation**

The sum of the weighted inputs to a neuron is transformed using a non-linear activation function as a forward propagation. Figure 5.5 represents an example of feedforward propagation of Figure 5.4. The example is computed using logistic activation function.

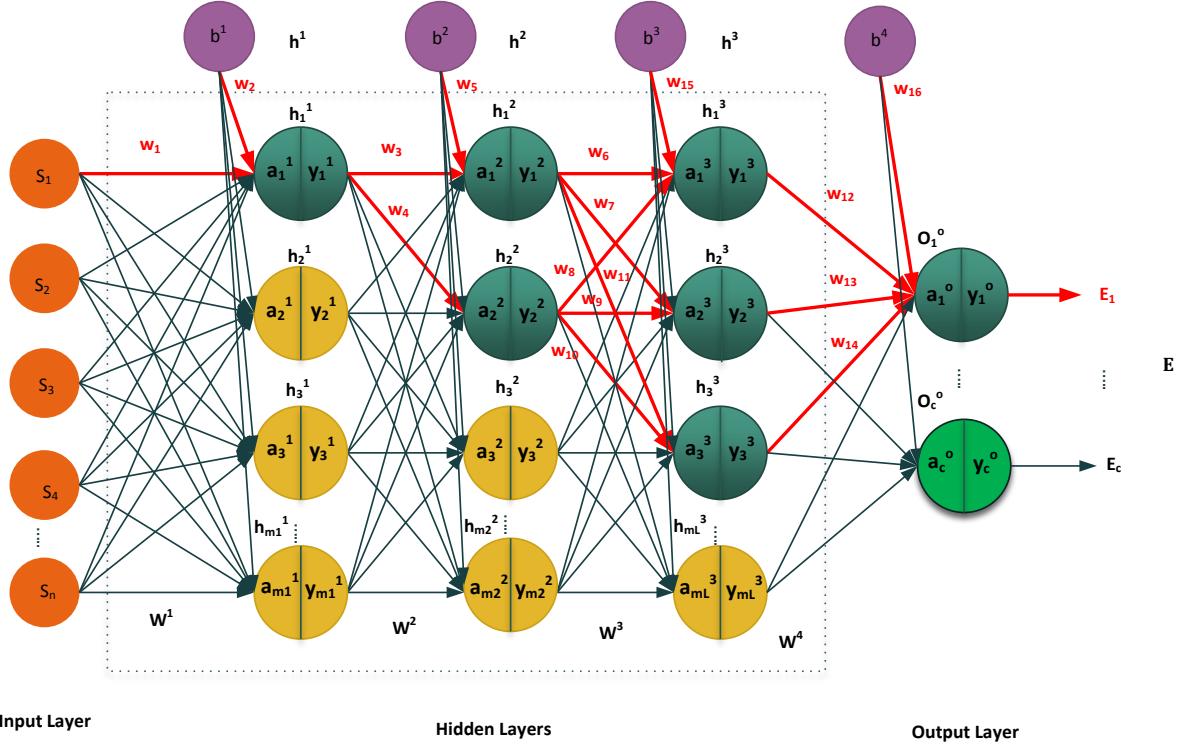


Figure.5.5 Example of a feedforward propagation of a four-layer feedforward network.

In general, the logistic or sigmoidal activation function or transformation function is given by:

$$y = f(a) = \frac{1}{1+e^{-a}} = \frac{1}{1+e^{-(\sum_n s_n w_n + b)}} \quad (5.6)$$

For Figure 5.5, weighted sum and logistic activation functions of each neuron in the corresponding layer is computed below:

$$a_1^1 = s_1 w_1 + b^1$$

$$y_1^1 = f(a_1^1) = \frac{1}{1 + e^{-(a_1^1)}}$$

Where  $a_1^1$  and  $y_1^1$  are the weighted sum and transformation function of neuron  $h_1^1$  in the first hidden layer.

$$a_1^2 = y_1^1 w_3 + b^2$$

$$y_1^2 = f(a_1^2) = \frac{1}{1 + e^{-(a_1^2)}}$$

Where  $a_1^2$  and  $y_1^2$  are the weighted sum and transformation function of neuron  $h_1^2$  in the

second hidden layer.

$$a_2^2 = y_1^1 w_4 + b^2$$

$$y_2^2 = f(a_2^2) = \frac{1}{1 + e^{-(a_2^2)}}$$

Similarly,  $a_2^2$  and  $y_2^2$  are the weighted sum and transformation function of neuron  $h_2^2$  in the second hidden layer.

$$a_1^3 = y_1^2 w_6 + y_2^2 w_8 + b^3$$

$$y_1^3 = f(a_1^3) = \frac{1}{1 + e^{-(a_1^3)}}$$

$$a_2^3 = y_1^2 w_7 + y_2^2 w_9 + b^3$$

$$y_2^3 = f(a_2^3) = \frac{1}{1 + e^{-(a_2^3)}}$$

$$a_3^3 = y_2^2 w_{10} + y_1^2 w_{11} + b^3$$

$$y_3^3 = f(a_3^3) = \frac{1}{1 + e^{-(a_3^3)}}$$

Where  $a_1^3, a_2^3, a_3^3$  and  $y_1^3, y_2^3, y_3^3$  are the weighted sums and activation functions of neurons  $h_1^3, h_2^3, h_3^3$  in the third hidden layer.

$$a_1^o = y_1^3 w_{12} + y_2^3 w_{13} + y_3^3 w_{14} + b^4$$

$$y_1^o = f(a_1^o) = \frac{1}{1 + e^{-(a_1^o)}}$$

Where  $a_1^o$  and  $y_1^o$  are the weighted sum and activation function of output neuron  $a_1^o$  in the output layer. The objective function of learning is to adapt the weights by minimising the loss. The mean square error or cost function for two classes is given by:

$$E = \frac{1}{2} \sum_c (y'_c - y_c)^2 \quad (5.7)$$

Where  $y'_c$  represents target output and  $y_c$  denotes observed or actual output. For the above example, the mean square error is estimated as:

$$E_1 = \frac{1}{2} (\text{target output} - \text{actual output})^2$$

$$E_1 = \frac{1}{2} (y_1^{o'} - y_1^o)^2$$

### 5.1.2 Backpropagation

The network is trained using gradient decent algorithm. It minimizes the error function in a downhill direction by taking small steps. Big steps may lead to take uphill again. Computing the partial derivatives of error function with respect to the weights using the chain rule of differentiation is referred as backpropagation of error or backprop [324]. The gradients of each neuron are computed backwards from the activated output neuron to the input of first hidden layer based on [322]. Figure 5.6 shows the backpropagation for the example network considered in Figure 5.4.

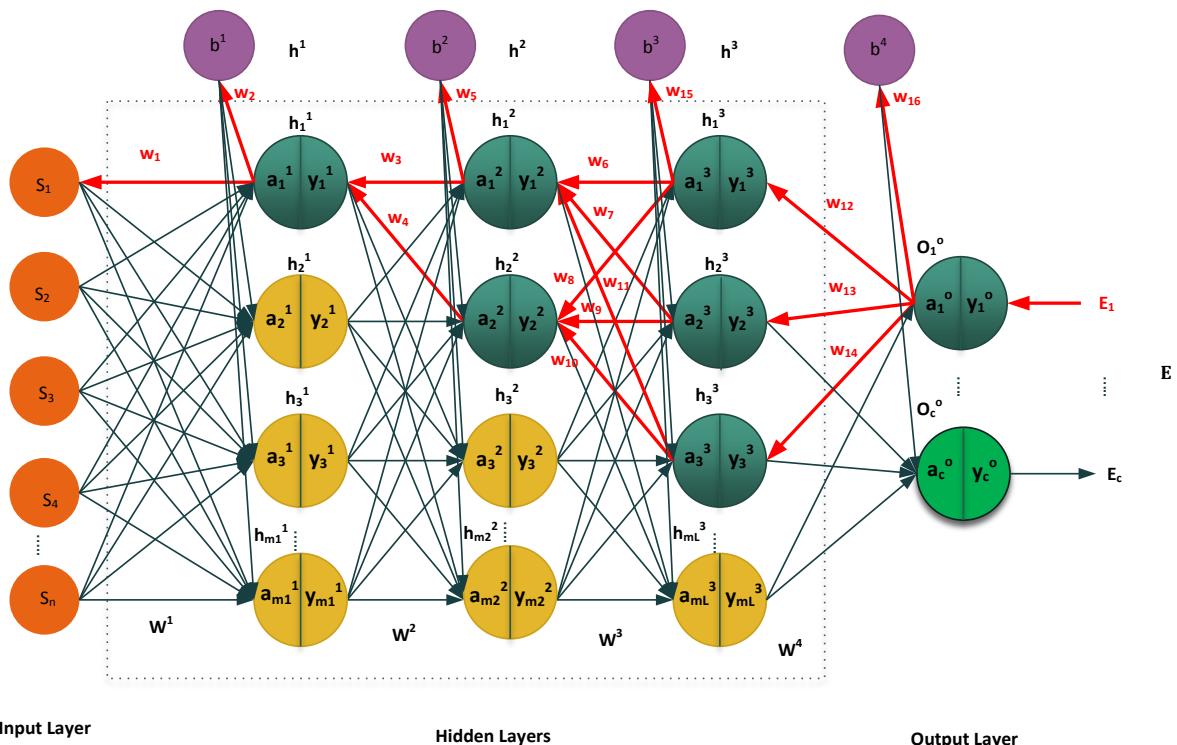


Figure.5.6 Example of a backpropagation of a four-layer feedforward network.

For the above example in the Figure 5.6, gradient of  $E_1$  with respect to weight  $w_1$ , ( $\frac{\partial E_1}{\partial w_1}$ ), is computed for each of the possible following paths from the output layer to the input

layer by applying chain rule of differentiation. Hence, the total gradient ( $\frac{\partial E_1}{\partial w_1}$ ) of error  $E_1$  with respect to  $w_1$  is the sum of the products of paths 1-6 (Figure 5.6.1 – 5.6.6) (computed based on [322]).

### Path 1

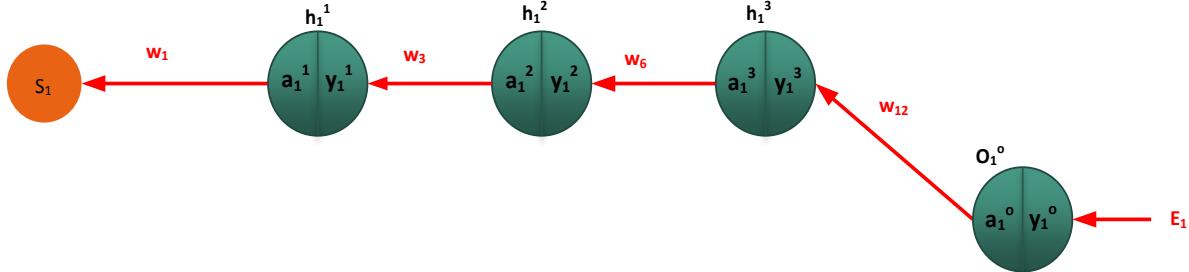


Figure.5.6.1 Path-1 between output and input layer.

$$\frac{\partial E_1}{\partial w_1} = \frac{\partial E_1}{\partial y_1^o} \frac{\partial y_1^o}{\partial a_1^o} \frac{\partial a_1^o}{\partial y_1^3} \frac{\partial y_1^3}{\partial a_1^3} \frac{\partial a_1^3}{\partial y_1^2} \frac{\partial y_1^2}{\partial a_2^2} \frac{\partial a_2^2}{\partial y_1^1} \frac{\partial y_1^1}{\partial a_1^1} \frac{\partial a_1^1}{\partial w_1} \quad (5.8)$$

### Path 2

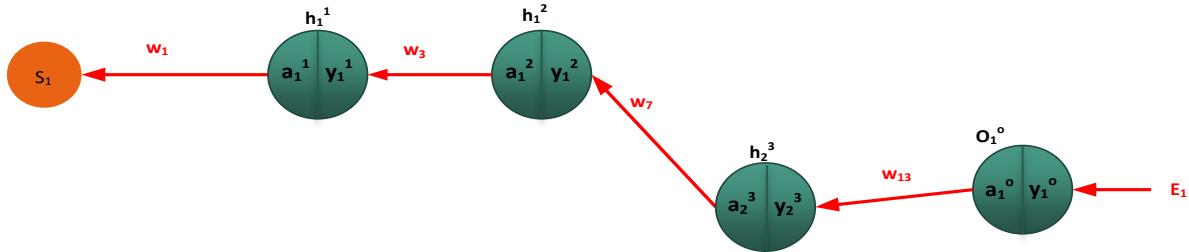


Figure.5.6.2 Path-2 between output layer and input layer.

$$\frac{\partial E_1}{\partial w_1} = \frac{\partial E_1}{\partial y_1^o} \frac{\partial y_1^o}{\partial a_1^o} \frac{\partial a_1^o}{\partial y_2^3} \frac{\partial y_2^3}{\partial a_2^3} \frac{\partial a_2^3}{\partial y_1^2} \frac{\partial y_1^2}{\partial a_2^2} \frac{\partial a_2^2}{\partial y_1^1} \frac{\partial y_1^1}{\partial a_1^1} \frac{\partial a_1^1}{\partial w_1} \quad (5.9)$$

### Path 3

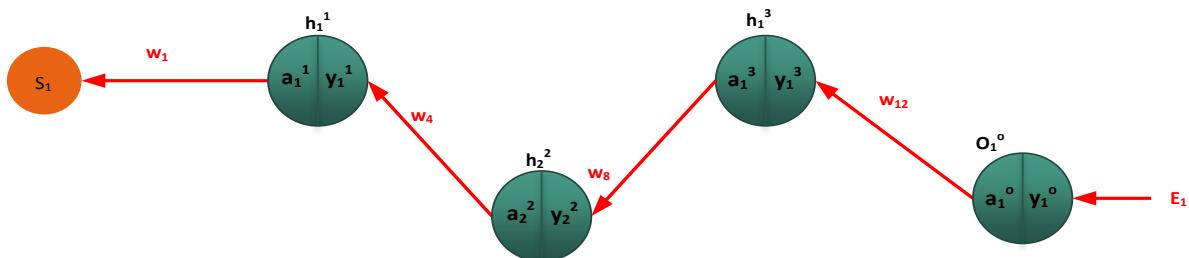


Figure.5.6.3 Path-3 between output layer and input layer.

$$\frac{\partial E_1}{\partial w_1} = \frac{\partial E_1}{\partial y_1^o} \frac{\partial y_1^o}{\partial a_1^o} \frac{\partial a_1^o}{\partial y_1^3} \frac{\partial y_1^3}{\partial a_1^3} \frac{\partial a_1^3}{\partial y_2^2} \frac{\partial y_2^2}{\partial a_2^2} \frac{\partial a_2^2}{\partial y_1^1} \frac{\partial y_1^1}{\partial a_1^1} \frac{\partial a_1^1}{\partial w_1} \quad (5.10)$$

#### Path 4

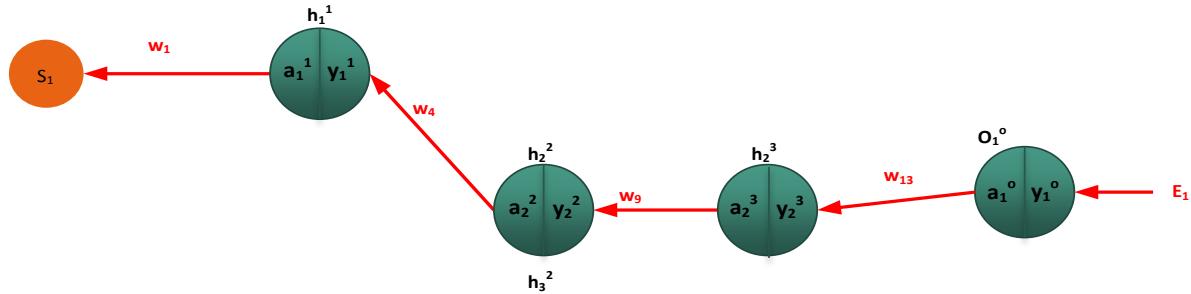


Figure.5.6.4 Path-4 between output layer and input layer.

$$\frac{\partial E_1}{\partial w_1} = \frac{\partial E_1}{\partial y_1^o} \frac{\partial y_1^o}{\partial a_1^o} \frac{\partial a_1^o}{\partial y_3^3} \frac{\partial y_3^3}{\partial a_3^3} \frac{\partial a_3^3}{\partial y_2^2} \frac{\partial y_2^2}{\partial a_2^2} \frac{\partial a_2^2}{\partial y_1^1} \frac{\partial y_1^1}{\partial a_1^1} \frac{\partial a_1^1}{\partial w_1} \quad (5.11)$$

#### Path 5

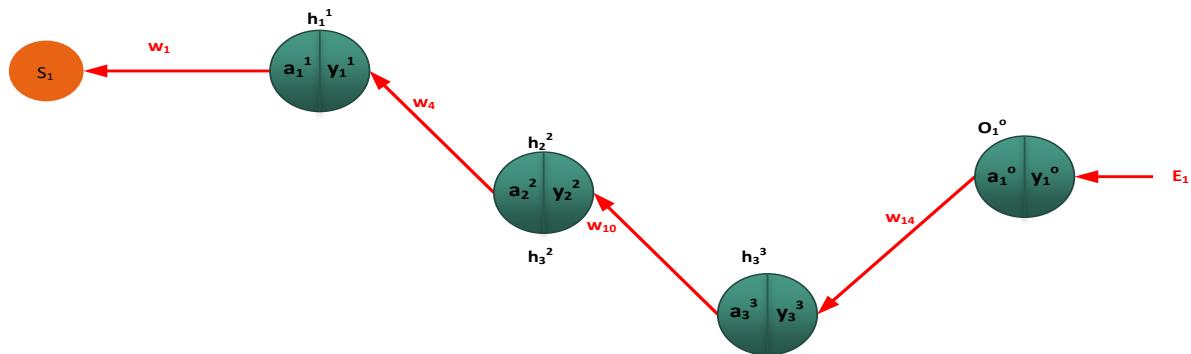


Figure.5.6.5 Path-5 between output layer and input layer.

$$\frac{\partial E_1}{\partial w_1} = \frac{\partial E_1}{\partial y_1^o} \frac{\partial y_1^o}{\partial a_1^o} \frac{\partial a_1^o}{\partial y_3^3} \frac{\partial y_3^3}{\partial a_3^3} \frac{\partial a_3^3}{\partial y_2^2} \frac{\partial y_2^2}{\partial a_2^2} \frac{\partial a_2^2}{\partial y_1^1} \frac{\partial y_1^1}{\partial a_1^1} \frac{\partial a_1^1}{\partial w_1} \quad (5.12)$$

#### Path 6

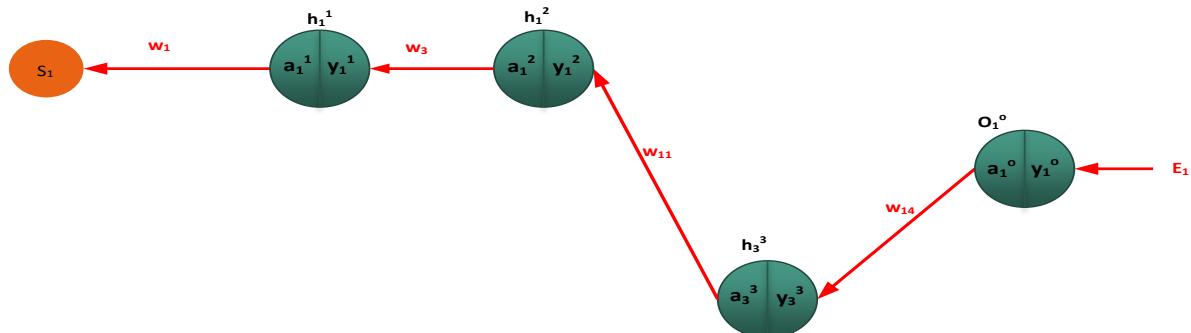


Figure.5.6.6 Path-6 between output layer and input layer.

$$\frac{\partial E_1}{\partial w_1} = \frac{\partial E_1}{\partial y_1^o} \frac{\partial y_1^o}{\partial a_1^o} \frac{\partial a_1^o}{\partial y_3^3} \frac{\partial y_3^3}{\partial a_3^3} \frac{\partial a_3^3}{\partial y_1^2} \frac{\partial y_1^2}{\partial a_1^2} \frac{\partial a_1^2}{\partial y_1^1} \frac{\partial y_1^1}{\partial a_1^1} \frac{\partial a_1^1}{\partial w_1} \quad (5.13)$$

Jacobian matrix of gradients of  $E_1$  with respect to weight  $w_1$  of all the possible paths 1- 6 with respect to weight  $w_1$  is computed as below:

$$\frac{\partial E_1}{\partial w_1} = \begin{bmatrix} \left[ \frac{\partial E_1}{\partial y_1^o} \frac{\partial y_1^o}{\partial a_1^o} \frac{\partial a_1^o}{\partial y_1^3} \right] \left[ \frac{\partial y_1^3}{\partial a_1^3} \frac{\partial a_1^3}{\partial y_1^3} \right] \left[ \frac{\partial y_1^2}{\partial a_2^2} \frac{\partial a_2^2}{\partial y_1^1} \right] + \\ \left[ \frac{\partial E_1}{\partial y_1^o} \frac{\partial y_1^o}{\partial a_1^o} \frac{\partial a_1^o}{\partial y_2^3} \right] \left[ \frac{\partial y_2^3}{\partial a_2^3} \frac{\partial a_2^3}{\partial y_1^2} \right] \left[ \frac{\partial y_1^2}{\partial a_1^2} \frac{\partial a_1^2}{\partial y_1^1} \right] + \\ \left[ \frac{\partial E_1}{\partial y_1^o} \frac{\partial y_1^o}{\partial a_1^o} \frac{\partial a_1^o}{\partial y_1^3} \right] \left[ \frac{\partial y_1^3}{\partial a_1^3} \frac{\partial a_1^3}{\partial y_2^2} \right] \left[ \frac{\partial y_2^2}{\partial a_2^2} \frac{\partial a_2^2}{\partial y_1^1} \right] + \\ \left[ \frac{\partial E_1}{\partial y_1^o} \frac{\partial y_1^o}{\partial a_1^o} \frac{\partial a_1^o}{\partial y_3^3} \right] \left[ \frac{\partial y_3^3}{\partial a_3^3} \frac{\partial a_3^3}{\partial y_2^2} \right] \left[ \frac{\partial y_2^2}{\partial a_2^2} \frac{\partial a_2^2}{\partial y_1^1} \right] + \\ \left[ \frac{\partial E_1}{\partial y_1^o} \frac{\partial y_1^o}{\partial a_1^o} \frac{\partial a_1^o}{\partial y_2^3} \right] \left[ \frac{\partial y_2^3}{\partial a_2^3} \frac{\partial a_2^3}{\partial y_2^2} \right] \left[ \frac{\partial y_2^2}{\partial a_2^2} \frac{\partial a_2^2}{\partial y_1^1} \right] + \\ \left[ \frac{\partial E_1}{\partial y_1^o} \frac{\partial y_1^o}{\partial a_1^o} \frac{\partial a_1^o}{\partial y_3^3} \right] \left[ \frac{\partial y_3^3}{\partial a_3^3} \frac{\partial a_3^3}{\partial y_1^2} \right] \left[ \frac{\partial y_1^2}{\partial a_1^2} \frac{\partial a_1^2}{\partial y_1^1} \right] + \\ \left[ \frac{\partial E_1}{\partial y_1^o} \frac{\partial y_1^o}{\partial a_1^o} \frac{\partial a_1^o}{\partial y_3^3} \right] \left[ \frac{\partial y_3^3}{\partial a_3^3} \frac{\partial a_3^3}{\partial y_2^2} \right] \left[ \frac{\partial y_2^2}{\partial a_2^2} \frac{\partial a_2^2}{\partial y_1^1} \right] \end{bmatrix} \left[ \frac{\partial y_1^1}{\partial a_1^1} \frac{\partial a_1^1}{\partial w_1} \right] \quad (5.14)$$

The equation (5.14) is rewritten as below by taking common factors.

$$\frac{\partial E_1}{\partial w_1} = \begin{bmatrix} \left[ \left[ \frac{\partial E_1}{\partial y_1^o} \frac{\partial y_1^o}{\partial a_1^o} \frac{\partial a_1^o}{\partial y_1^3} \right] \left[ \frac{\partial y_1^3}{\partial a_1^3} \frac{\partial a_1^3}{\partial y_1^3} \right] + \right] \left[ \frac{\partial y_1^2}{\partial a_2^2} \frac{\partial a_2^2}{\partial y_1^1} \right] \\ \left[ \left[ \frac{\partial E_1}{\partial y_1^o} \frac{\partial y_1^o}{\partial a_1^o} \frac{\partial a_1^o}{\partial y_1^3} \right] \left[ \frac{\partial y_1^3}{\partial a_1^3} \frac{\partial a_1^3}{\partial y_2^2} \right] \right] \left[ \frac{\partial y_2^2}{\partial a_2^2} \frac{\partial a_2^2}{\partial y_1^1} \right] \\ \left[ \left[ \frac{\partial E_1}{\partial y_1^o} \frac{\partial y_1^o}{\partial a_1^o} \frac{\partial a_1^o}{\partial y_2^3} \right] \left[ \frac{\partial y_2^3}{\partial a_2^3} \frac{\partial a_2^3}{\partial y_1^2} \right] + \right] \left[ \frac{\partial y_1^2}{\partial a_1^2} \frac{\partial a_1^2}{\partial y_1^1} \right] \left[ \frac{\partial y_1^1}{\partial a_1^1} \frac{\partial a_1^1}{\partial w_1} \right] \\ \left[ \left[ \frac{\partial E_1}{\partial y_1^o} \frac{\partial y_1^o}{\partial a_1^o} \frac{\partial a_1^o}{\partial y_3^3} \right] \left[ \frac{\partial y_3^3}{\partial a_3^3} \frac{\partial a_3^3}{\partial y_1^2} \right] + \right] \left[ \frac{\partial y_2^2}{\partial a_2^2} \frac{\partial a_2^2}{\partial y_1^1} \right] \\ \left[ \left[ \frac{\partial E_1}{\partial y_1^o} \frac{\partial y_1^o}{\partial a_1^o} \frac{\partial a_1^o}{\partial y_2^3} \right] \left[ \frac{\partial y_2^3}{\partial a_2^3} \frac{\partial a_2^3}{\partial y_2^2} \right] + \right] \left[ \frac{\partial y_2^2}{\partial a_2^2} \frac{\partial a_2^2}{\partial y_1^1} \right] \\ \left[ \left[ \frac{\partial E_1}{\partial y_1^o} \frac{\partial y_1^o}{\partial a_1^o} \frac{\partial a_1^o}{\partial y_3^3} \right] \left[ \frac{\partial y_3^3}{\partial a_3^3} \frac{\partial a_3^3}{\partial y_2^2} \right] \right] \left[ \frac{\partial y_2^2}{\partial a_2^2} \frac{\partial a_2^2}{\partial y_1^1} \right] \end{bmatrix} \quad (5.15)$$

The weight  $w_1$  is updated with new weight  $w'_1$  after one iteration is given below.

$$w'_1 = w_1 - \eta \frac{\partial E_1}{\partial w_1} \quad (5.16)$$

Similarly  $\frac{\partial E_1}{\partial w_3}$  are computed for paths 7-9 (Figure 5.6.7 – Figure 5.6.9) with respect to weight  $w_3$  from the output layer to the first hidden layer(computed based on [322]).

### Path 7

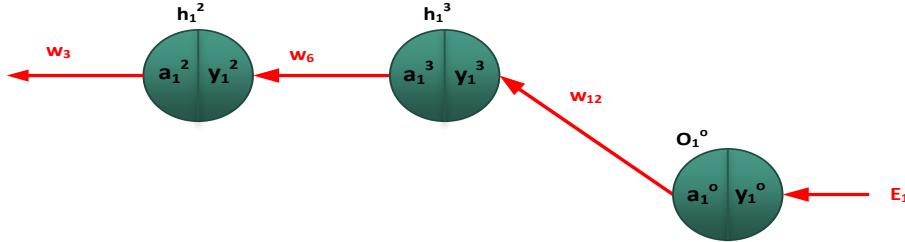


Figure 5.6.7 Path-7 between output layer and first hidden layer.

$$\frac{\partial E_1}{\partial w_3} = \frac{\partial E_1}{\partial y_1^o} \frac{\partial y_1^o}{\partial a_1^o} \frac{\partial a_1^o}{\partial y_1^3} \frac{\partial y_1^3}{\partial a_1^3} \frac{\partial a_1^3}{\partial y_1^2} \frac{\partial y_1^2}{\partial a_1^2} \frac{\partial a_1^2}{\partial w_3} \quad (5.17)$$

### Path 8

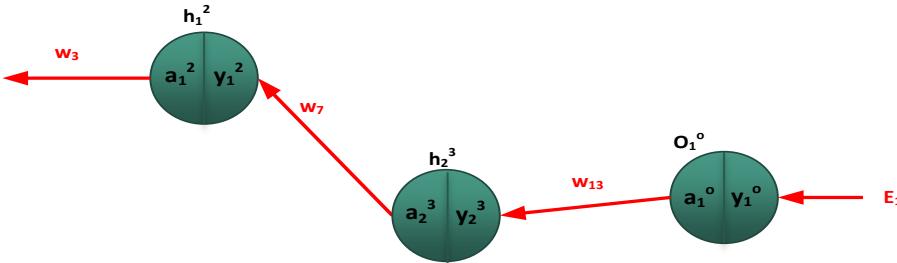


Figure 5.6.8 Path-8 between output layer and first hidden layer.

$$\frac{\partial E_1}{\partial w_3} = \frac{\partial E_1}{\partial y_1^o} \frac{\partial y_1^o}{\partial a_1^o} \frac{\partial a_1^o}{\partial y_2^3} \frac{\partial y_2^3}{\partial a_2^3} \frac{\partial a_2^3}{\partial y_1^2} \frac{\partial y_1^2}{\partial a_1^2} \frac{\partial a_1^2}{\partial w_3} \quad (5.18)$$

### Path 9

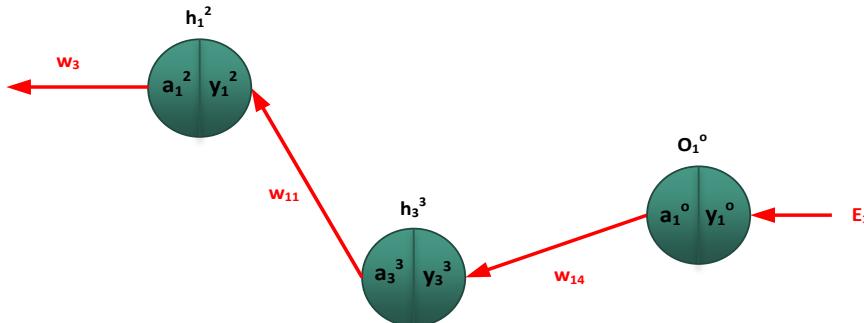


Figure 5.6.9 Path-9 between output layer and first hidden layer.

$$\frac{\partial E_1}{\partial w_3} = \frac{\partial E_1}{\partial y_1^o} \frac{\partial y_1^o}{\partial a_1^o} \frac{\partial a_1^o}{\partial y_3^3} \frac{\partial y_3^3}{\partial a_3^3} \frac{\partial a_3^3}{\partial y_1^2} \frac{\partial y_1^2}{\partial a_1^2} \frac{\partial a_1^2}{\partial w_3} \quad (5.19)$$

Jacobian matrix of  $E_1$  with respect to weight  $w_3$  for paths 7- 9 is:

$$\frac{\partial E_1}{\partial w_3} = \left[ \begin{bmatrix} \frac{\partial E_1}{\partial y_1^o} \frac{\partial y_1^o}{\partial a_1^o} \frac{\partial a_1^o}{\partial y_1^3} \\ \frac{\partial E_1}{\partial y_1^o} \frac{\partial y_1^o}{\partial a_1^o} \frac{\partial a_1^o}{\partial y_2^3} \\ \frac{\partial E_1}{\partial y_1^o} \frac{\partial y_1^o}{\partial a_1^o} \frac{\partial a_1^o}{\partial y_3^3} \end{bmatrix} \begin{bmatrix} \frac{\partial y_1^3}{\partial a_1^3} \frac{\partial a_1^3}{\partial y_1^2} \\ \frac{\partial y_2^3}{\partial a_2^3} \frac{\partial a_2^3}{\partial y_1^2} \\ \frac{\partial y_3^3}{\partial a_3^3} \frac{\partial a_3^3}{\partial y_1^2} \end{bmatrix} + \right] \begin{bmatrix} \frac{\partial y_1^2}{\partial a_1^2} \frac{\partial a_1^2}{\partial w_3} \\ \frac{\partial y_2^2}{\partial a_1^2} \frac{\partial a_1^2}{\partial w_3} \\ \frac{\partial y_3^2}{\partial a_1^2} \frac{\partial a_1^2}{\partial w_3} \end{bmatrix} \quad (5.20)$$

Gradient  $\frac{\partial E_1}{\partial w_6}$  is computed for paths 10 (Figure 5.6.10) with respect to weight  $w_6$  from the output layer to the second hidden layer and their corresponding Jacobian matrix is computed based on [322].

### Path 10

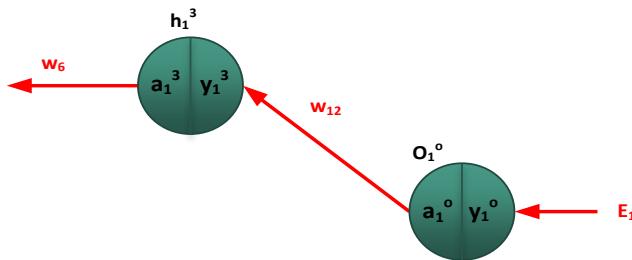


Figure 5.6.10 Path-10 between output layer and second hidden layer.

$$\frac{\partial E_1}{\partial w_6} = \frac{\partial E_1}{\partial y_1^o} \frac{\partial y_1^o}{\partial a_1^o} \frac{\partial a_1^o}{\partial y_1^3} \frac{\partial y_1^3}{\partial a_1^3} \frac{\partial a_1^3}{\partial w_6} \quad (5.21)$$

$$\frac{\partial E_1}{\partial w_6} = \left[ \begin{bmatrix} \frac{\partial E_1}{\partial y_1^o} \frac{\partial y_1^o}{\partial a_1^o} \frac{\partial a_1^o}{\partial y_1^3} \\ \frac{\partial E_1}{\partial y_1^o} \frac{\partial y_1^o}{\partial a_1^o} \frac{\partial a_1^o}{\partial y_2^3} \\ \frac{\partial E_1}{\partial y_1^o} \frac{\partial y_1^o}{\partial a_1^o} \frac{\partial a_1^o}{\partial y_3^3} \end{bmatrix} \begin{bmatrix} \frac{\partial y_1^3}{\partial a_1^3} \frac{\partial a_1^3}{\partial w_6} \\ \frac{\partial y_2^3}{\partial a_1^3} \frac{\partial a_1^3}{\partial w_6} \\ \frac{\partial y_3^3}{\partial a_1^3} \frac{\partial a_1^3}{\partial w_6} \end{bmatrix} \right] \quad (5.22)$$

Finally, gradient of  $E_1$  with respect to  $w_{12}$  is computed for paths 11 (Figure 5.6.11) as below based on [322]:

### Path 11

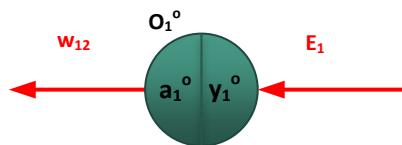


Figure 5.6.11 Path-11 between output layer and second hidden layer.

$$\frac{\partial E_1}{\partial w_{12}} = \left[ \begin{bmatrix} \frac{\partial E_1}{\partial y_1^o} \frac{\partial y_1^o}{\partial a_1^o} \frac{\partial a_1^o}{\partial w_{12}} \end{bmatrix} \right] \quad (5.23)$$

After one iteration, the weights are updated with new weights using backpropagation.

$$\begin{bmatrix} w'_1 \\ w'_3 \\ w'_6 \\ w'_{12} \end{bmatrix} = \begin{bmatrix} w_1 \\ w_3 \\ w_6 \\ w_{12} \end{bmatrix} - \eta \begin{bmatrix} \frac{\partial E_1}{\partial w_1} \\ \frac{\partial E_1}{\partial w_3} \\ \frac{\partial E_1}{\partial w_6} \\ \frac{\partial E_1}{\partial w_{12}} \end{bmatrix} \quad (5.24)$$

### 5.1.3 Mean square error estimate for logistic function

For the Figure 5.4, gradient of mean square loss function at the output layer is computed as:

$$\begin{aligned} \frac{\partial E_1}{\partial y_1^o} &= \frac{\partial \frac{1}{2} (y_1^{o'} - y_1^o)^2}{\partial y_1^o} \\ &= y_1^o - y_1^{o'} \end{aligned}$$

Where  $\frac{\partial y_1^o}{\partial a_1^o} = 1$  for linear function  $y_1^o = f(a_1^o)$ . The output gradient with respect to its weighted sum of the neuron  $a_1^o$  is:

$$\begin{aligned} \frac{\partial y_1^o}{\partial a_1^o} &= \frac{\partial \left( \frac{1}{1 + e^{-a_1^o}} \right)}{\partial a_1^o} \\ &= \left( \frac{1}{1 + e^{-a_1^o}} \right) \left( 1 - \left( \frac{1}{1 + e^{-a_1^o}} \right) \right) \\ &= y_1^o (1 - y_1^o) \end{aligned}$$

Similarly, gradients  $\frac{\partial a_1^o}{\partial y_1^3}, \frac{\partial y_1^3}{\partial a_1^3}, \frac{\partial y_1^2}{\partial a_1^2}$ , and  $\frac{\partial y_1^1}{\partial a_1^1}$  are computed as follows:

$$\begin{aligned} \frac{\partial a_1^o}{\partial y_1^3} &= \frac{\partial (y_1^3 w_{12} + y_2^3 w_{13} + y_3^3 w_{14})}{\partial y_1^3} \\ &= w_{12} \end{aligned}$$

$$\frac{\partial y_1^3}{\partial a_1^3} = \frac{\partial \left( \frac{1}{1 + e^{-a_1^3}} \right)}{\partial a_1^3} = \left( \frac{1}{1 + e^{-a_1^3}} \right) \left( 1 - \left( \frac{1}{1 + e^{-a_1^3}} \right) \right) = y_1^3 (1 - y_1^3)$$

$$\frac{\partial a_1^3}{\partial y_1^2} = \frac{\partial (y_1^2 w_6 + y_2^2 w_8)}{\partial y_1^2} = w_6$$

$$\frac{\partial y_1^2}{\partial a_1^2} = \frac{\partial \left( \frac{1}{1 + e^{-a_1^2}} \right)}{\partial a_1^2} = \left( \frac{1}{1 + e^{-a_1^2}} \right) \left( 1 - \left( \frac{1}{1 + e^{-a_1^2}} \right) \right) = y_1^2 (1 - y_1^2)$$

$$\frac{\partial a_1^2}{\partial y_1^1} = \frac{\partial (y_1^1 w_3)}{\partial y_1^1} = w_3$$

$$\frac{\partial y_1^1}{\partial a_1^1} = \frac{\partial \left( \frac{1}{1 + e^{-a_1^1}} \right)}{\partial a_1^1} = \left( \frac{1}{1 + e^{-a_1^1}} \right) \left( 1 - \left( \frac{1}{1 + e^{-a_1^1}} \right) \right) y_1^1 (1 - y_1^1)$$

$$\frac{\partial a_1^1}{\partial w_1} = \frac{\partial (s_1 w_1)}{\partial w_1} = s_1$$

Substituting the above equations in equation 5.13, the overall error gradient with respect to the weight  $w_1$  is obtained:

$$\frac{\partial E_1}{\partial w_1} = (y_1^o - y_1^{o'}) y_1^o (1 - y_1^o) w_{12} y_1^3 (1 - y_1^3) w_6 y_1^2 (1 - y_1^2) w_3 y_1^1 (1 - y_1^1) s_1 \quad (5.25)$$

## 5.1.4 Cross entropy error estimate for logistic function

For the Figure 5.4, cross entropy error at the output layer is estimated as:

$$E_1 = - (y_1^{o'} \log y_1^o + (1 - y_1^{o'}) \log(1 - y_1^o))$$

Where  $y_1^o$  is the observed output, and  $y_1^{o'}$  is the target output. Logistic activation function of an output unit in the output layer is given by:

$$y_1^o = \frac{1}{1 + e^{-a_1^o}}$$

The cross entropy error gradient with respect to the observed output is obtained:

$$\frac{\partial E_1}{\partial y_1^o} = \frac{\partial (- (y_1^{o'} \log y_1^o + (1 - y_1^{o'}) \log(1 - y_1^o)))}{\partial y_1^o}$$

$$= \frac{(1 - y_1^{o'})}{(1 - y_1^o)} - \frac{y_1^{o'}}{y_1^o}$$

$$= \frac{y_1^o - y_1^{o'}}{y_1^o (1 - y_1^o)}$$

The gradients between output layer to the input layer are computed using chain rule of differentiation as in previous section:

$$\frac{\partial y_1^o}{\partial a_1^o} = y_1^o (1 - y_1^o)$$

$$\frac{\partial a_1^o}{\partial y_1^3} = w_{12}$$

$$\frac{\partial y_1^3}{\partial a_1^3} = y_1^3 (1 - y_1^3)$$

$$\frac{\partial a_1^3}{\partial y_1^2} = w_6$$

$$\frac{\partial y_1^2}{\partial a_1^2} = y_1^2 (1 - y_1^2)$$

$$\frac{\partial a_1^2}{\partial y_1^1} = w_3$$

$$\frac{\partial y_1^1}{\partial a_1^1} = y_1^1 (1 - y_1^1)$$

$$\frac{\partial a_1^1}{\partial w_1} = s_1$$

Cross entropy error gradient  $E_1$  with respect to weight  $w_1$  is estimated by substituting the above gradient values in the equation 5.13:

$$\begin{aligned} \frac{\partial E_1}{\partial w_1} &= \frac{y_1^o - y_1^{o'}}{y_1^o (1 - y_1^o)} y_1^o (1 - y_1^o) w_{12} y_1^3 (1 - y_1^3) w_6 y_1^2 (1 \\ &\quad - y_1^2) w_3 y_1^1 (1 - y_1^1) s_1 \end{aligned}$$

$$= (y_1^o - y_1^{o'}) w_{12} y_1^3 (1 - y_1^3) w_6 y_1^2 (1 - y_1^2) w_3 y_1^1 (1 - y_1^1) s_1 \quad (5.26)$$

## 5.2 Deep Learning Method

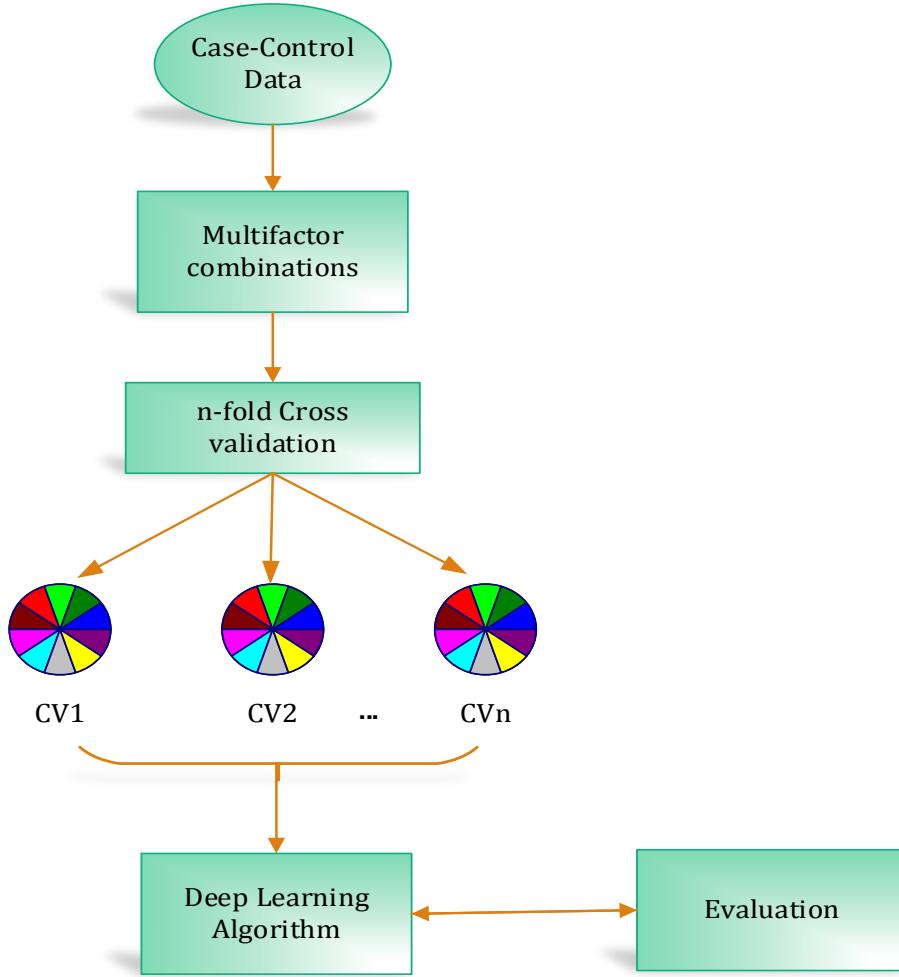


Figure 5.7 Overview of the deep learning method.

The workflow of the proposed deep learning method is illustrated in Figure 5.7 (based on [321]). Stage one comprises of case-control based data input. Multi-factor combinations of SNPs at various locations are combined together in stage two. It is performed to improve the prediction accuracy of the models such that none of the SNPs are left. Ten-fold cross validation is conducted in stage three to analyse the predictive power of the method. Stages four and five consist of the deep learning algorithm and its evaluation respectively. The deep learning algorithm classifies high-risk genotype combinations, and discovers multi-locus SNP interactions associated to a disease manifestation. Finally, the model is evaluated by varying various parameters. The model with the lowest classification error is identified as the best model.

### 5.2.1 Case-control Data

Case-control datasets comprise of  $s$  samples with  $m$  factors, and a class label  $c$ , which either takes 0 (control) or 1(case). Each factor is a SNP at a locus. A SNP is the variation in a single nucleotide of a DNA sequence. Due to duplication of genes, SNPs are biallele (A and a), whose genotypes are homozygous dominant (AA), heterozygous (aA\Aa), and homozygous recessive (aa). Statistically, AA, aA\Aa, and aa are represented by the values 1, 2, and 3 respectively. The datasets, such as, whole genome data, and sporadic breast cancer data explained in Section 4.2.3, and Section 3.3 of previous chapters are used in the evaluation of the proposed deep learning method.

### 5.2.2 Multifactor combinations and Cross-validation

In two-locus models, consider SNP A with genotypes AA, Aa\aA, and aa, and SNP B at locus B with genotypes BB, Bb\bB, and bb. The contingency table is generated with 9 multi-factor cells. That is, in case of two-locus SNPs from the pool of  $m$  factors, each factor with three genotypes creates contingency table with 9 multifactor cells, represented in 9 dimensional spaces. Hence, there are  $3^k$  genotype combinations for  $k$  loci in  $k$  dimensional space. Sum of all the combinations for  $k$  loci from the pool of  $m$  factors is given below using binomial coefficient.

$$\sum_{0 < k \leq m} \binom{m}{k} = \sum_{k=1}^m \frac{m!}{k!(m-k)!} \quad (5.27)$$

Where  $\frac{m!}{k!(m-k)!}$  is equal to 1, whenever  $= m$ .

### 5.2.3 Data partition and data analysis

Data partition and data analysis in this chapter is performed as in the previous chapters. The datasets are split into equal parts of  $m$  for training, and independent testing without losing the data. That is, in  $m$ -fold cross validation,  $m - 1$  splits are used for training and remaining one split is used for testing. The method runs  $m$  times on training data by excluding different split each time for testing. In the proposed method, 10-fold cross validation is performed. It is the most successful internal validation method used in high-dimensional genome data. The models are also evaluated by splitting the data into three parts (80% for training, 10% for validation, and 10% for

testing). The performance of the models during training, validation, and testing are evaluated by determining the model's metrics. Training speed and time to execute the models are evaluated by varying width and depth of the network, along with various activation functions. The overall best model with highest prediction accuracy and lowest logloss along with the highest cross validation consistency (CVC) is selected. The final results are statistically evaluated with a 1,000 fold permutation test, whose p values are compared with 0.05 in determining the statistical significance of the findings.

#### 5.2.4 Deep learning algorithm

The method is based on multilayered feedforward neural networks [310, 311, 321]. It is also known as multilayer perceptrons (MLPs) that comprise of multiple layers interconnected with neurons. It comprises of single input and output layer along with multiple hidden layers. The output of the input layer forms the input to the hidden layers, and the output of last hidden layer is fed into the output layer as an input as discussed in the previous section. Each neuron computes weighted sum of its input. The generalised sum of all weighted inputs to a neuron is computed as below:

$$a = \sum_n s_n w_n + b \quad (5.28)$$

Where weighted sum  $\sum_n s_n w_n$  is less than or greater than a threshold level,  $b$  is bias ( $b \equiv -\text{threshold}$ ), and  $w$  is a weight assigned to the input  $s$  for each neuron. The number of neurons in each layer is represented as width of the model and the number of layers in the network is represented as depth of the model. Not only, weights and biases linking the neurons determine the output of the entire network, but it also depends on the width and depth of the network. The weighted sum of a neuron is transformed by a non-linear activation function  $f(\cdot)$  to compute the output of a neuron. The hyperbolic tangent activation function is used in this method to transform the output of a neuron. The hyperbolic tangent ( $\tanh$ ) function is a rescaled, and shifted logistic function, whose symmetry is around 0, such that it allows the algorithm to converge faster. The  $\tanh$  activation function or transformation function is given by:

$$\begin{aligned} y = f(a) = \tanh(a) &= \frac{\sinha}{\cosh a} = \frac{e^a - e^{-a}}{e^a + e^{-a}} = \frac{e^{2a} - 1}{e^{2a} + 1} \\ &= \frac{e^{2(\sum_n s_n w_n + b)} - 1}{e^{2(\sum_n s_n w_n + b)} + 1} \end{aligned} \quad (5.29)$$

Where  $\tanh(a) \in (-1, 1)$ . The gradient of  $\tanh(a)$  function is:

$$f(a)' = \tanh(a)' = \left(1 - \left(\frac{e^a - e^{-a}}{e^a + e^{-a}}\right)^2\right) = 1 - \tanh^2(a) \quad (5.30)$$

The network is trained by three hidden layers with 50 neurons in each hidden layer. Backpropagation is used to adapt the weights by reducing the loss. The output error is estimated by using a cross entropy objective function. The cross entropy error or cost function is given by:

$$E == - \sum_c (y'_c \log(y_c) + (1 - y'_c) \log(1 - y_c)) \quad (5.31)$$

Where  $y'_c$  represents target output and  $y_c$  denotes observed or actual output. The network is trained by propagating the error backwards using gradient decent algorithm. The new adapted weight after an iteration using backpropagation is given by:

$$w' = w - \eta \frac{\partial E}{\partial w} \quad (5.32)$$

Where  $\eta$  is the learning rate, which specifies how much the parameters have to be adjusted in the direction of the gradient. Multinomial distribution function is used along with cross entropy (log-loss) for the response variables in classification. In practise, most of the researchers use parallel versions of stochastic gradient descent (SGD) to minimise the log-loss by handling the memory efficiently. It is used to compute the partial derivative of each parameter with respect to cross entropy loss function. It minimises the loss function by optimising the best fitting parameters using mini-batch strategy. However, the execution time of the algorithm drops drastically. In this method, SGD is parallelised by using a lock free approach to handle the memory efficiently [311, 325]. The method is scalable and can specify the number of training samples [326]. It is trained with  $N$  epochs (number of passes over training data) per iteration on  $M$  nodes. For example, consider the training samples per iteration to be 200,000 running on 8 nodes. Each node processes 25,000 samples per iteration. Hence, if the data has 20 million samples, there will be 80 distributed iterations to process one epoch.

## 5.2.5 Classification for more than two classes

In the trained method, softmax function is used to classify more than two classes [321]. Softmax is a generalized activation function applied in the output layer for multiple-

class classification problem. It takes multi-dimensional input data, and transforms into the range of  $(0, 1)$ . The softmax activation function at  $c^{th}$  output neuron is given by:

$$y_c^o = \frac{e^{a_c^o}}{\sum_k e^{a_k^o}} \quad (5.33)$$

Where  $k$  is the number of classes or number of output nodes in the output layer. Softmax is a two-step process in which,  $a_c^o$  is estimated for each class at step one, and softmax is applied to each class in step two.

Cross- entropy error estimation for multi-class output is given by:

$$E = - \sum_c y_c^o' \log(y_c^o) \quad (5.34)$$

Where  $c$  is the number of classes. The cross entropy error gradient is computed as:

$$\frac{\partial E}{\partial y_c^o} = - \frac{y_c^o'}{y_c^o} \quad (5.35)$$

### 5.2.6 Other activation functions

The hyperbolic tangent activation function is implemented in the proposed method. However, other non-linear activation functions are also explored in this study [321]. The sigmoidal activation function is a default choice in DNNs, and it is represented by:

$$f(a) = \frac{1}{1 + e^{-a}} \quad (5.36)$$

Where  $f(a) \in (0,1)$ , the gradient of sigmoidal activation function is:

$$f(a)' = \frac{-e^{-a}}{1 + e^{-a}} = f(a)(1 - f(a)) \quad (5.37)$$

Rectified linear unit activation (ReLU) function is a most popular choice in deep learning models as it does not saturate even for a larger values of  $a$ . It is given by:

$$f(a) = \max(0, a) \quad (5.38)$$

Where  $f(a) \in (0, \infty)$ . The gradient of ReLU function is:

$$f(a)' = \begin{cases} 1 & : a \geq 0 \\ 0 & : a < 0 \end{cases} \quad (5.39)$$

Maxout activation function is a generalised version of ReLU and its leaky form.

$$f(\vec{a}) = \max_i a_i \quad (5.40)$$

Where  $f(\vec{a}) \in (-\infty, \infty)$ .

$$f(\vec{a})' = \frac{\partial f}{\partial a_j} = \begin{cases} 1 & : j = \arg \max_i a_i \\ 0 & : j \neq \arg \max_i a_i \end{cases} \quad (5.41)$$

### 5.2.7 Dropout

Dropout is a regularisation technique used in the trained DNNs to prevent overfitting [327]. It randomly drops out neurons along with their connections to prevent neurons from co-adapting too much in the networks. That is, each neuron in the network prevents its activation with a probability  $p$  of 0.2 and 0.5 for the neurons in the input and hidden layers respectively. Hence, only the weights that are connected to the surviving neurons are updated with new weights during backpropagation. Forward propagation of dropout is given by:

$$a = \sum_n w_n s'_n + b \quad (5.42)$$

Where  $s'_n = s_n d_n$   $d_n$  is drawn randomly from independent Bernoulli distribution that has the probability ( $p$ ). Input neuron  $s_n$  is dropped when  $d_n = 0$ . As a result, sigmoid, tanh, ReLU, and maxout activation functions along with dropout are also explored to improve generalization of the method by preventing the data from overfitting.

### 5.2.8 Estimation of variable importance

Gedeon method is used for calculating variable importance [311, 328]. Initial studies in 1995, cancelled the weights with opposite signs. In 1997, he modified the contribution measures by removing the cancellation problem. The contribution of an input neuron  $s$  to a neuron in the hidden layer  $h$  is given by:

$$P_{sh} = \frac{|w_{sh}|}{\sum_{i=1}^{n_i} |w_{ih}|} \quad (5.43)$$

Where  $w_{sh}$  is the weight of the connection between input neuron  $s$  to the hidden neuron  $h$ ,  $n_i$  is the number of neurons in the input layer. The contribution of a neuron in the hidden layer  $h$  to a neuron in the output layer  $o$  is given by:

$$P_{ho} = \frac{|w_{ho}|}{\sum_{j=1}^{n_h} |w_{jh}|} \quad (5.44)$$

Where  $w_{ho}$  is the weight of the connection between hidden neuron  $h$  to the output neuron  $o$ , and  $n_h$  is the number of neurons in the hidden layer. The contribution of an input neuron  $s$  to a neuron in the output layer  $o$  is computed as:

$$P_{so} = \sum_{h=1}^{n_h} P_{sh} P_{ho} \quad (5.45)$$

Applying equations 5.42 and 5.43 in the equation 5.44,  $P_{so}$  is given by:

$$P_{so} = \sum_{h=1}^{n_h} \left( \frac{|w_{sh}|}{\sum_{i=1}^{n_i} |w_{ih}|} \right) \left( \frac{|w_{ho}|}{\sum_{j=1}^{n_h} |w_{jh}|} \right) \quad (5.46)$$

## 5.3 Evaluation over whole genome data

Several experiments are performed over the published data obtained from whole genome study [297] to evaluate the accuracy of the trained deep feedforward neural network. The goal of this study is to determine whether the model is a better approach for identifying SNP interactions in genome-wide interaction studies. It identifies statistically significant genotype combinatorial associations based on cases and controls. The approach is developed and analysed in R using H2O package [326].

Figure 5.8 shows the receiver operating characteristic curve (ROC) during training, validation and testing. It illustrates the true positive rate (sensitivity) vs false positive rate (specificity) of the method. Figure 5.9 shows the scoring history of the best model. Timestep (a unit of measurement for the x-axis) and metrics (a unit of measurement for the y-axis) are the arguments available in the scoring history of the model. Timestep for the model must be either epochs or samples. Metrics of the model are log-loss, r2 (a measure of goodness-of-fit for linear regression), area under the curve (AUC), mean square error value (MSE), and classification error. It is observed that as the number of training samples increased, synchronization and model convergence decreased. When the number of training samples is too low, network communication dominated the runtime by affecting the computational performance.

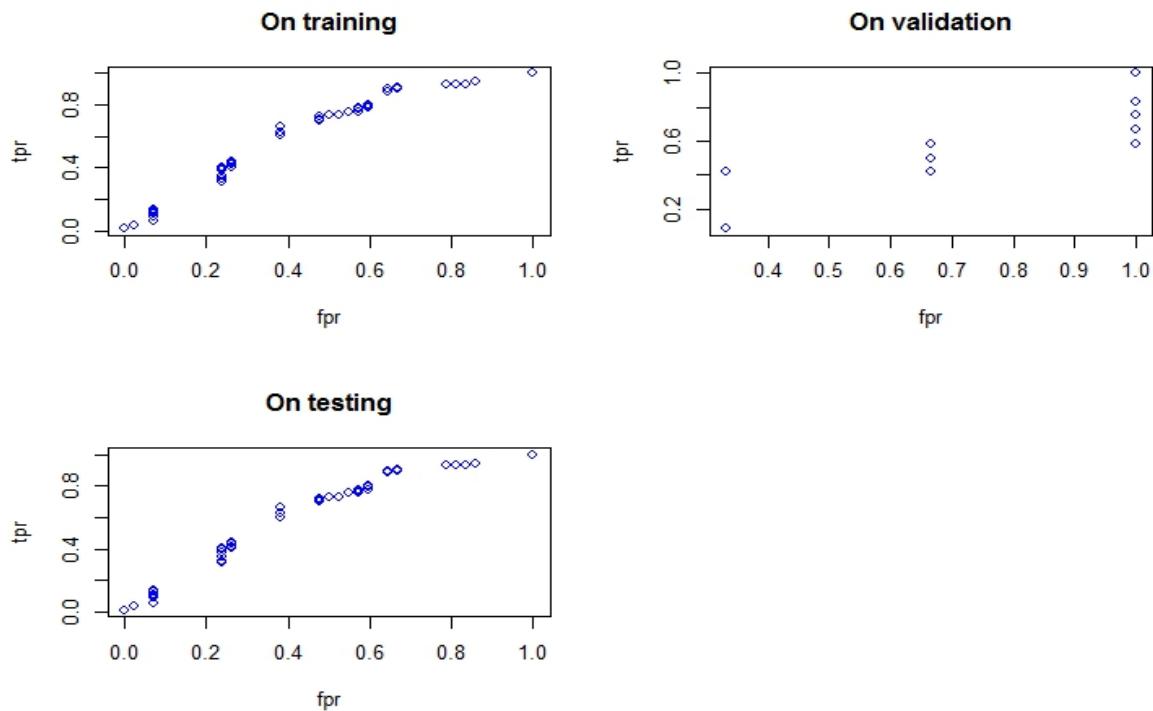


Figure.5.8 ROC curve for training, validation and testing.

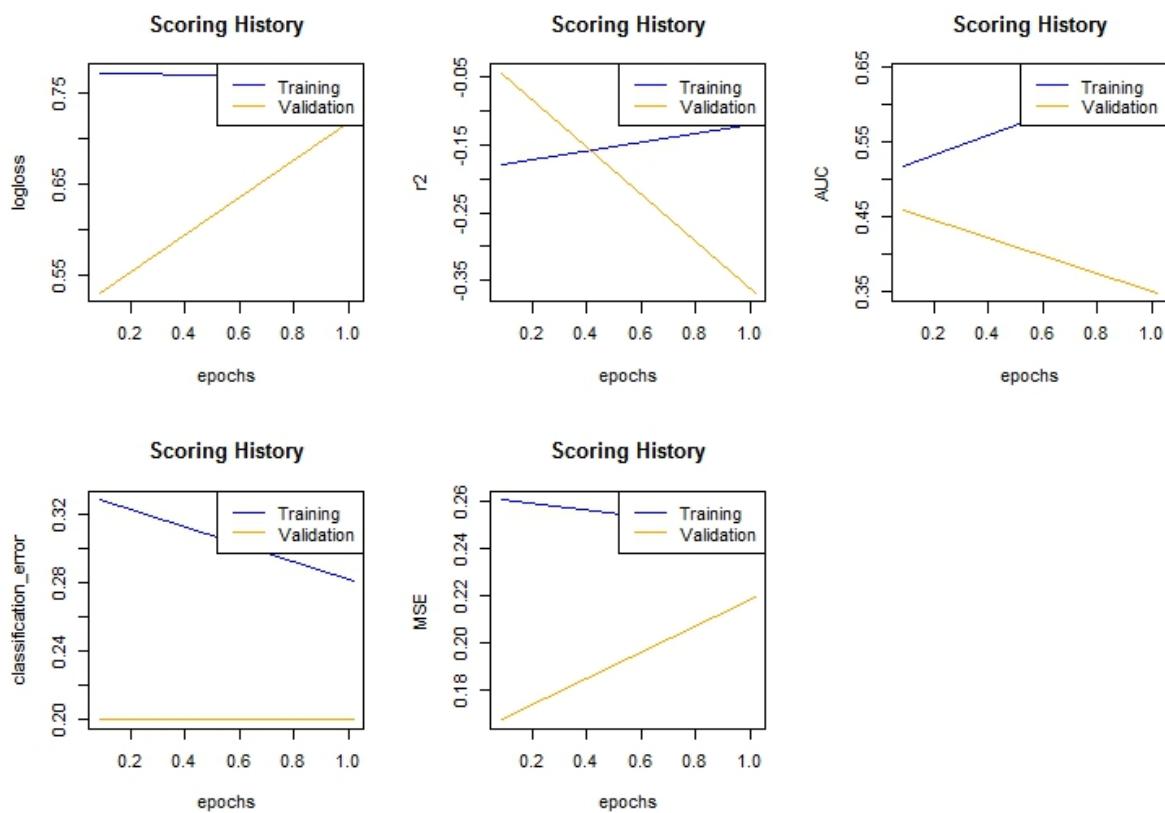


Figure.5.9 Scoring history of samples vs metrics of the deep learning model.

Figure 5.10 shows the test data predicted for the classification on the trained deep neural network. The model predicted single-locus and two-locus SNPs interactions associated to the disease. Top 20 highly ranked SNPs acting independently or due to two-way interactions are plotted as a bar chart in Figure 5.11.

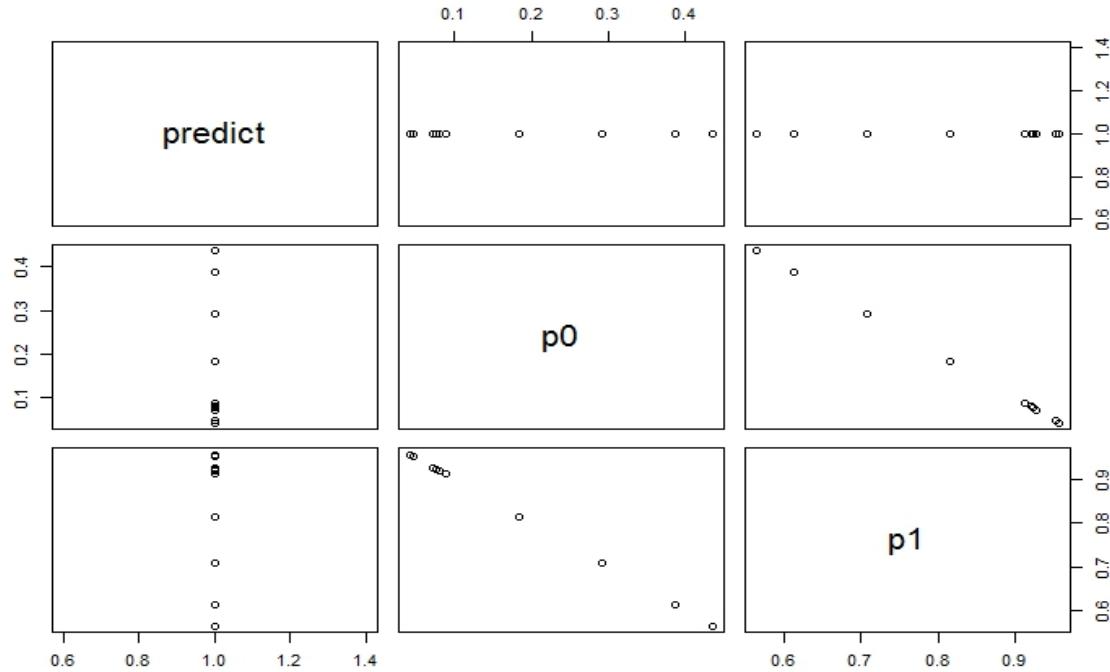


Figure 5.10 Predicting test data on the deep learning model.

The deep learning method is further analysed and compared with some of the previous approaches evaluated in the previous chapters, such as MDR, MDRAC, RF, LR, gradient boosted machines (GBM), naïve Bayes, CPAR, SVM, and NN. The accuracy of the trained deep learning model has the highest prediction accuracy of 77.01% when compared with other previous approaches. The best two-way SNP interaction identified by the model is.snp100012 (presence of Aa/aA) and.snp100019 (presence of Aa/aA). The java implementation of MDR (version-3.0.2) [17] is used to analyse the published data. The best two-locus model identified by MDR is.snp10001 and.snp10005, providing a training accuracy of 64.72 % and cross-validation consistency of 10 out of 10. The best two-locus SNP interaction identified by MDRAC (in chapter 3) is SNP 100033 and SNP 10005, whose prediction accuracy is 75.15 %. Even though, the accuracy is better than previous approaches, the accuracy of the model is low compared to the trained deep neural network model.

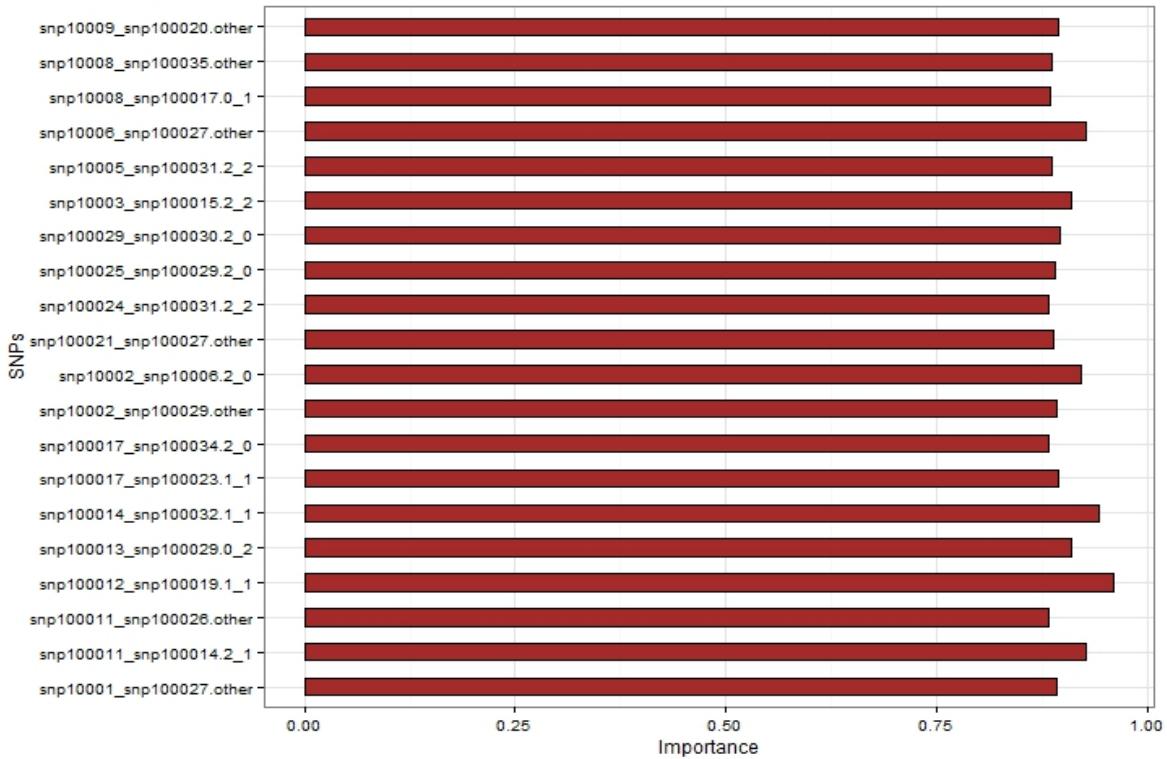


Figure 5.11 Top 20 two-locus SNP interactions identified by the deep learning model.

The dataset is also used to analyse LR using LogicFS [96] available for R. The accuracy of best two-way interaction model identified by LR is 60.91%. When the number of SNPs increased, it is observed that searching the interacting SNPs among all the possible logic trees became computationally hard. LR uses the simulated annealing as a searching algorithm by improving the variable selection. However, measuring the importance of interacting variables is restricted only to binary variables. GBM, RF, and Naïve Bayes are analysed using H2O interface [326] developed for the R environment. GBM built gradient boosted classification trees on the dataset. The prediction accuracy of the model is 69.47%. RF analysis determines the importance of variables that allows for possible interactions. The prediction accuracy of the model is 70.06%. The power of RF is reduced as it requires a marginal effect in at least one of the SNP interacting pair. However, it is observed that RF outperforms the prediction when the trees do not exhibit a correlation with each other. Figure 5.12, demonstrates the scoring metrics history of GBM, RF, LR, and the deep learning method during training and validation. Naïve Bayes classifier is also analysed, whose prediction accuracy is 54.14% with high classification error compared with other methods. Further, the dataset is evaluated on CPAR, SVM, and NN using weka tool whose accuracies to detect two-locus interacting

SNPs are 63.69%, 70.07% and 64.97% respectively. The accuracy of all these models is represented as a bar chart in Figure 5.13.

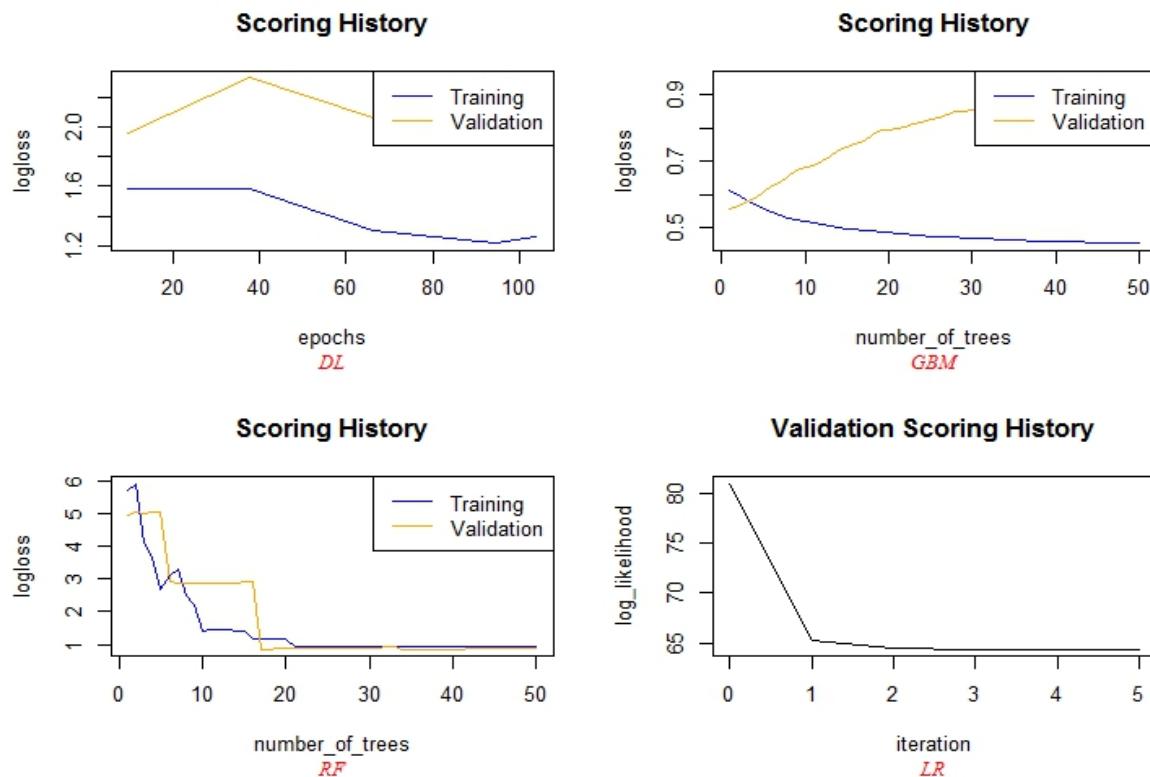


Figure.5.12 Comparing metrics of the deep learning model with GBM, RF, and LR.

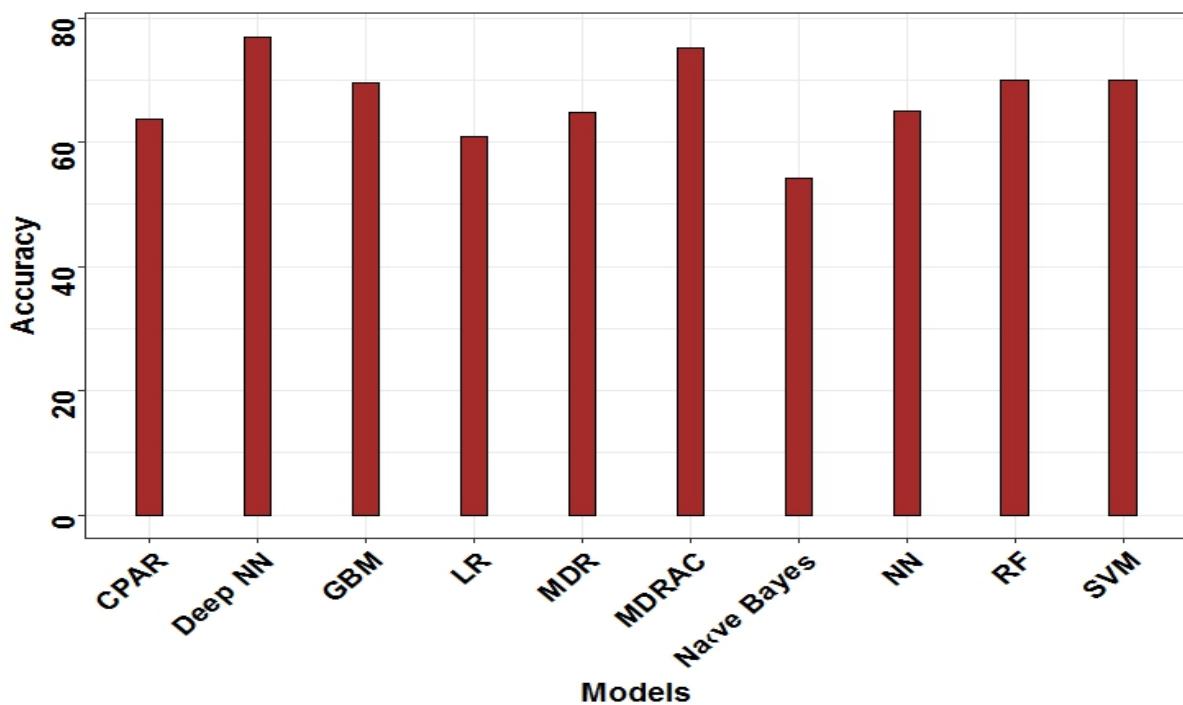


Figure.5.13 Accuracy of the deep learning model compared with the previous approaches.

## 5.4 Evaluations on sporadic breast cancer data for two-locus interactions

Multiple experiments are performed to evaluate the performance of the proposed method on sporadic breast cancer data (explained in Section 3.3). The models are evaluated by splitting the data (randomly into 80%, 10%, and 10% of data for training, validation, and testing respectively), and performing  $n$ -fold cross validation to determine the statistical significance of the models. The aim of this study is to confirm whether the trained DNN is an effective method to discover the two-locus SNP interactions. Furthermore, this study is evaluated and analysed by varying the parameters to identify the best model with low test set errors. The method is built and analysed in R using the H2o package [326].

### 5.4.1 Evaluation and analysis of the proposed method

The deep feed forward network trained in this chapter. It has a single input and output layers along with three hidden layers. Each layer in the hidden layers is trained with 1024 and 2048 computational units. The method processes 1000 epochs per 1000 iterations on 10 compute nodes. By default, the entire data is processed on every node locally by shuffling the training samples in each iteration. The model is trained with training samples of 320,000. The model took 17.658 seconds to train the data. The training speed of the model is estimated as 8122.098 samples/second. The validation error of the model is 0.294. Finally, test set error of the model is estimated as 0.661.

The model is validated by passing various non-linear activation functions such as, rectifier, tanh, maxout, rectifier with dropout, tanh with drop out, and maxout with drop out. Figure 5.14 compares the metrics of the model by varying the activation functions. Among all, tanh with dropout has high prediction accuracy with low classification error. Hence, it is chosen as an appropriate activation function to achieve better approximation. The input drop out ratio is set to 0.2 and hidden dropout ratios for the three hidden layers are each set to 0.5. The model is predicted using test data for the classification. The algorithm is executed ten times as 10-fold cross validation is performed. Each time a different split is omitted for testing the data. The model with a high CVC and low classification error is selected. The best model chosen from  $n$ -fold

cross validation, predicted the two-locus SNPs and SNPs with main effects that are highly related to breast cancer.

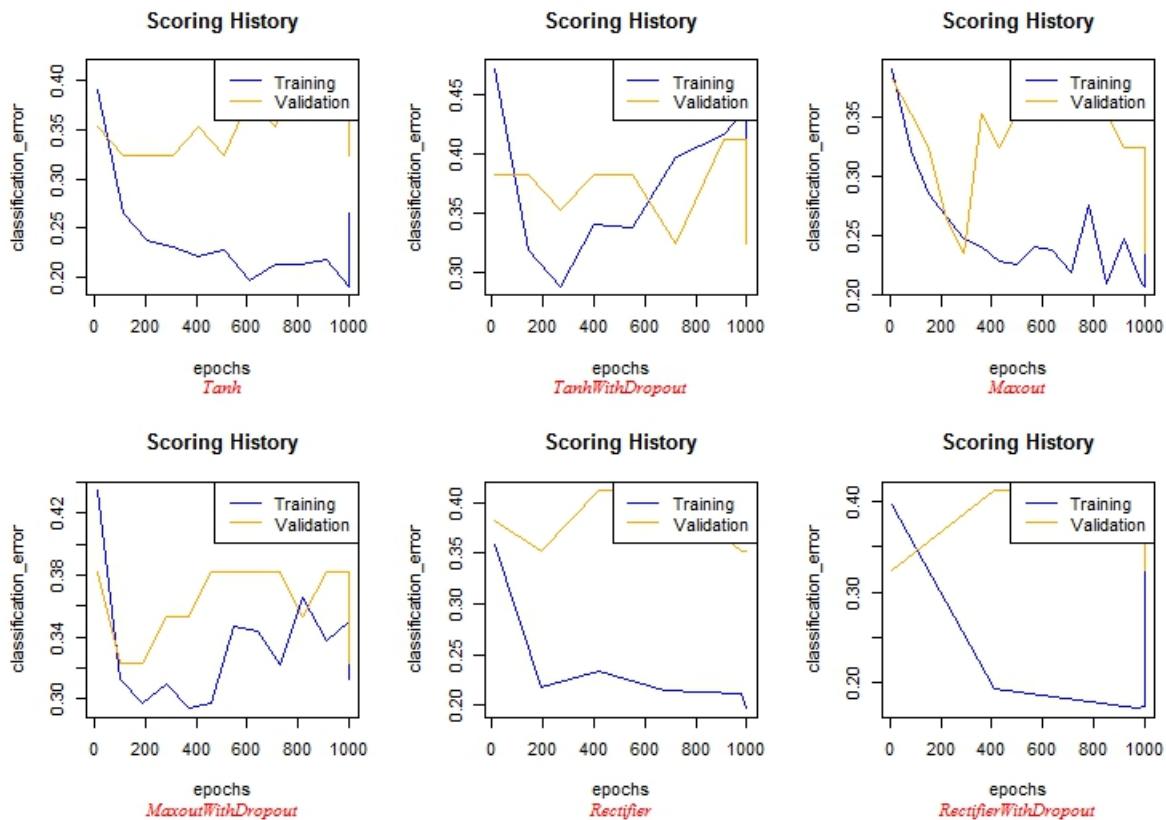


Figure 5.14 Model metrics the deep learning model by changing activation functions (tanh, tanhhwithdropout, maxout, maxoutwithdropout, rectifier, and rectifierwithdropout).

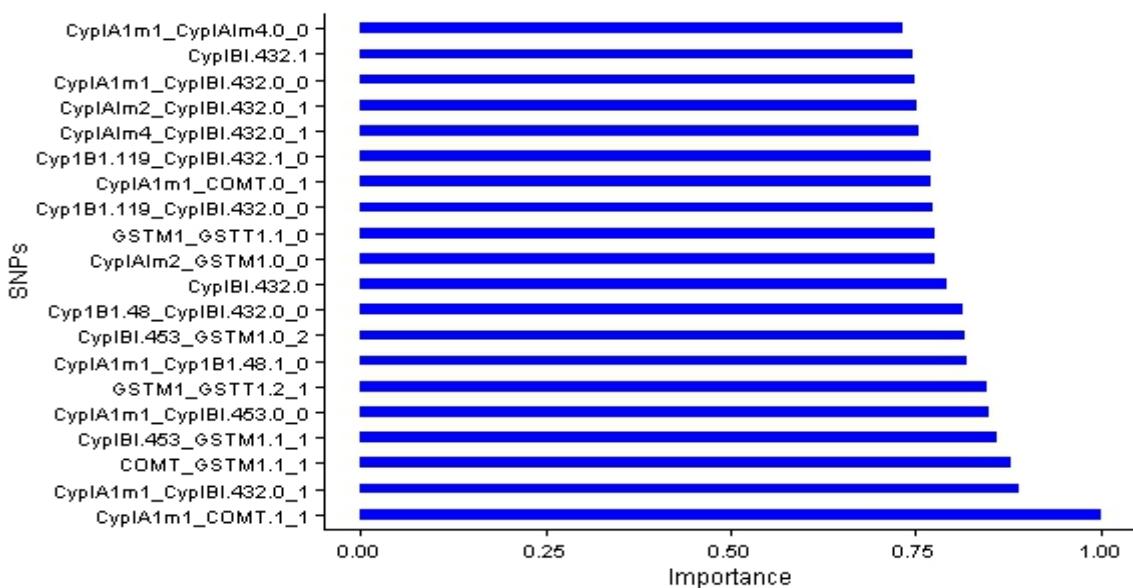


Figure 5.15 Top 20 single-locus, and two-locus SNP interactions

Figure 5.15 represents highly ranked top 20 interacting genetic polymorphisms in ascending order. The best model is validated by dividing the entire data into three parts with the probabilities of 0.8, 0.1 and 0.1 for training, validation, and testing respectively. The performance of the model for training, validating, and testing is shown in Figure 5.16.

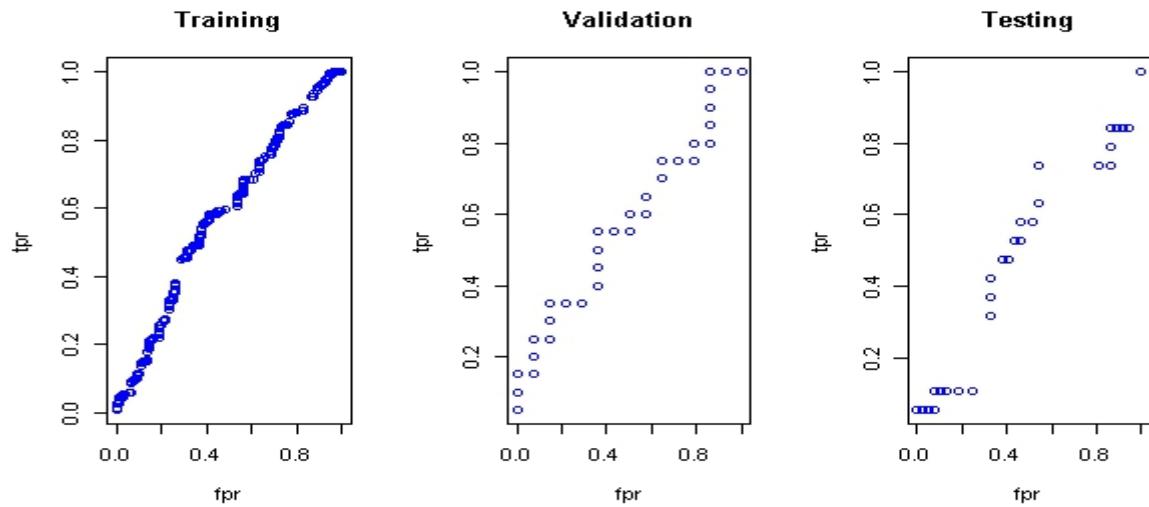


Figure 5.16 Performance of the model while training, validating, and testing

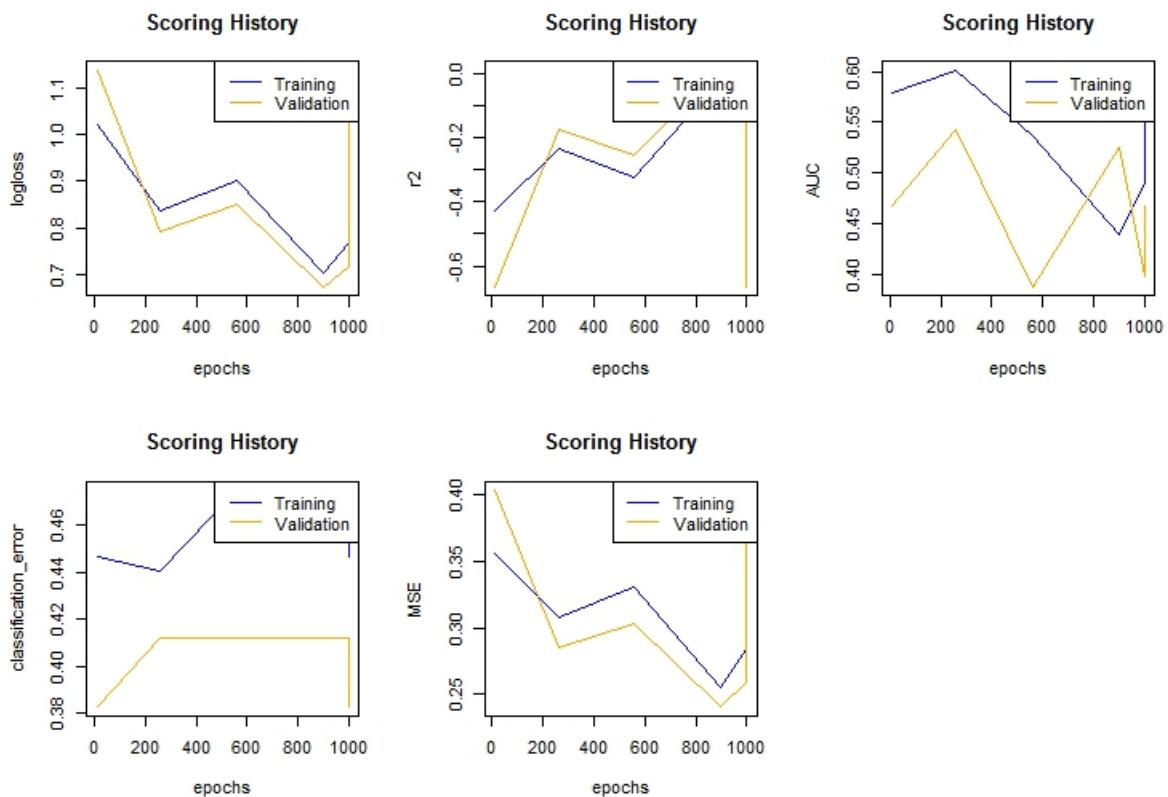


Figure 5.17 Scoring history of epochs vs metrics of the model.

Scoring history of the best model metrics are shown in Figure 5.17. The plots are outlined between timesteps on the x-axis (epochs and samples) and metrics on the y-axis (log-loss, classification error, r2, mse, and auc). It is noted that there is a fall in synchronisation and model convergence when training samples/epochs increased. It is also observed that the performance of the model is reduced drastically when the training samples are too small. This occurs due to the dominance of network communication between computational units increase by affecting the execution time of the algorithm.

## **5.4.2 Evaluation and analysis of the proposed method by changing the parameters**

A number of studies are carried out based on the study [329] to find the best model with improved accuracy and speed, by changing the parameters. The performance of the method is evaluated in terms of training speed, and training time for each set of parameters. In the first study, the model's network topologies (hidden layers) are changed by setting rest of the parameters with their default values. The model is evaluated for one, two, three, and four hidden layers with 100 epochs. It is observed that, the network with three hidden layers (each with 64 neurons), performed better than other models with a training speed 4661.972 samples/seconds. In the second study, all the models are trained with three hidden layers (with 2048 neurons) by varying scoring selections (score training samples, duty cycle, and interval). The best model identified in this study has a training speed of 41.586 samples/second, whose training time is 8.272 seconds. The third study is a comparative evaluation of manual and adaptive learning rates along with momentum. It is observed that the model with manual learning rate along with no momentum has performed well compared to other models. This is due to less usage of memory and low computational burden. Training samples per iteration is varied in the study four with the same 3 layer network. The model performed well as the number of training samples increased in terms of training speed (48.258 samples/second), and training time (6.631 seconds). Figure 5.18 shows the performance of the models evaluated under these four studies are plotted in terms of training speed and training time.

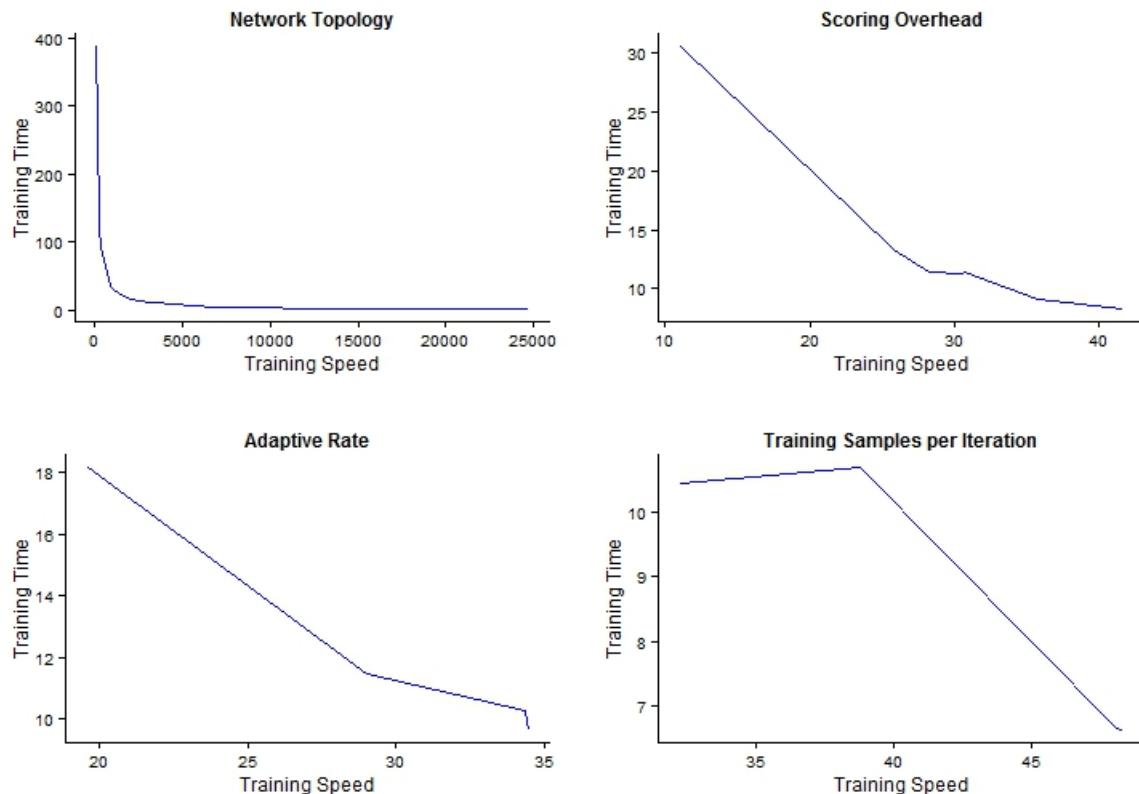


Figure 5.18 Training speed vs Training time (seconds) of the model by varying parameters which includes, Network topology, Scoring selections, Adaptive learning rate, and Training samples per iteration.

Study five is performed by varying activation functions (Rectifier, Rectifier with dropout, Tanh, Tanh with dropout, Maxout, and Maxout with dropout) for all the models evaluated in the above four studies. Rectifier, and Rectifier with dropout performed reasonably well compared to other activation functions. In study six, the performance of the method is observed with large deep networks. The networks are trained with four, and five hidden layers, under various parameter settings along with different activation functions. The best model has the highest training speed of 57.982 samples/second, and took 5.519 seconds to train the model. The best model identified by study seven has minimum test set error of 0.5, which took 2.53 seconds to train the two layered network. Study eight validates the benchmark model, and calculates AUC. Figure 5.19 illustrates studies five, six, seven, and eight as a line graph.

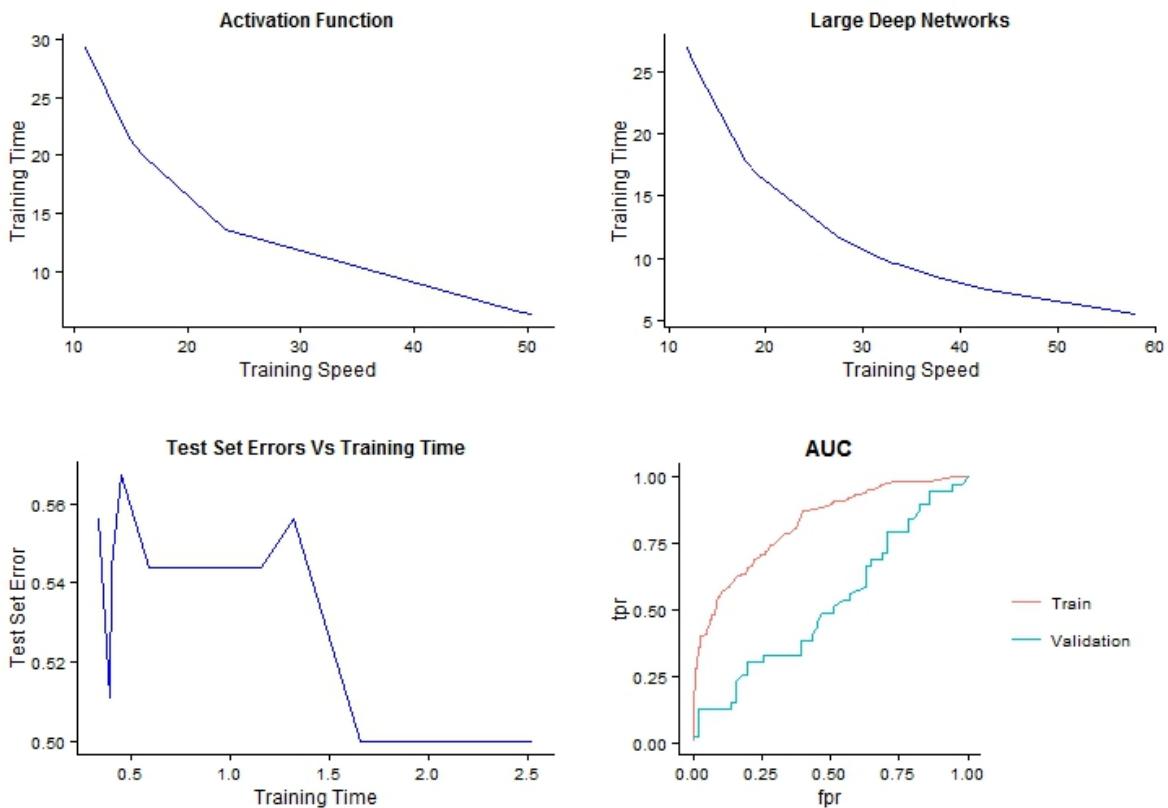


Figure 5.19 Training speed vs Training time (seconds) of the model by varying parameters (activation function, large deep networks), Training time vs Test set error to obtain the model with low test set error, and AUC in distributed model.

### 5.4.3 Prediction accuracy of the proposed method compared with previous methods

The DNN is evaluated and compared with a few pioneering works, such as Naïve Bayes, RF, GBM, LR, and MDR, to detect two-locus SNP interactions. Sporadic breast cancer data is used to evaluate the performance of the method in this study. Prediction accuracy of all these approaches is noted and tabulated in Table 5.1. It is observed that the prediction accuracy of the trained DNN is 68.78 %, which is higher than other current machine learning approaches. The network identifies the two-locus model (interaction between common homozygous CypIBI.453, and recessive homozygous GSTM1) that has high association to the disease. The MDR tool implemented in java, version-3.0.2, is used to analyse the data [17, 109]. The best single-locus model (GSTM1) is detected with a testing accuracy of 56.82%. MDR identified high interaction between Cyp1B1-432 and GSTM1, and demonstrated as the best two-locus model with

default parameters. The balanced accuracy of the two-locus model during testing is observed as 57.1% with the highest CVC (10 out of 10). The experimental results of the best two-locus models are represented in Figure 5.20. It is observed that the results were sensitive to the choice of random number seed.

**Table 5.1:** Metrics of the deep learning method compared with the previous approaches.

Methods	Accuracy	MSE	r2	Logloss	AUC	Gini
Deep learning	68.78	0.2750	-0.1001	1.192	0.7436	0.4873
RF	55.85	0.3117	-0.2471	1.2019	0.5139	0.0279
LR	67.07	0.2123	0.1508	0.6132	0.7360	0.4720
Naïve Bayes	62.68	0.3092	-0.2369	1.2403	0.6564	0.3128
GBM	65.85	0.2346	0.0616	0.6620	0.7297	0.4593

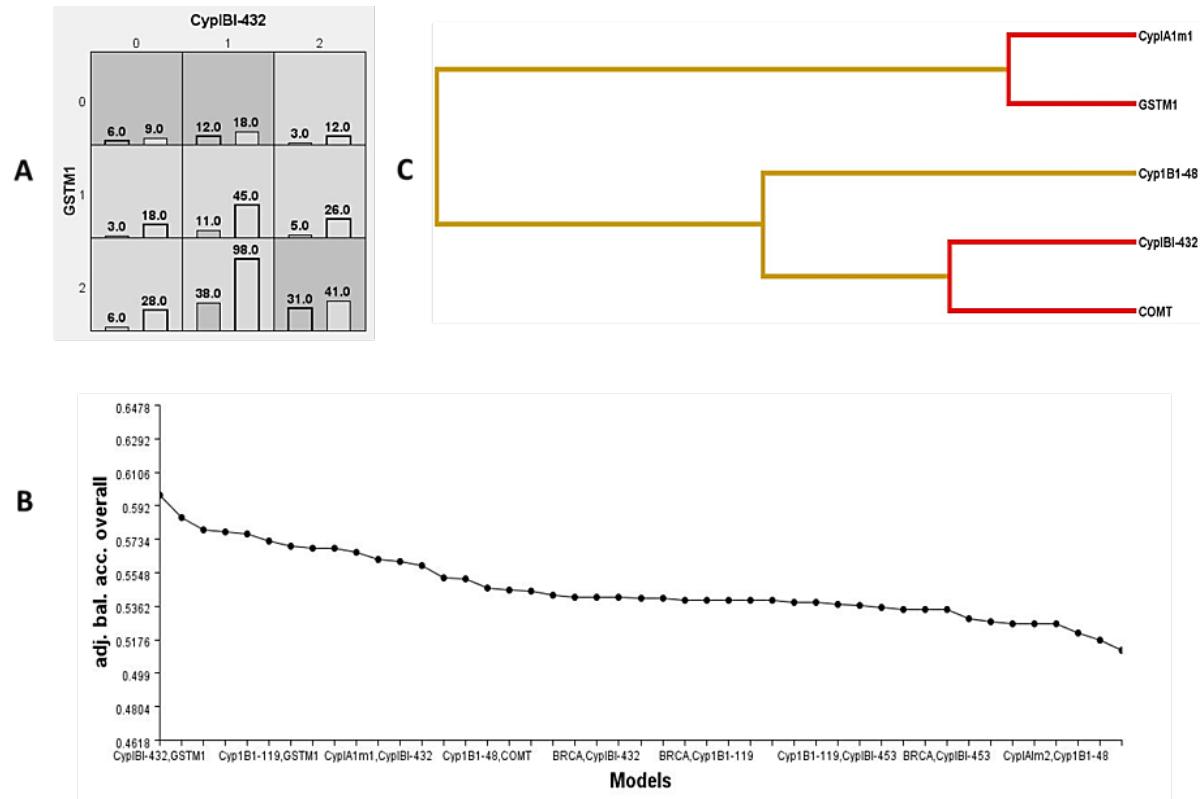


Figure 5.20 Performance of MDR. A) Allocation of high-risk and low-risk cells in the two-locus contingency table of genotype combinations. B) The line graph represents the overall adjusted balanced accuracy of top two-locus interacting models. C) An interaction dendrogram summarising the information gain associated with constructing pairs of SNPs. Shorter connections among nodes represents stronger synergistic (red lines) interactions.

The breast cancer dataset is also analysed by using LogicFS, developed for the R environment [96]. The best two-locus interaction model identified by LR is Cyp1B1.119\_1 & !GSTT1\_1. Furthermore, bagging version of LR is used to compute out-

of-bag error (OOB) rate (49.76%). It is observed that searching for SNP interactions among all the possible logic trees became computationally difficult as the number of SNPs grows. It is also observed that variable selection is improved by adopting the simulated annealing algorithm. Figure 5.21 illustrates the evaluations of LR using LogicFS.

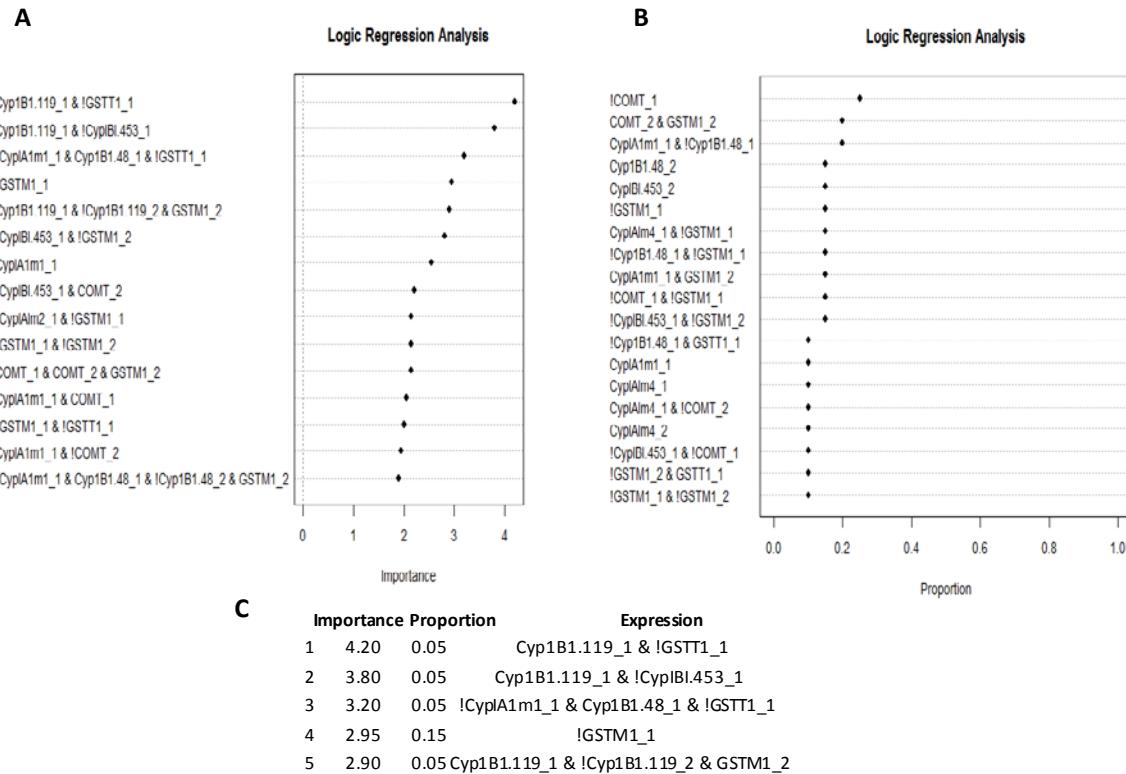


Figure 5.21 Performance of LR. A) The graph plots the variable importance of sporadic breast cancer data using LR analysis. B) The graph shows the proportions of models that contain the interactions of interest C) The top 5 important interactions were identified by Logic Regression (LR). The best interaction model identified by LR was Cyp1B1.119\_1 & !GSTT1\_1, where !GSTT1\_1 stands for NOT GSTT1\_1 (representing the complement of GSTT1\_1). Hence, the logical expression of the top model is interpreted as: Cyp1B1.119\_1 is of the homozygous variant genotype and !GSTT1\_1 is of the homozygous reference genotype.

Similarly, breast cancer data is evaluated and analysed on RF, GBM, and Naïve Bayes machine learning approaches by using H2o package implemented for the R environment [326]. The classification accuracy of RF is 55.85%. It is observed that RF analysis allowed the models to decide the importance for the variables that can have high possibility of Interactions. However, a high classification error during testing is observed compared with other methods. As noted in the previous section, prediction

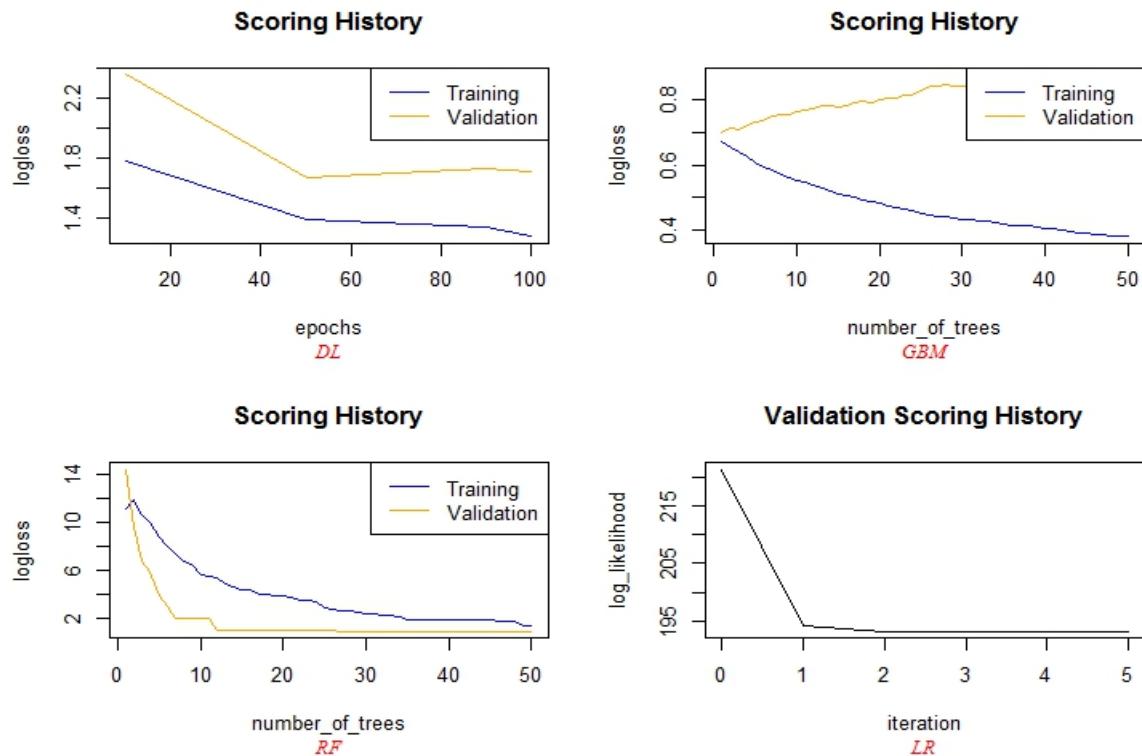


Figure.5.22 Scoring history of deep learning method, RF, GBM, and LR.

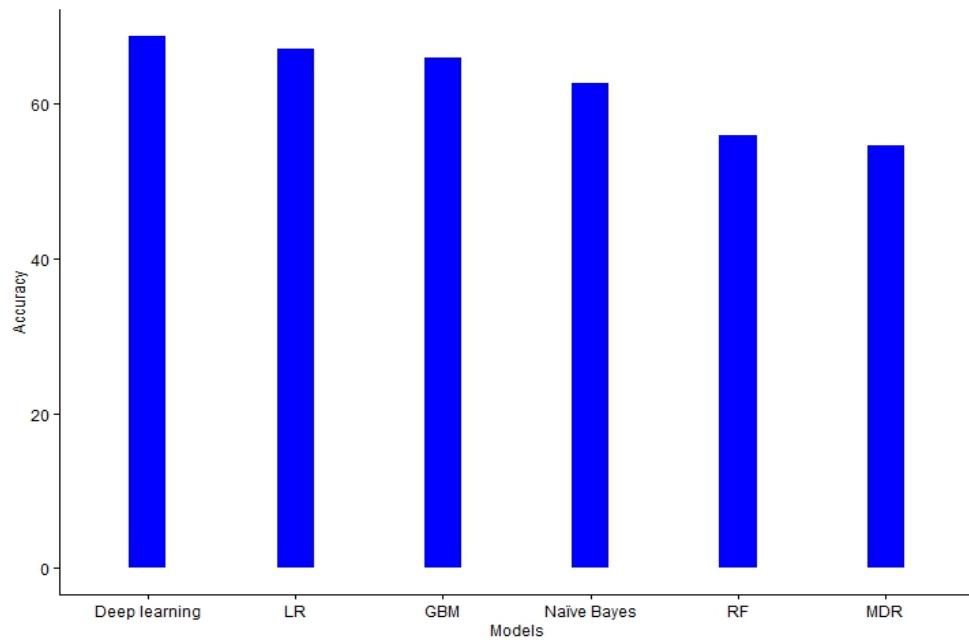


Figure.5.23 Prediction accuracy compared with other machine learning methods.

accuracy of RF analysis on breast cancer dataset degrades the performance of the models, as it required that at least one SNP in the SNP pair should have marginal effect. However, it has been noted that when there is no interrelationship between trees in the

forest, the prediction accuracy is significantly high. The classification accuracy of GBM is 65.85%, which is higher than all other previous methods excluding the proposed method. Furthermore, the Naïve Bayes classifier is evaluated and analysed using the same dataset. The prediction accuracy is observed as 62.68%, which is higher than for RF and MDR. The scoring history of RF, GBM, and LR is illustrated in Figure 5.22 and compared with the trained DNN. Figure 5.23, summarises the prediction accuracy of all the methods and presented as a bar chart.

## 5.5 Chapter Summary

In this chapter, a deep feedforward neural network is trained to identify two-locus interacting genetic variants responsible for a disease risk. The method is evaluated on the data obtained from a whole genome study, and sporadic breast cancer data to predict the performance of the method. The method identifies top twenty highly ranked two-locus SNP interactions, which are highly related to the disease manifestation. In depth studies are performed by varying parameters of the models to identify the best model. The experimental results demonstrated significant improvements in the prediction accuracy over the previous machine learning approaches. In the next chapter, studies will be performed by validating the performance of the method on the higher-order SNP interactions. Further studies will be performed to maximise the prediction accuracy by improving network learning, and optimising hyper-parameters.

# Chapter 6

## Improving Deep Learning Method for an intensive search of higher-order interactions

In the previous chapter, a multi-layered feed forward DNN is trained to detect two-locus SNP interactions. In this chapter, further studies are performed to validate the behavior of the network for higher-order interactions (three-locus or above) in high-dimensional data. The proposed method is extended for unsupervised learning. Features are learnt by discovering anomalies in the reduced representation of the original data. Furthermore studies are performed to maximise the performance of the method by improving the network learning, performing sensitivity analysis, and optimising hyper-parameters.

Section 6.1 improves the deep learning method proposed in the previous chapter by implementing dimensionality reduction, optimising hyper-parameters, and improving the learning. The performance of the method is studied for higher-order interactions and their combined effects in Section 6.2. Evaluations for unsupervised feature learning and optimising hyper-parameters are studied in Section 6.3 and Section 6.4 respectively. Section 6.5 discusses the experimental results of the models by improving the network learning.

---

This chapter is based on the following publications:

- S. Uppu and A. Krishna, "Improving strategy for discovering interacting genetic variants in association studies," in *International Conference on Neural Information Processing*, 2016, pp. 461-469: © 2016 Springer, "The original publication is available at [https://link.springer.com/chapter/10.1007/978-3-319-46687-3\\_51](https://link.springer.com/chapter/10.1007/978-3-319-46687-3_51) ."
- S. Uppu and A. Krishna, "Tuning Hyperparameters for Gene Interaction Models in Genome-Wide Association Studies," in *International Conference on Neural Information Processing*, 2017, pp. 791-801: © 2017 Springer, "The original publication is available at [https://link.springer.com/chapter/10.1007/978-3-319-70139-4\\_80](https://link.springer.com/chapter/10.1007/978-3-319-70139-4_80) ."
- S. Uppu and A. Krishna, "An intensive search for higher-order gene-gene interactions by improving deep learning model," in *18th IEEE International Conference on Bioinformatics and Bioengineering (BIBE)*, 2018, pp.104-109: © 2018 IEEE, "The original publication is available at <https://ieeexplore.ieee.org/document/8567466>".

## 6.1 Extended Method

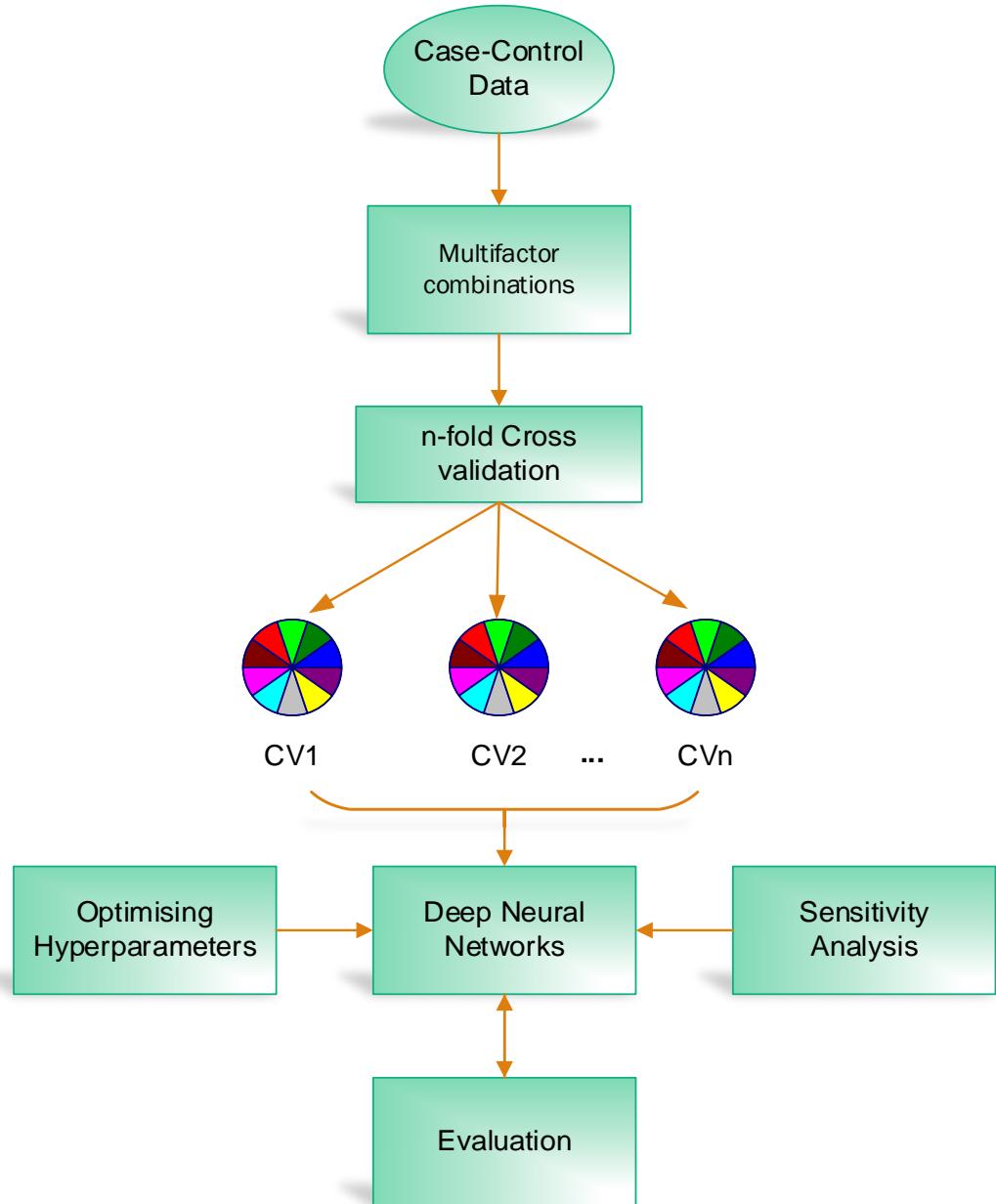


Figure 6.1 Overview of the extended deep learning method: ©2018 IEEE.

Figure 6.1 illustrates the extension to the deep learning method proposed in the previous chapter to detect higher-order SNP interactions (updated to Figure 5.7). The generalised sum of all weighted inputs to a neuron for single locus and higher-order loci, and cross entropy function are computed as explained in the previous chapter based on [321, 323]:

$$a_m = \sum_{i=1}^n w_i s_i + b_m \quad (6.1)$$

Where,  $s$  is a three dimensional input layer with  $n$  SNPs at  $n$  loci,  $s = \{s_1, s_2, s_3, \dots, s_n\}$ . By including multiplicative effect of two SNPs ( $s_i$  and  $s_j$ ), the sum of all weighted two-locus SNP inputs to a neuron is computed as:

$$a_{I_t} = \sum_{i=1}^n \sum_{j=i+1}^n \sum_{l=n+1}^{(\binom{n}{r+1})+n} w_l(s_i s_j) + b_{I_t} \quad (6.2)$$

The sum of all weighted multiplicative three-locus SNP inputs ( $s_i, s_j$ , and  $s_k$ ) to a neuron is computed as:

$$a_{I_{th}} = \sum_{i=1}^n \sum_{j=i+1}^n \sum_{k=j+1}^n \sum_{m=l+1}^{(\binom{n}{r+2})+l} w_m(s_i s_j s_k) + b_{I_{th}} \quad (6.3)$$

Similarly, the sum of all weighted higher-order inputs to a neuron is defined by:

$$\begin{aligned} a_{I_h} = & \sum_{i=1}^n w_i s_i + \sum_{i=1}^n \sum_{j=i+1}^n \sum_{l=n+1}^{(\binom{n}{r+1})+n} w_l(s_i s_j) + \sum_{i=1}^n \sum_{j=i+1}^n \sum_{k=j+1}^n \sum_{m=l+1}^{(\binom{n}{r+2})+l} w_m(s_i s_j s_k) \dots \\ & + \sum_{i=1}^n \sum_{j=i+1}^n \sum_{k=j+1}^n \dots \sum_{t=(n-1)+1}^n \sum_{u=\left(\binom{n}{r+1}\right)-1+1}^{\binom{n}{r}+\left(\left(\sum_{r=1}^n \binom{n}{r}\right)-1\right)} w_u(s_i s_j s_k \dots s_n) \\ & + w_{u+1} b_{I_h} \end{aligned} \quad (6.4)$$

Where,  $r = 1$ ,  $s_n$  takes three factor levels,  $s_n = [s_{n1}, s_{n2}, s_{n3}]$  at  $n^{th}$  input variable along with a bias  $b$ . A non-linear hyperbolic tangent activation function  $f(.)$  is applied on the weighted inputs of the neuron to keep the values in the manageable range as used in previous chapter. Where, tanh function converges faster as its range lies in between -1 to 1,  $\tanh(a_{I_h}) \in (-1,1)$ .

$$y_{I_h} = \tanh(a_{I_h}) = \frac{\exp(y_{I_h}) - \exp(-y_{I_h})}{\exp(y_{I_h}) + \exp(-y_{I_h})} \quad (6.5)$$

Gradient of  $\tanh(y_{I_h})$  is calculated as follows:

$$\nabla \tanh(a_{I_h}) = \left( 1 - \left( \frac{\exp(a_{I_h}) - \exp(-a_{I_h})}{\exp(a_{I_h}) + \exp(-a_{I_h})} \right)^2 \right) = 1 - \tanh^2(a_{I_h}) \quad (6.6)$$

The total error  $e$  of neurons for higher-order interactions in the output layer with multiplicative effects is calculated by using cross-entropy function for training samples  $t$  is given by:

$$e(W, B|t) = -\frac{1}{n} \sum_{c \in C} (\log y_{I_h} * y_{I_h}' + \log(1 - y_{I_h}) * (1 - y_{I_h}')) \quad (2.7)$$

Where  $y_{I_h}$  is the predicted or observed higher-order output obtained, and  $y_{I_h}'$  is the desired output.  $W$  is the collection  $\{wi\}_{1:N-1}$  and  $B$  is the collection  $\{bi\}_{1:N-1}$ .  $wi$  and  $bi$  are the weight matrix connecting layers  $i$  and  $i+1$  for  $N$  layers and the vector columns of biases for layer  $i+1$  respectively. Let  $C$  represent the output units and  $c$  represent the output layer. The gradient of cross-entropy function with respect to  $y_{I_h}$  is computed as:

$$\frac{\partial e}{\partial y_{I_h}} = \frac{t_{I_h} - y_{I_h}}{y_{I_h}(1 - y_{I_h})} \quad (6.8)$$

Where,  $t_{I_h}$  and  $y_{I_h}$  represents the predicted and actual higher-order interaction outputs respectively. The objective of the loss function is to adapt the weights by minimising the loss. A lock-free parallel version of SGD is used compute the partial derivative of each parameter with respect to cross entropy loss function [325]. It minimises the loss function by optimising the best fitting parameters using mini-batch strategy. The parameters  $(w_i, b_i)$  of the DNN model are updated for every epoch from time  $t$  to  $t+1$ .

$$w_i' \leftarrow w_i - \eta \frac{\partial e}{\partial w_i} \quad (6.9)$$

$$b_i' \leftarrow b_i - \eta \frac{\partial e}{\partial b_i} \quad (6.10)$$

Where,  $\frac{\partial e}{\partial w_i}$  is the partial derivative of loss function with respect to  $w_i$ ,  $\frac{\partial e}{\partial b_i}$  is the partial derivative of loss function with respect to  $b_i$ , and  $\eta$  is the learning rate,  $n > 0$ . Partial derivative of the loss function with respect to the layer's parameters  $w_i$  and it's input  $s_i$  are computed. Where,  $\frac{\partial e}{\partial s_i}$  is the partial derivative of the loss function with respect to the

input of  $i$  th layer, that is,  $(i - 1)$  th layer. Using chain rule of differentiation,  $\frac{\partial e}{\partial w_i}$  and  $\frac{\partial e}{\partial s_i}$  are computed as defined in the previous chapter.

## 6.1.1 Improving learning

The predictive performance of the extended deep learning method for higher-order interactions is maximized by improving the way deep networks learn (based on [323, 330]).

### 6.1.1.1 Regularization

Overfitting is a major challenge of DNNs as they are very difficult to train due to non-linear and high-dimensional parameters. It is reduced by using  $l_2$ regularization (or weight decay) technique. The regularized cross-entropy term is given by [323]:

$$e_R = -\frac{1}{n} \sum_{c \in C} (\log y_{I_h} * y_{I_h}' + \log(1 - y_{I_h}) * (1 - y_{I_h}')) + \frac{\mu}{2n} \sum_w w^2 \quad (6.31)$$

Sum of the squares of all the weights in the network is added to the cross-entropy function. Where,  $\mu$  is the regularization parameter,  $\mu > 0$  and  $n$  is the size of the training set. Regularization makes the network to learn with small weights by minimizing the original cross-entropy function. Cross-entropy function ( $e$ ) is minimized when  $\mu$  is small, and small weights are chosen when  $\mu$  is large. Partial derivative of  $e_R$  is computed by:

$$\frac{\partial e_R}{\partial w_i} = \frac{\partial e}{\partial w_i} + \frac{\mu}{n} w_i \quad (6.12)$$

The weights of the SGD learning rule are updated as:

$$w_i \leftarrow w_i - \frac{\eta}{m} \frac{\partial e}{\partial w_i} - \frac{\eta \mu w_i}{n} \quad (6.13)$$

$$w_i \leftarrow w_i \left(1 - \frac{\eta \mu}{n}\right) - \frac{\eta}{m} \frac{\partial e}{\partial w_i} \quad (6.14)$$

Where  $1 - \frac{\eta \mu}{n}$  is a rescaling factor that is referred as weight decay by shrinking the weight  $w_i$ ,  $\eta$  is the learning rate and  $m$  is mini-batch of training samples. Dropout is combined with  $l_2$ regularization to avoid overfitting by improving generalization in the

proposed DNNs. Activation of some the neurons in the network is suppressed with the probability of  $P$  in each forward pass. Usually,  $P < 0.5$  for hidden layers and  $P < 0.2$  for input layer.

### 6.1.1.2 Weight initialization

Weights are initialized using independent Gaussian random variables that are normalized with mean 0 and standard deviation 1. When  $y_m \gg 1$  or  $y_m \gg -1$ , the output of the neuron is close to 0 or 1 by attaining saturation. Hence, a small change in the weights does not affect the other neurons in the networks by leaving extremely small change in cross entropy function. This leads the network to learn weights slowly in SGD algorithm. Hence, the weights are further initialized by mean 0 and standard deviation  $\frac{1}{\sqrt{s_i}}$  to reduce the neuron's saturation by reducing the Gaussians [323]. It improves the speed of learning the proposed method by improving the model performance.

### 6.1.1.3 Learning rate and batch size

The learning rate and batch size in SGD are chosen carefully, which have great impact on training speed and performance of the model. When  $\eta$  is too large, the steps will be large that may overshoot the minimum and causes the algorithm go beyond the valley [323]. Choosing  $\eta$  too small slows down the algorithm. Learning rate such as, 0.0001, 0.001, 0.01, and 0.1 are explored [330]. The batch size of the training sample is increased to improve the speed of training, where else, it is decreased to improve the memory efficiency. Hence, the smaller learning rate  $\eta$  and the larger batch size  $m$  are chosen.

### 6.1.1.4 Momentum and adaptive learning rate

The current parameter  $w$  is updated by a fraction of previous iteration of back propagation to improve training. Momentum based SGD changes the velocity vector  $v$  instead of directly changing the position. The weights are updated by using Nesterov accelerated gradient (NAG) method from time  $t$  to  $t + 1$  as follows [331]:

$$v_{t+1} = \vartheta v_t - \eta \nabla e(w_t + \vartheta v_t) \quad (6.15)$$

$$w_{t+1} = w_t + v_{t+1} \quad (6.16)$$

Where  $\vartheta$  ( $\vartheta \in [0,1]$ ) is a momentum parameter that sets to a typical value  $\vartheta = 0.9$ , and  $\eta > 0$  represents the learning rate. Momentum parameter controls oscillations by fastening the convergence to local minima. Adaptive learning is implemented by using ADADELTA, which combines learning rate annealing and momentum training to avoid slow convergence [332]. Typical values 0.9 and  $10^{-9}$  are set to two hyper-parameters of adaptive learning  $\rho$  and  $\tau$  respectively.

## 6.1.2 Optimising hyper-parameters

The objective of the deep learning algorithm is to find a function that minimizes classification error. It produces the function  $HP_G$  through the optimization of training criteria with respect to hyper-parameters. Traditionally, a number of models are trained manually with various combinations of hyper-parameters in the previous chapter. The performances of all the models are compared to find the best model. This kind of manual search becomes tedious when the desired values of the network increases. Reproducing the result is one of the major drawbacks of manual search that we observed in the previous chapter. Choosing set of configurations is a critical step in the hyper-parameter optimization of the method. Additionally, achieving the optimal hyper-parameters is more complex when dealing with multi-dimensional data. Hence, automatic and reproducible approaches for tuning hyper-parameters are required. The most widely used strategies such as, grid, and random search are evaluated in this study to maximize the prediction accuracy [333].

### 6.1.2.1 Grid Search

Grid search (Cartesian search) [333] exhaustively builds models for every combination of hyper-parametric values specified. That is, deep learning algorithm is trained accordingly with a number of configurations of hyper-parameters. Bounds and steps between values of hyper-parameters are specified to form a grid of configurations. The search begins at limited grid with relatively large steps between the parameter values by making the grid finer at the best configuration. This searching process further continues on a new grid till it searches all the configurations. Finally, the hyper-parameter configuration that provides the best performance is chosen as the optimal value. Grid search is expensive as it searches exhaustively for all the configurations.

Consider  $n$  hyperparameters each with 5 values, which make  $5^n$  configurations in total. Hence, number of configurations in grid search is represented as  $G$  elements below:

$$HP_G = \prod_{g=1}^G |H^s| \quad (6.17)$$

The product over  $G$  set leads grid search to suffer from issues of dimensionality as the number of values exponentially increases with the number of hyperparameters. Hence, grid search is feasible for a small number of configurations. However, grid search can compute in parallel to improve the computational power.

### 6.1.2.2 Random Grid Search

In random grid search [333], the grid of configurations of hyper-parameters is searched randomly. The hyper-parameter values are chosen within the specified values without repeats by building the models sequentially. Optimal combination of parameter values is identified to maximize the predictive performance of the model. Adding new configurations or removing failed configurations are feasible in the random search. Hence, random search is simple, and as effective as full grid search. The number of trials is much less than grid search with comparable performance.

### 6.1.3 Sensitivity analysis

Sensitivity analysis in DNN refers the behaviour of the output by the influence of input and weight perturbations in the network. That is, it obtains the relationship between the two-locus interaction variables and the response variable rather than a categorical description. Lekprofile method [334, 335] is implemented to analysis the sensitivity of the improved deep learning method for two-locus interactions. This function returns the predicted output (as a plot) by evaluating the effect of each two-locus variables across the range of values by keeping other two-locus variables constant. The main idea of this method is to construct a matrix with the range between minimum and maximum values of all the input variables. The obtained matrix is then used to predict the values of the output variables.

### 6.1.4 Dimensionality reduction

Evaluating multi-locus combinations of SNPs in genome-wide data increase

exponentially. Finding an optimal combination among an unusually large number of combinations is not feasible within the existing computational techniques. That is, for a study of 300,000 SNPs in GWA, there will be  $4.5 \times 10^{10}$  two-way interactions and  $4.5 \times 10^{15}$  three-way interactions to be examined [2]. This computational challenge has been addressed in this method by implementing Principal Component Analysis (PCA) [336] in the preprocessing step. PCA is used to reduce high-dimensional genome-wide data into a low-dimensional data, and applied to the proposed deep learning method to detect higher-order SNP interactions associated with a disease. It is an unsupervised learning algorithm, which transforms high-dimensional data linearly into a new set of low dimensional data with uncorrelated features. Further, the extended method is trained for unsupervised feature learning, and to detect anomalies in the data by using deep autoencoder [311]. It learns nonlinearly from the reduced representation of the actual data. The model is trained on a training data by ignoring class labels. Reconstruction error is computed between the output and input layers with anomaly detection to determine the outliers for higher-order interacting SNP test data.

### 6.1.5 Case-control Data

The extended deep learning method is evaluated on sporadic breast cancer, and hypertension data explained in Section 3.3, and Section 4.2.3 respectively. Further, the method is also analysed on a chronological dialysis patient data.

Chronic dialysis patients data comprise of 897 samples, among which, there are 193 patients and 704 healthy controls [337]. The study is conducted at Kaohsiung Chang Gung Memorial Hospital, based on unrelated Taiwanese of ethnic Chinese background with 390 men and 507 women, whose average age is 50.45 years. This study investigates the prevalence of SNPs in the mitochondrial D-loop. Large scale bioinformatic analysis and D-loop sequencing, which stretches between nt16180–16195 and nt303–315, are used to identify 77 SNPs (whose frequencies greater than 1%) that matched with the positions given in the Revised Cambridge Reference Sequence (CRS). Chi-square tests are performed to compare distributions of SNPs between cases and controls. It was observed that nine SNPs (statistically significant SNPs,  $p < 0.05$ , that are selected by logistic regression), such as, SNP5 (16108Y), SNP17 (16172Y), SNP21 (16223Y), SNP34 (16274R), SNP35 (16278Y), SNP55 (16463R), SNP56 (16519Y), SNP64 (185R), and SNP65 (189R) in D-loop of CRS, are frequently

occurring among patients. It was also observed that chronical dialysis patients had low cholesterol, body mass, and blood thiols compared with healthy controls.

## 6.2 Evaluations for higher-order interactions

The extended model is evaluated for one-locus to ten-locus SNP interactions individually on sporadic breast cancer data [17] and analysed in R [311]. Figure 6.2 plots true positives vs false positives from one-locus to ten-locus SNP interactions. True positive rate constantly rises as false positive rate increases. Furthermore, the method is evaluated by binding all higher-order combinations together and predicted on test data for identifying interacting SNPs responsible for breast cancer. Table 6.1 represents the top 10 highly ranked SNPs due to main effect (single-locus). Table 6.2 to 6.10 shows the top 10 highly ranked SNP interactions (two to ten loci).

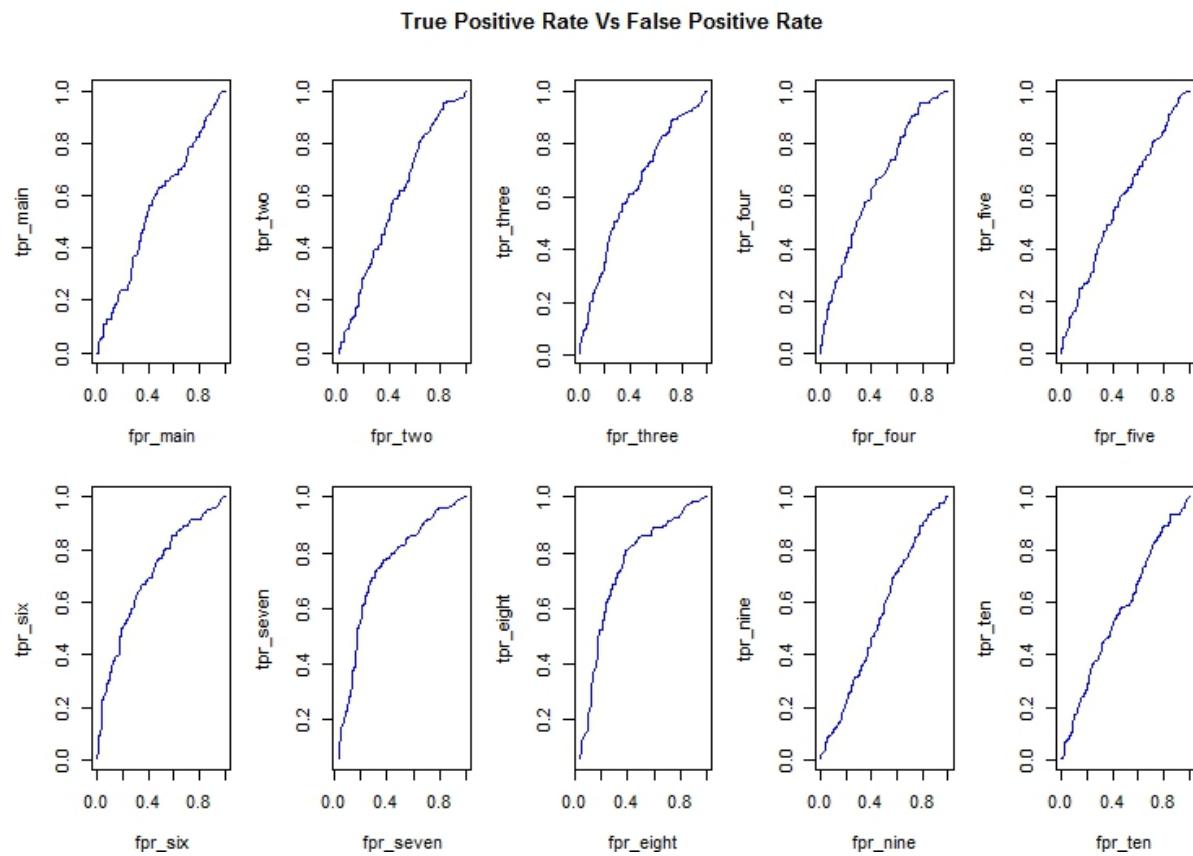


Figure 6.2 ROC plots for one-locus to ten-locus interactions

**Table 6.1:** Top 10 single-locus SNPs.

Single-locus	Relative importance	Scaled importance	Percentage
CypIAIm2.0	1	1	0.038374

GSTM1.2	0.994644	0.994644	0.038169
CypIA1m1.0	0.942725	0.942725	0.036176
CypIA1m2.1	0.930519	0.930519	0.035708
Cyp1B1.119.2	0.921988	0.921988	0.03538
Cyp1B1.119.0	0.904081	0.904081	0.034693
CypIA1m4.2	0.886577	0.886577	0.034022
CypIBI.453.2	0.865571	0.865571	0.033215
CypIBI.453.1	0.865297	0.865297	0.033205
Cyp1B1.48.2	0.851473	0.851473	0.032674

**Table 6.2:** Top 10 two-locus SNPs.

Two-locus	Relative importance	Scaled importance	Percentage
CypIBI.432_GSTM1.0_0	1	1	0.003316
CypIA1m4_COMT.0_1	0.969585	0.969585	0.003215
CypIA1m2_Cyp1B1.119.0_2	0.962495	0.962495	0.003192
CypIA1m1_COMT.0_1	0.955504	0.955504	0.003169
Cyp1B1.119_GSTM1.other	0.948031	0.948031	0.003144
CypIA1m4_CypIBI.453.1_1	0.944119	0.944119	0.003131
CypIA1m1_COMT.1_2	0.93672	0.93672	0.003106
CypIA1m1_CypIA1m4.2_0	0.936491	0.936491	0.003106
Cyp1B1.119_GSTM1.2_0	0.933997	0.933997	0.003097
CypIA1m2_Cyp1B1.119.1_1	0.933644	0.933644	0.003096

**Table 6.3:** Top 10 three-locus SNPs.

Three-locus	Relative importance	Scaled importance	Percentage
GSTM1_CypIA1m1_Cyp1B1.119.1_2_0	1	1	0.000541
CypIA1m4_CypIBI.453_COMT.0_0_0	0.999697	0.999697	0.000541
Cyp1B1.119_CypIBI.453_COMT.0_1_2	0.997203	0.997203	0.00054
CypIA1m1_Cyp1B1.48_Cyp1B1.119.0_1_1	0.993222	0.993222	0.000538
GSTM1_Cyp1B1.119_CypIBI.432.0_1_1	0.983041	0.983041	0.000532
Cyp1B1.48_CypIBI.453_COMT.0_0_2	0.981969	0.981969	0.000532
CypIA1m2_CypIBI.432_CypIBI.453.0_2_1	0.970226	0.970226	0.000525
CypIA1m2_CypIA1m4_COMT.0_0_1	0.968185	0.968185	0.000524
CypIA1m2_Cyp1B1.119_CypIBI.432.0_1_1	0.962698	0.962698	0.000521
GSTT1_CypIA1m4_Cyp1B1.48.1_1_2	0.954735	0.954735	0.000517

**Table 6.4:** Top 10 four-locus SNPs.

Four-locus	Relative importance	Scaled importance	Percentage
GSTT1_Cyp1B1.48_Cyp1B1.119_CypIBI.453.0_0_0_0	1	1	0.000187
CypIA1m2_CypIA1m4_CypIBI.432_CypIBI.453.0_0_1_0	0.993707	0.993707	0.000185
CypIA1m1_CypIA1m2_Cyp1B1.119_CypIBI.453.0_0_1_0	0.937532	0.937532	0.000175
GSTM1_GSTT1_CypIA1m2_COMT.1_0_0_1	0.917958	0.917958	0.000171

GSTM1_CypIA1m2_CypIA1m4_CypIBI.432.2_0_0_1	0.894873	0.894873	0.000167
GSTM1_CypIA1m2_CypIBI.432_CypIBI.453.2_0_1_0	0.888379	0.888379	0.000166
GSTM1_CypIA1m4_CypIBI.432_CypIBI.453.2_0_1_0	0.887178	0.887178	0.000166
CypIA1m1_Cyp1B1.48_Cyp1B1.119_CypIBI.453.0_1_1_0	0.878258	0.878258	0.000164
GSTT1_CypIA1m1_CypIA1m2_CypIBI.453.0_0_0_0	0.877966	0.877966	0.000164
GSTT1_CypIA1m4_CypIBI.432_COMT.0_0_1_2	0.873926	0.873926	0.000163

**Table 6.5:** Top 10 five-locus SNPs.

Five-locus	Relative importance	Scaled importance	Percentage
CypIA1m4_Cyp1B1.119_CypIBI.432_CypIBI.453_COMT.0_0_2_1_2	1	1	0.000861
CypIA1m4_Cyp1B1.48_Cyp1B1.119_CypIBI.432_CypIBI.453.0_1_0_2_0	0.986704	0.986704	0.000849
CypIA1m2_Cyp1B1.119_CypIBI.432_CypIBI.453_COMT0.0_0_0_0_2	0.966052	0.966052	0.000832
CypIA1m2_Cyp1B1.119_CypIBI.432_CypIBI.453_COMT0.0_0_1_0_1	0.961081	0.961081	0.000827
CypIA1m4_Cyp1B1.48_Cyp1B1.119_CypIBI.432_CypIBI.453.1_2_2_1_1	0.956407	0.956407	0.000823
CypIA1m4_Cyp1B1.48_CypIBI.432_CypIBI.453_COMT0.0_1_2_1_0	0.950821	0.950821	0.000818
CypIA1m2_Cyp1B1.119_CypIBI.432_CypIBI.453_COMT.0_1_2_0_1	0.947256	0.947256	0.000815
CypIA1m2_Cyp1B1.119_CypIBI.432_CypIBI.453_COMT.0_1_2_1_2	0.946094	0.946094	0.000814
Cyp1B1.48_Cyp1B1.119_CypIBI.432_CypIBI.453_COMT.1_1_1_1_1	0.939614	0.939614	0.000809
CypIA1m4_Cyp1B1.119_CypIBI.432_CypIBI.453_COMT0.0_0_2_2_0	0.939176	0.939176	0.000808

**Table 6.6:** Top 10 six-locus SNPs.

Six-locus	Relative importance	Scaled importance	Percentage
GSTM1_CypIA1m1_CypIA1m2_CypIA1m4_CypIBI.453_COMT.2_0_0_0_0_1	1	1	9.37E-05
GSTM1_CypIA1m1_CypIA1m2_CypIA1m4_Cyp1B1.48_Cyp1B1.119.2_0_0_0_0_0	0.872903	0.872903	8.18E-05
GSTM1_GSTT1_CypIA1m1_CypIA1m2_CypIA1m4_COMT.2_0_0_0_0_1	0.870073	0.870073	8.15E-05
GSTM1_CypIA1m1_CypIA1m2_CypIA1m4_Cyp1B1.119_CypIBI.453.1_0_0_0_0_1	0.854312	0.854312	8.00E-05
GSTM1_GSTT1_CypIA1m2_CypIA1m4_Cyp1B1.48_Cyp1B1.119.1_0_0_0_0_0	0.84855	0.84855	7.95E-05
GSTM1_GSTT1_CypIA1m1_CypIA1m4_Cyp1B1.48_Cyp1B1.119.1_0_0_0_0_0	0.811257	0.811257	7.60E-05
GSTM1_GSTT1_CypIA1m2_CypIA1m4_Cyp1B1.48_CypIBI.453.1_0_0_0_0_1	0.804048	0.804048	7.53E-05
GSTT1_CypIA1m1_CypIA1m4_Cyp1B1.48_Cyp1B1.119_CypIBI.432.0_0_0_0_0_1	0.792282	0.792282	7.42E-05
GSTT1_CypIA1m1_CypIA1m2_CypIA1m4_CypIBI.432_COMT.0_0_0_0_0_1	0.784495	0.784495	7.35E-05
GSTT1_CypIA1m1_CypIA1m2_CypIA1m4_CypIBI.453_	0.781435	0.781435	7.32E-05

COMT.1_0_0_0_0_1			
------------------	--	--	--

**Table 6.7:** Top 10 seven-locus SNPs.

Seven-locus	Relative importance	Scaled importance	Percentage
GSTT1_CypIA1m1_CypIAIm2_Cyp1B1.48_Cyp1B1.119_CypIBI.432_COMT.0_0_0_1_1_1_1	1	1	9.88E-05
GSTM1_GSTT1_CypIA1m1_CypIAIm2_CypIBI.432_CypIBI.453_COMT.2_0_0_0_0_0_1	0.915253	0.915253	9.05E-05
GSTM1_GSTT1_CypIA1m1_CypIAIm2_Cyp1B1.48_CypIBI.432_CypIBI.453.0_1_0_0_1_1_0	0.903594	0.903594	8.93E-05
GSTM1_GSTT1_CypIAIm2_CypIAIm4_Cyp1B1.48_CypIBI.453_COMT.2_1_0_0_1_0_1	0.88419	0.88419	8.74E-05
GSTT1_CypIA1m1_CypIAIm4_Cyp1B1.119_CypIBI.432_CypIBI.453.COMT.0_0_0_0_0_1	0.874434	0.874434	8.64E-05
GSTM1_CypIA1m1_CypIAIm2_CypIAIm4_Cyp1B1.48_Cyp1B1.119_COMT.2_0_0_0_1_1_1	0.852067	0.852067	8.42E-05
GSTM1_GSTT1_CypIA1m1_CypIAIm2_CypIAIm4_Cyp1B1.119_COMT.1_1_1_1_0_1_1	0.851814	0.851814	8.42E-05
GSTM1_GSTT1_CypIA1m1_Cyp1B1.48_Cyp1B1.119_CypIBI.432_CypIBI.453.2_0_0_1_1_1_0	0.850934	0.850934	8.41E-05
GSTM1_CypIA1m1_CypIAIm2_Cyp1B1.119_CypIBI.432_CypIBI.453.COMT.1_0_0_1_1_0_0	0.845271	0.845271	8.35E-05
GSTT1_CypIA1m1_CypIAIm2_CypIAIm4_Cyp1B1.48_CypIBI.432_COMT.0_0_0_0_0_1_0	0.838415	0.838415	8.29E-05

**Table 6.8:** Top 10 eight-locus SNPs.

Eight-locus	Relative importance	Scaled importance	Percentage
GSTT1_CypIA1m1_CypIAIm2_CypIAIm4_Cyp1B1.48_Cyp1B1.119_CypIBI.432_CypIBI.453.0_0_0_0_1_1_1_0	1	1	0.000186
GSTM1_GSTT1_CypIA1m1_CypIAIm4_Cyp1B1.48_CypIBI.432_CypIBI.453_COMT.1_0_0_0_1_1_0_2	0.96285	0.96285	0.000179
GSTT1_CypIA1m1_CypIAIm2_CypIAIm4_Cyp1B1.48_CypIBI.432_CypIBI.453_COMT.0_0_0_0_1_2_0_0	0.941824	0.941824	0.000175
GSTM1_GSTT1_CypIA1m1_CypIAIm4_Cyp1B1.119_CypIBI.432_CypIBI.453_COMT.1_1_1_0_2_1_0_2	0.933696	0.933696	0.000173
GSTM1_GSTT1_CypIAIm2_CypIAIm4_Cyp1B1.48_Cyp1B1.119_CypIBI.432_COMT.2_1_0_0_0_0_1_2	0.93294	0.93294	0.000173
GSTM1_GSTT1_CypIA1m1_CypIAIm4_Cyp1B1.48_CypIBI.432_CypIBI.453_COMT.2_0_0_0_1_1_0_1	0.929989	0.929989	0.000173
GSTM1_GSTT1_CypIA1m1_CypIAIm2_CypIAIm4_Cyp1B1.119_CypIBI.453_COMT.2_0_1_1_0_1_1_2	0.919637	0.919637	0.000171
GSTM1_CypIA1m1_CypIAIm2_CypIAIm4_Cyp1B1.119_CypIBI.432_CypIBI.453_COMT.2_0_0_0_2_1_0_2	0.918979	0.918979	0.000171
CypIA1m1_CypIAIm2_CypIAIm4_Cyp1B1.48_Cyp1B1.119_CypIBI.432_CypIBI.453_COMT.2_1_0_2_2_1_0_0	0.914692	0.914692	0.00017
GSTM1_GSTT1_CypIA1m1_CypIAIm2_CypIAIm4_Cyp1B1.119_CypIBI.432_CypIBI.453.2_0_0_0_1_0_1_0	0.912552	0.912552	0.000169

**Table 6.9:** Top 10 nine-locus SNPs.

Nine-locus	Relative importance	Scaled importance	Percentage
GSTM1_GSTT1_CypIA1m1_CypIAIm2_CypIAIm4_Cyp1B1.48_CypIBI.432_CypIBI.453_COMT.other_1	1	1	0.010758
GSTM1_GSTT1_CypIA1m1_CypIAIm4_Cyp1B1.48_Cyp1B1.119_CypIBI.432_CypIBI.453_COMT.other_0_1	0.996381	0.996381	0.010719
GSTM1_GSTT1_CypIA1m1_CypIAIm4_Cyp1B1.48_Cyp1B1.119_CypIBI.432_CypIBI.453_COMT.other_1_0	0.982216	0.982216	0.010567
GSTM1_GSTT1_CypIA1m1_CypIAIm2_CypIAIm4_Cyp1B1.119_CypIBI.432_CypIBI.453_COMT.other	0.96474	0.96474	0.010379
GSTM1_CypIA1m1_CypIAIm2_CypIAIm4_Cyp1B1.48_Cyp1B1.119_CypIBI.432_CypIBI.453_COMT.2_0_0_0_1_1_1_0_1	0.961017	0.961017	0.010339
GSTM1_GSTT1_CypIAIm2_CypIAIm4_Cyp1B1.48_Cyp1B1.119_CypIBI.432_CypIBI.453_COMT.other_0_1	0.956817	0.956817	0.010293
GSTM1_GSTT1_CypIA1m1_CypIAIm2_CypIAIm4_Cyp1B1.48_CypIBI.432_CypIBI.453_COMT.other_0_1	0.956505	0.956505	0.01029
GSTM1_GSTT1_CypIA1m1_CypIAIm2_CypIAIm4_Cyp1B1.48_Cyp1B1.119_CypIBI.432_COMT.other_2_2_1	0.948387	0.948387	0.010203
GSTM1_GSTT1_CypIA1m1_CypIAIm2_CypIAIm4_Cyp1B1.48_CypIBI.432_CypIBI.453_COMT.other_0_2	0.936297	0.936297	0.010073
GSTM1_GSTT1_CypIA1m1_CypIAIm2_CypIAIm4_Cyp1B1.48_Cyp1B1.119_CypIBI.432_CypIBI.453.other_0	0.936251	0.936251	0.010072

**Table 6.10:** Top 10 ten-locus SNPs.

Ten-locus	Relative importance	Scaled importance	Percentage
CypIA1m1_CypIAIm2_CypIAIm4_Cyp1B1.48_Cyp1B1.119_CypIBI.432_CypIBI.453_COMT_GSTM1_GSTT1.other	1	1	0.10578
CypIA1m1_CypIAIm2_CypIAIm4_Cyp1B1.48_Cyp1B1.119_CypIBI.432_CypIBI.453_COMT_GSTM1_GSTT1.other_0_0	0.956917	0.956917	0.101223
CypIA1m1_CypIAIm2_CypIAIm4_Cyp1B1.48_Cyp1B1.119_CypIBI.432_CypIBI.453_COMT_GSTM1_GSTT1.0_0_0_1_1_1_0_1_2_0	0.915503	0.915503	0.096842
CypIA1m1_CypIAIm2_CypIAIm4_Cyp1B1.48_Cyp1B1.119_CypIBI.432_CypIBI.453_COMT_GSTM1_GSTT1.other_1_0	0.901195	0.901195	0.095328
CypIA1m1_CypIAIm2_CypIAIm4_Cyp1B1.48_Cyp1B1.119_CypIBI.432_CypIBI.453_COMT_GSTM1_GSTT1.0_0_0_0_1_1_1_2_0	0.882414	0.882414	0.093342
CypIA1m1_CypIAIm2_CypIAIm4_Cyp1B1.48_Cyp1B1.119_CypIBI.432_CypIBI.453_COMT_GSTM1_GSTT1.other_2_0	0.851962	0.851962	0.09012
CypIA1m1_CypIAIm2_CypIAIm4_Cyp1B1.48_Cyp1B1.119_CypIBI.432_CypIBI.453_COMT_GSTM1_GSTT1.other_1	0.817296	0.817296	0.086453
CypIA1m1_CypIAIm2_CypIAIm4_Cyp1B1.48_Cyp1B1.119_CypIBI.432_CypIBI.453_COMT_GSTM1_GSTT1.other_2_1	0.8107	0.8107	0.085756
CypIA1m1_CypIAIm2_CypIAIm4_Cyp1B1.48_Cyp1B1.119_CypIBI.432_CypIBI.453_COMT_GSTM1_GSTT1.other_0	0.800813	0.800813	0.08471
CypIA1m1_CypIAIm2_CypIAIm4_Cyp1B1.48_Cyp1B1.119_CypIBI.432_CypIBI.453_COMT_GSTM1_GSTT1.other_0_1_2_0	0.778783	0.778783	0.08238

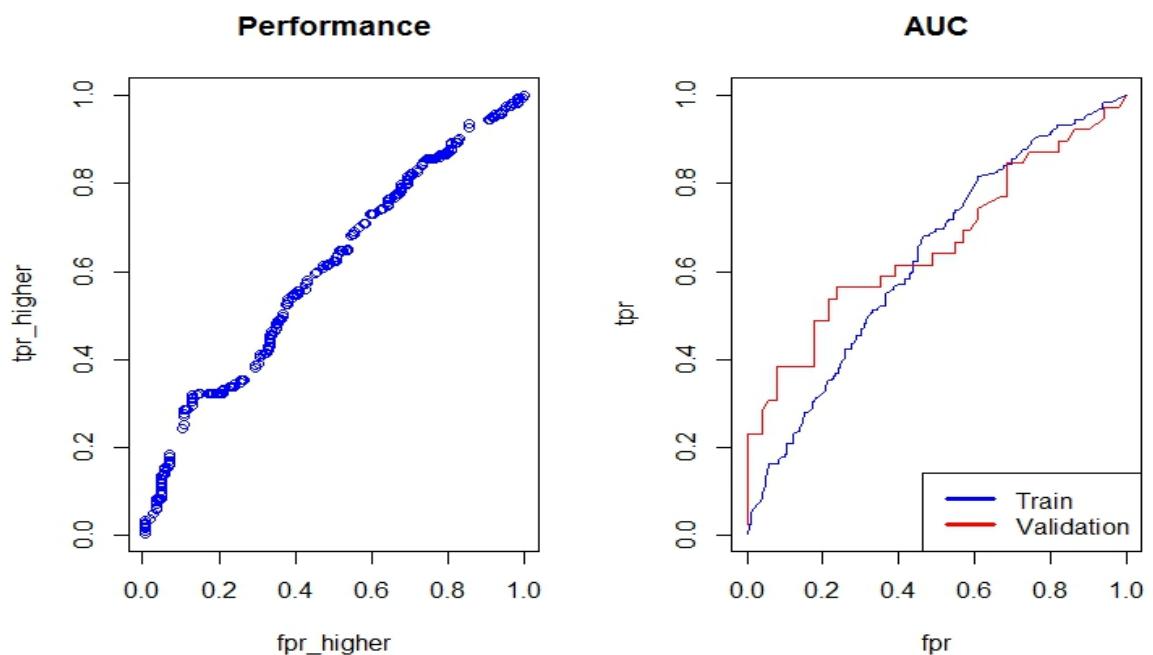


Figure 6.3 ROC plots of higher-order interaction model a) AUC for training data b) AUC for validation data

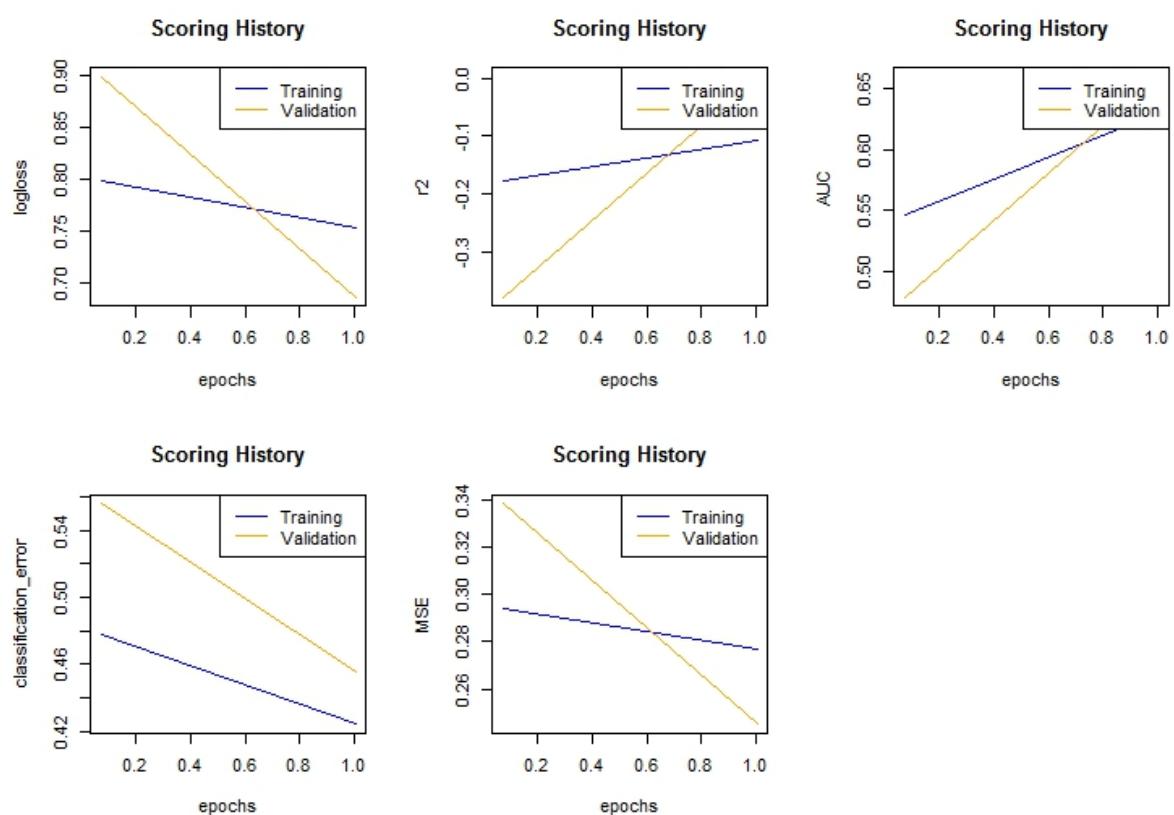


Figure 6.4 Scoring metrics of higher-order interaction model

The method is further analysed by combining single-locus to ten-locus SNPs to observe the performance of the method over the combined effect. Figure 6.3 illustrates the ROC graph of the model for the combined higher-order interactions, during training and validation respectively. Figure 6.4 shows the performance metrics of the model during training and validation.

**Table 6.11:** Top 10 higher-order SNPs.

SNP Interactions	Importance
CyplAIm2_Cyp1B1.119.0_0	1
GSTM1_GSTT1_CyplA1m1_CyplAIm4_Cyp1B1.48_Cyp1B1.119_CypIBI.432_CypIBI.453 _COMT2.other	0.9608197
Cyp1B1.119.0	0.9038942
GSTM1_GSTT1_CyplA1m1_CyplAIm4_Cyp1B1.48_Cyp1B1.119_CypIBI.432_CypIBI.453 _COMT3.other	0.8887891
GSTM1_GSTT1_CyplA1m1_CyplAIm2_CyplAIm4_Cyp1B1.48_CypIBI.432_CypIBI.453 _COMT0.other_2_0_1	0.8884753
GSTM1_GSTT1_CyplA1m1_CyplAIm4_Cyp1B1.48_Cyp1B1.119_CypIBI.432_CypIBI.453 _COMT1.other	0.8826617
GSTM1_GSTT1_CyplA1m1_CyplAIm2_CyplAIm4_Cyp1B1.48_Cyp1B1.119_CypIBI.432 _CypIBI.4532.other_0	0.8727763
Cyp1B1.119_GSTM1.0_2	0.8687097
GSTM1_GSTT1_CyplA1m1_CyplAIm2_CyplAIm4_Cyp1B1.48_Cyp1B1.119_CypIBI.432 _CypIBI.4533.2_0_0_0_0_0_1_0	0.8671682
CyplAIm2_Cyp1B1.48_Cyp1B1.119_CypIBI.432_CypIBI.4530.0_1_1_2_1	0.8668877
CyplA1m1_GSTT1.1_0	0.866191
GSTM1_GSTT1_CyplA1m1_CyplAIm2_CyplAIm4_Cyp1B1.48_Cyp1B1.119_CypIBI.432 _COMT4.other_1_1	0.8606181
GSTM1_GSTT1_CyplA1m1_CyplAIm2_CyplAIm4_Cyp1B1.119_CypIBI.432_CypIBI.453 _COMT.other	0.859943
Cyp1B1.119_GSTT1.1_0	0.8564746
GSTM1_GSTT1_CyplAIm2_CyplAIm4_Cyp1B1.48_Cyp1B1.119_CypIBI.432_CypIBI.453 _COMT2.other_0_1	0.8562852
CyplA1m1_GSTM1.1_1	0.8538693
CyplA1m1_CyplAIm2.0_0	0.8512251
Cyp1B1.48_GSTM1.1_2	0.85115
CyplAIm4_CypIBI.432.2_1	0.8479716
CyplAIm4_Cyp1B1.48_Cyp1B1.119_CypIBI.453_COMT0.0_1_2_0_1	0.837077

Top 20 highly ranked combined higher-order SNP interactions are shown in Figure 6.5 and Table 6.11. It is observed from the results that the two-locus SNP interaction (CyplAIm2\_Cyp1B1.119) is being highly associated with the sporadic breast cancer. These experimental results showed that the extended deep learning method can be easily extended to higher-order interactions.

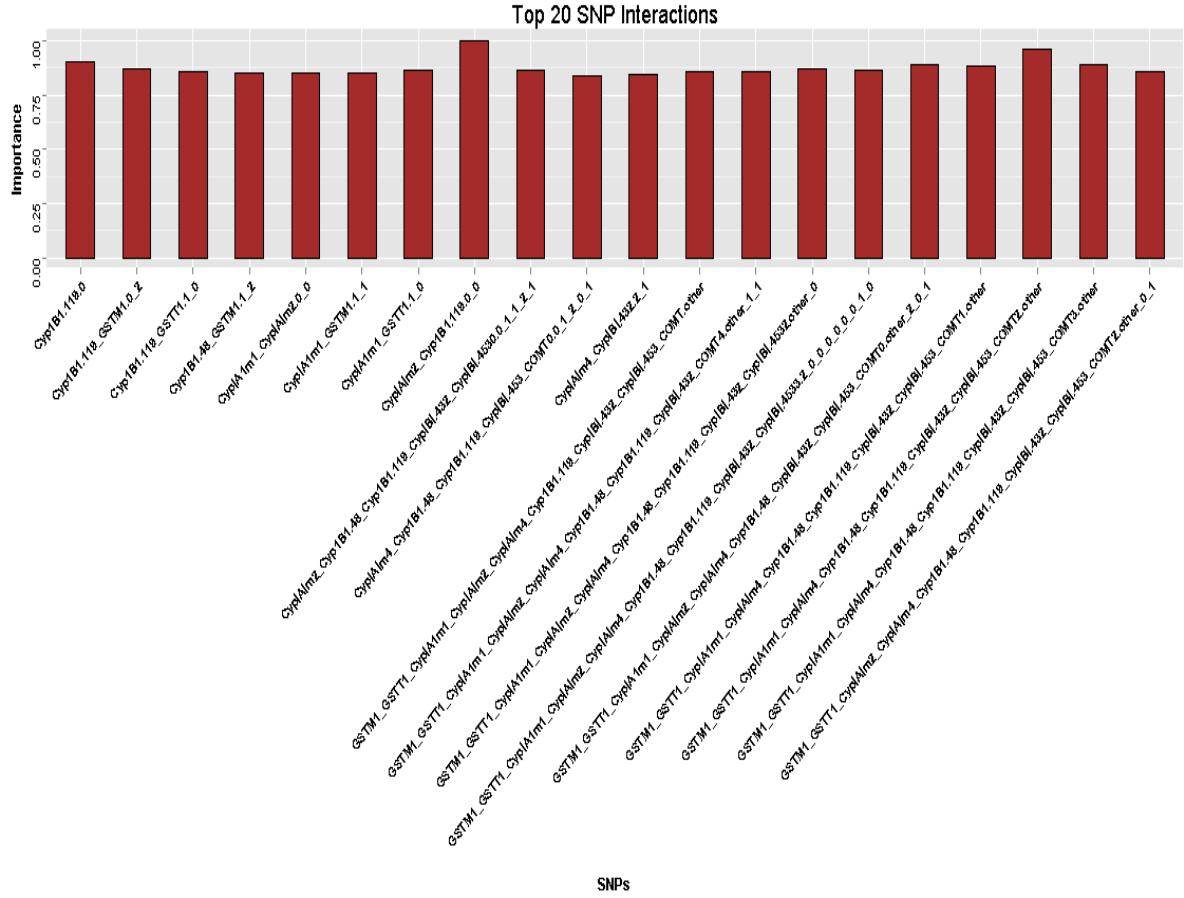


Figure.6.5 Top 20 higher-order SNP-interactions for sporadic breast cancer data

### 6.3 Evaluations for unsupervised learning

The extended method is further evaluated for unsupervised learning tasks such as dimensionality reduction and feature learning using [311] over sporadic breast cancer data. High-dimensional data (888-dimensional data) is reduced to low dimensional data (50-dimensional data) using PCA. PCA computes principal components of breast cancer data and their deviations are computed. PCA computes the singular value decomposition of the gram matrix by using power method [338]. Figure 6.6 and Figure 6.7 represents the eigenvectors of PCA over single-locus and two-locus SNPs respectively (illustrated based on [338]). Figure 6.8 shows the standard deviations and cumulative variances of first 50 principal components. It is observed that the first 10 or 20 components covered the majority of variance in the breast cancer dataset.

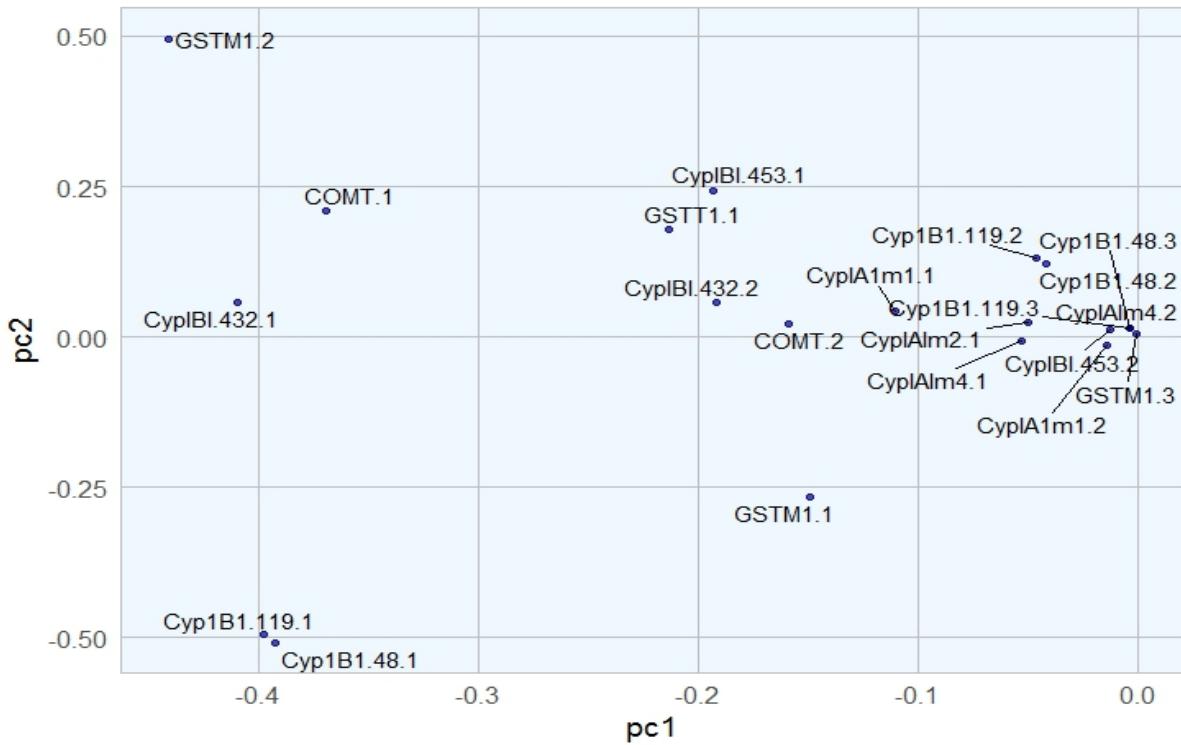


Figure.6.6 PCA on single-locus SNP

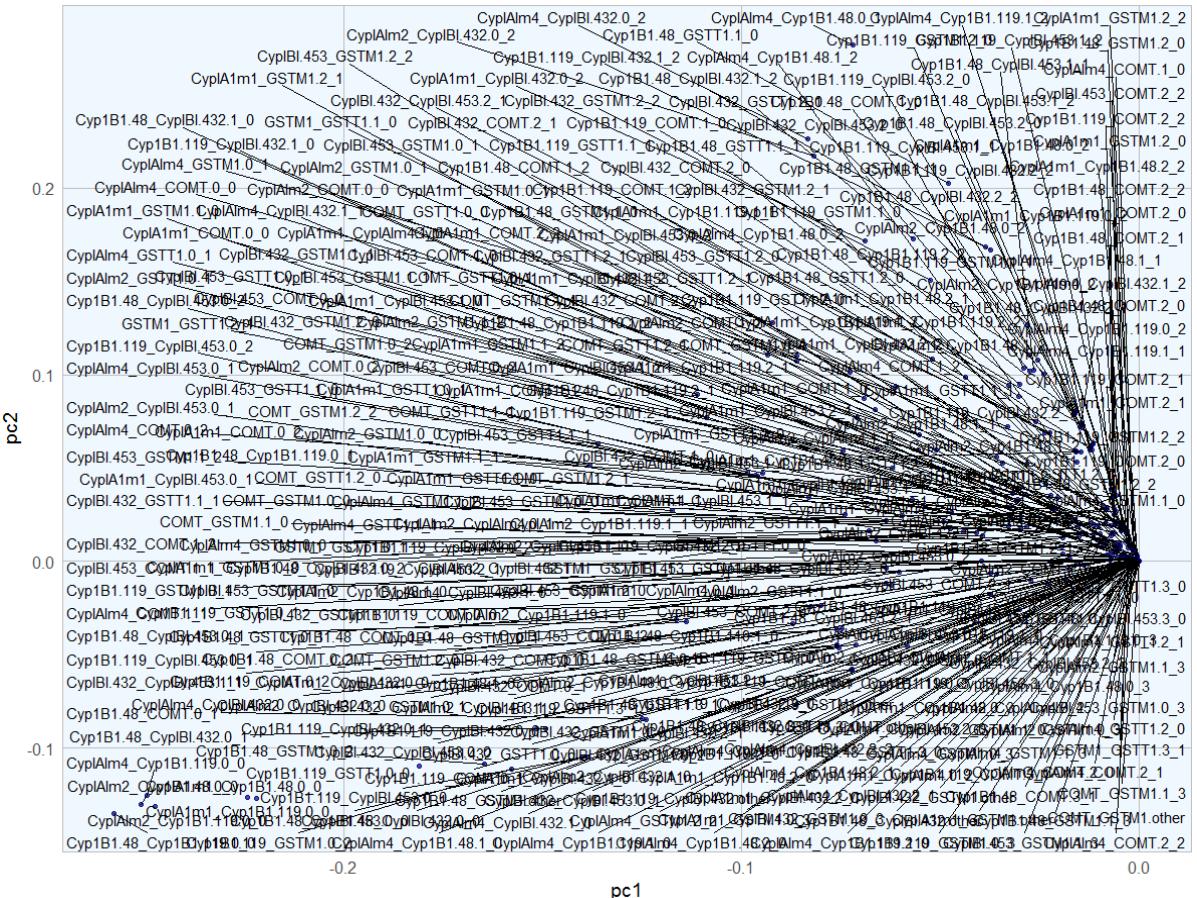


Figure.6.7 PCA on two-locus SNP interactions

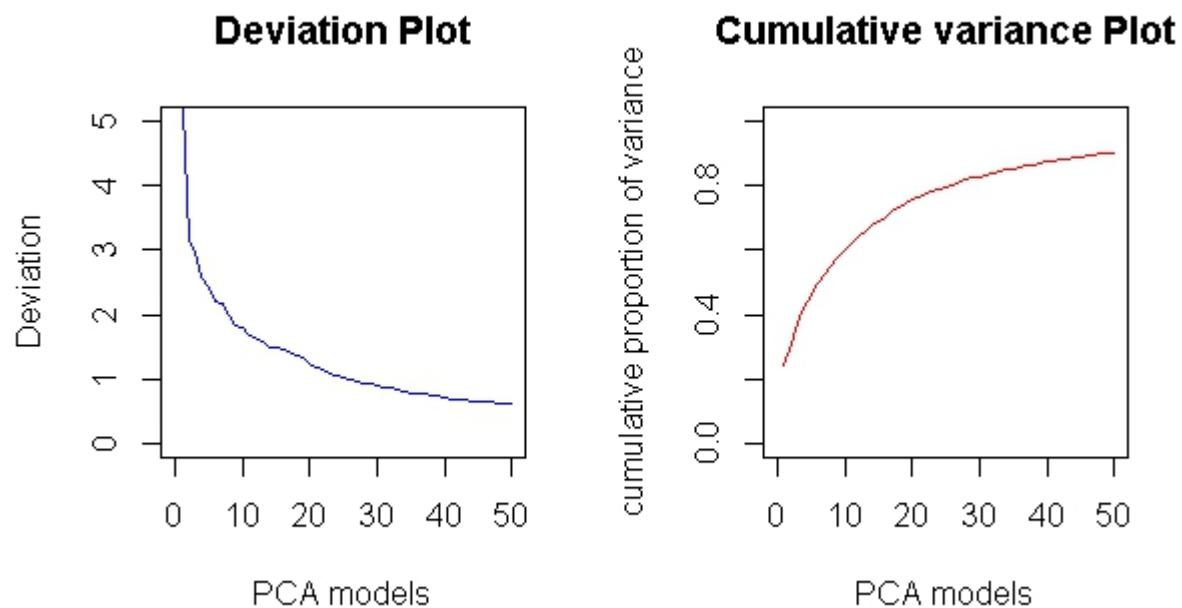


Figure 6.8      a) Deviation plot and b) Cumulative variance plot of PCA for higher-order interactions.

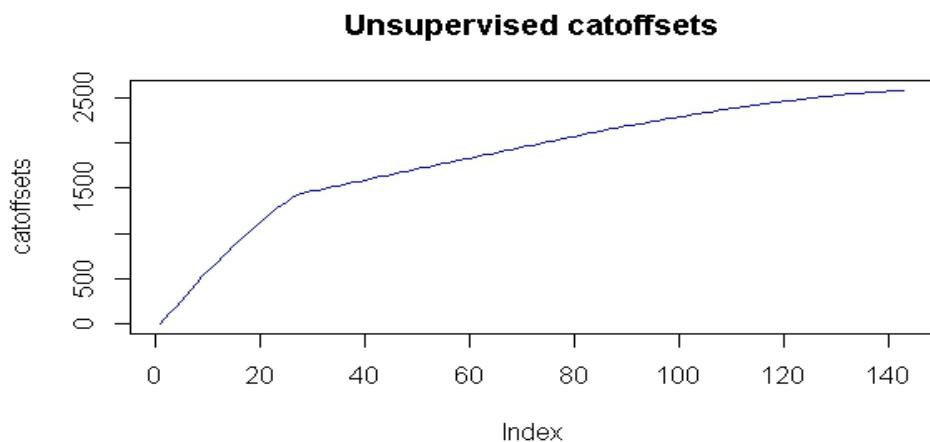


Figure 6.9      Cutoff offset plot for unsupervised higher-order model learning

Deep autoencoder is used for unsupervised feature learning by discovering the anomalies in the reduced representation of the original data [311]. Categorical offsets for unsupervised learning due to combined effect of higher-order interactions are plotted in Figure 6.9. The reconstruction error between output layer and input layer are plotted in Figure 6.10 to find the outliers. The highest error occurs when the test points do not match with the learned pattern.

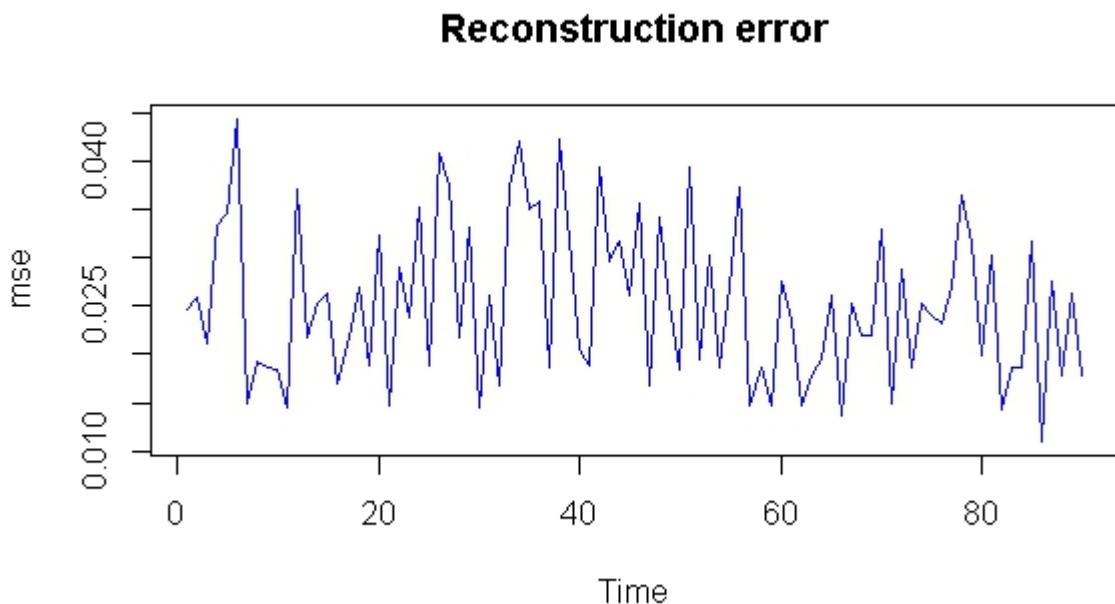


Figure 6.10 Reconstruction error plot for unsupervised higher-order model learning

## 6.4 Evaluations for optimising hyper-parameters

The objective of this study is to tune the hyper-parameters for improving the predictive performance of the deep learning algorithm. Grid search and random grid search are performed by optimizing the hyper-parameters that can have impact on the model accuracy. Several experiments were conducted over hypertension data by evaluating the model metrics of each model for both the grid and random grid search methods using [326]. In grid search, all possible combinations of hyper-parameters (such as, hidden layers, epochs, activation function, input drop ratio, epsilon, momentum, learning rate, annealing rate, L1 and L2 penalties') are tested with Cartesian grid or exhaustive search. The search is performed by specifying parameter values that would be common to all the models, and a map that specifies the parameter spaces to be travelled. In random grid search, hyper-parameters and search criteria is defined to tune the models. Hyper-parameter values are chosen randomly within the specified values without any repeats by building the models sequentially. Optimal combination of parameter values is identified to maximize the model accuracy. Results of each model

are tested in the grid, and the best model with the highest accuracy or the predictive performance is selected.

Figure 6.11 and Figure 6.12 illustrates the performance metrics of each model in terms of accuracy, auc, error, logloss, mean square error, precision, recall and specificity, using grid search, and random grid search respectively over breast cancer data. The performance of the model is evaluated for the prediction accuracy on the test data. The auc of the best model identified from the grid search is 0.8242673 along with mse 0.1949181 on breast cancer data. The auc and mse of the best model identified for hypertension data are 0.6751302 and 0.266421 respectively. The search fairly performed well in large hyper-parameter space. However, it was sensitive for few parameter combinations (peaked error functions). Figure 6.13 represents the prediction of breast cancer on the test data. Similarly, Figure 6.14 and Figure 6.15 illustrates the performance metrics of extended DNN over hypertension data using grid and random search, with respect to accuracy, auc, error, logloss, mean square error, precision, recall and specificity. Prediction of hypertension on the test data for the best model is shown in Figure 6.16.

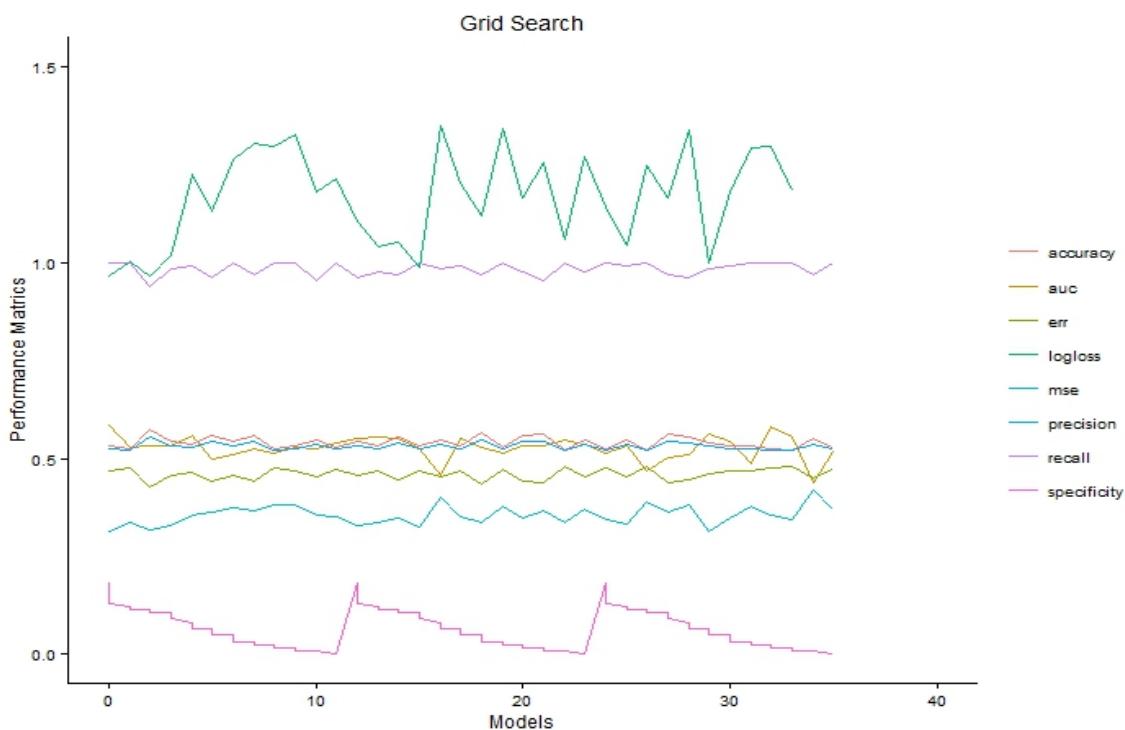


Figure.6.11 Model metrics (accuracy, auc, err, logloss, mse, precision, recall, and specificity) of DNN using Grid Search on sporadic breast cancer data.

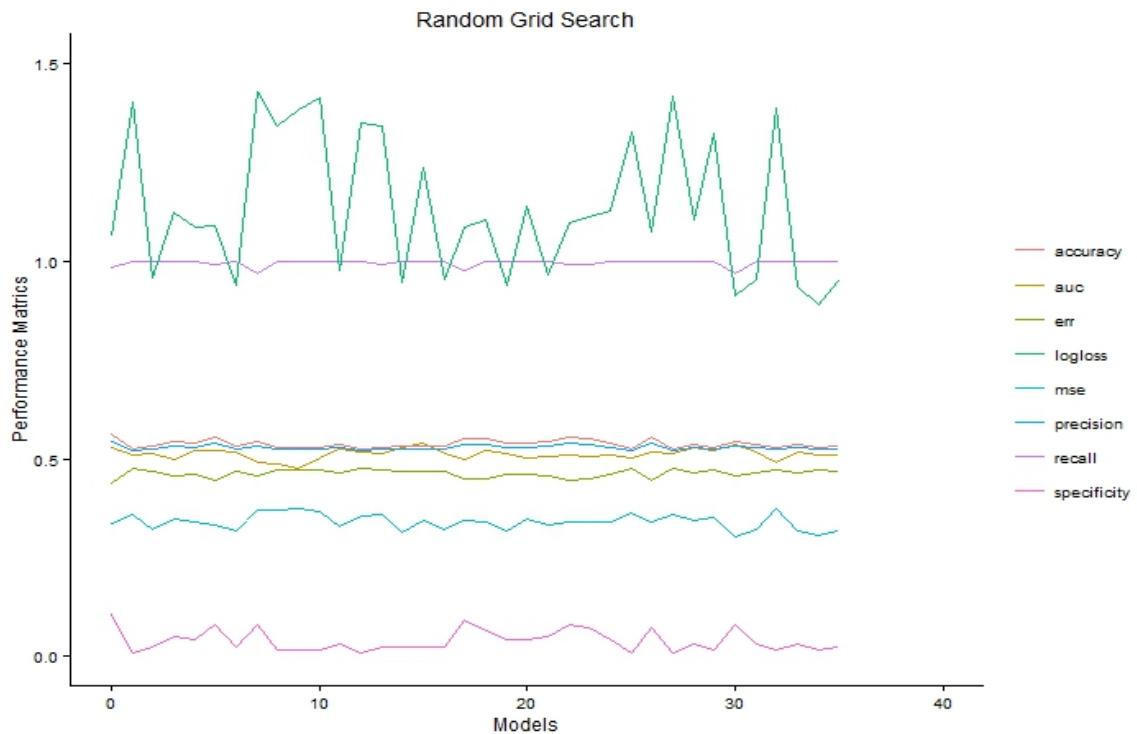


Figure.6.12 Model metrics (accuracy, auc, err, logloss, mse, precision, recall, and specificity) of DNN using Random Grid Search on sporadic breast cancer data.

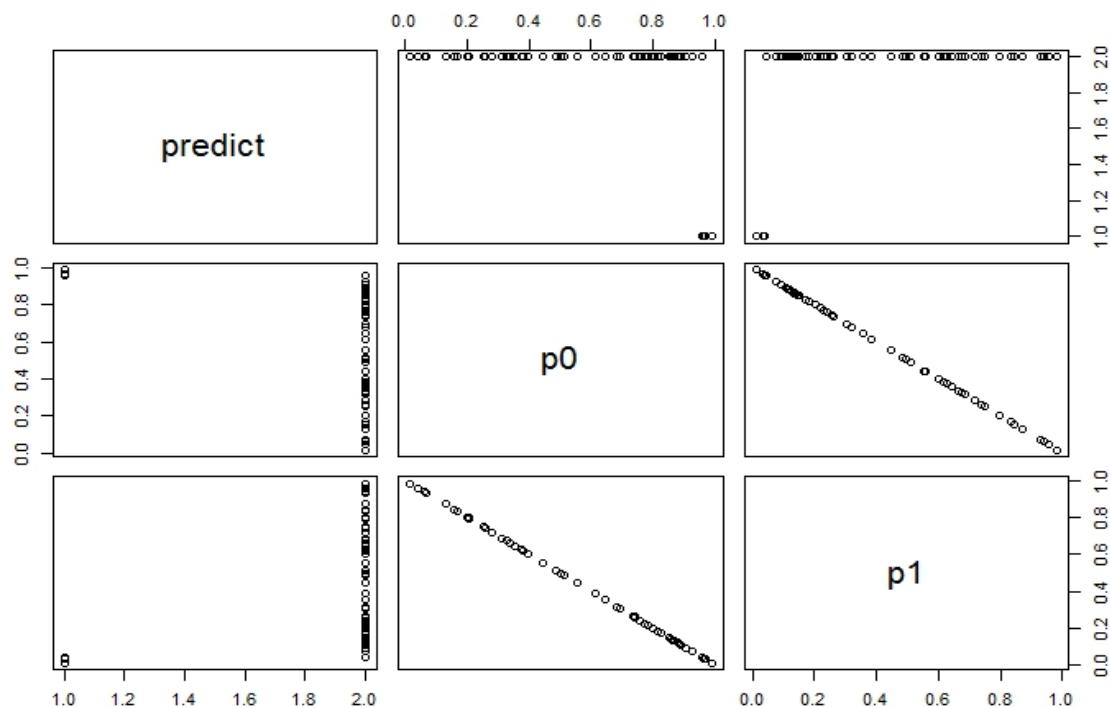


Figure.6.13 Prediction of the test data on the best model for sporadic breast cancer data.

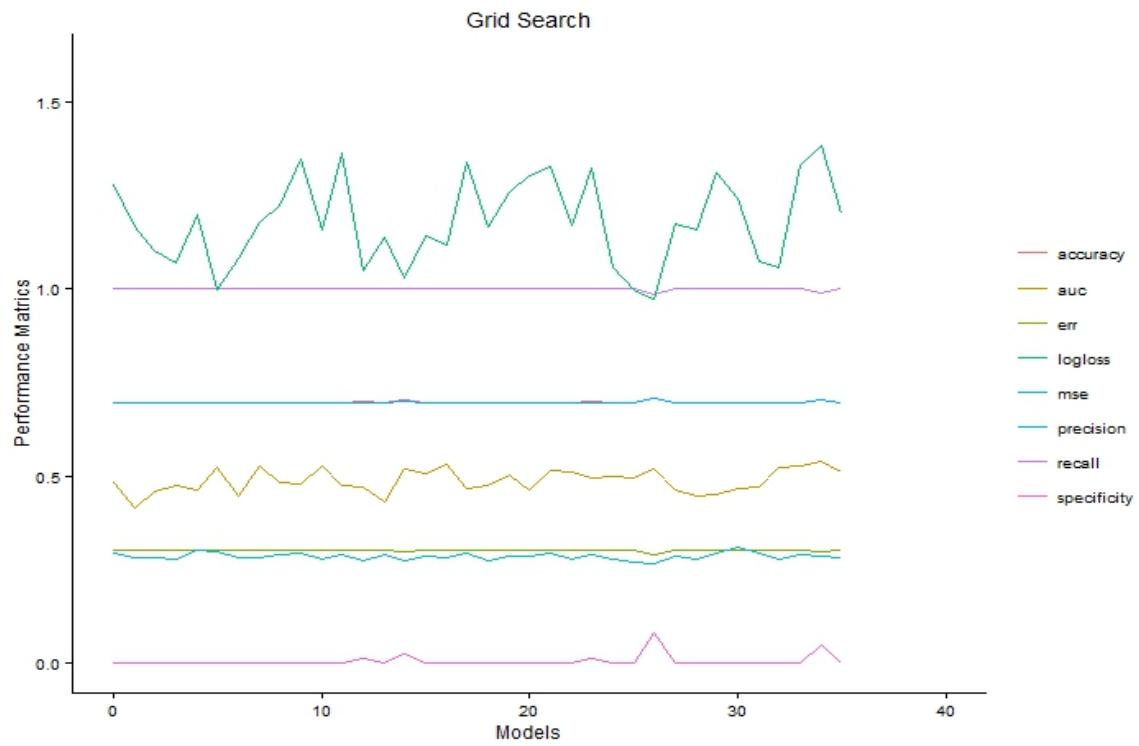


Figure.6.14 Model metrics (accuracy, auc, err, logloss, mse, precision, recall, and specificity) of DNN using Grid Search on hypertension data.

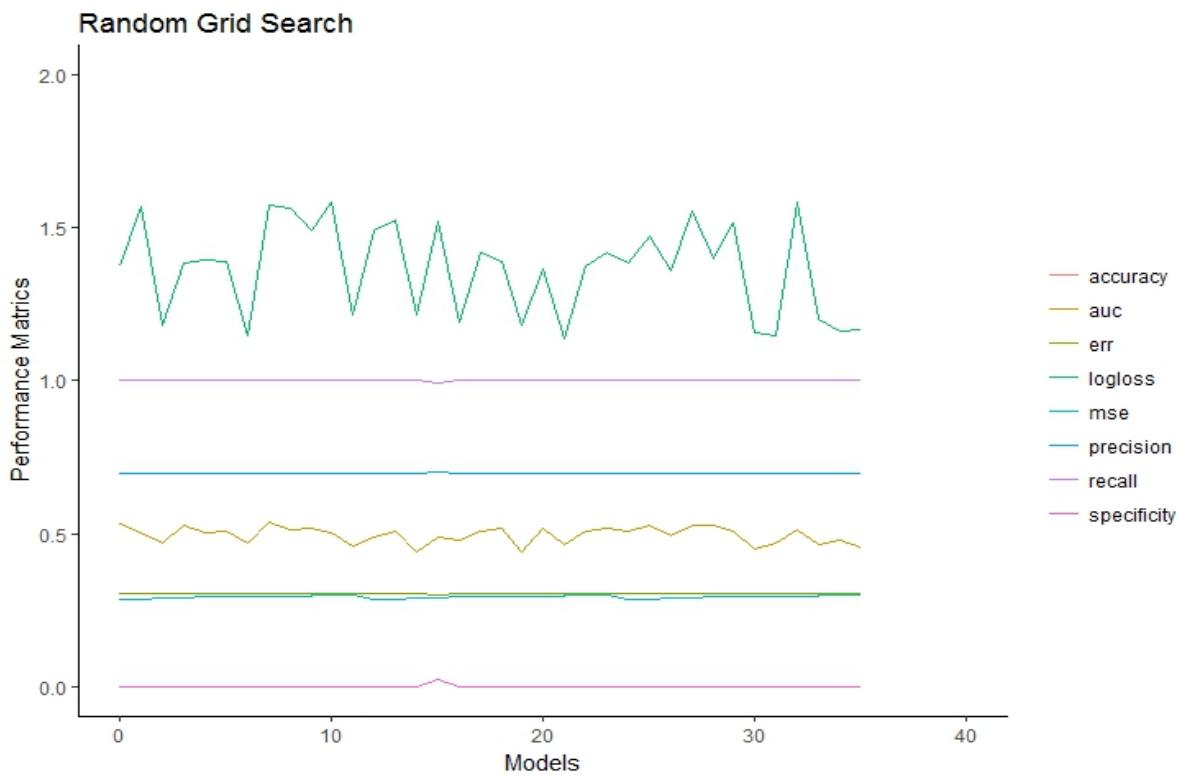


Figure.6.15 Model metrics (accuracy, auc, err, logloss, mse, precision, recall, and specificity) of DNN using Random Grid Search on hypertension data.

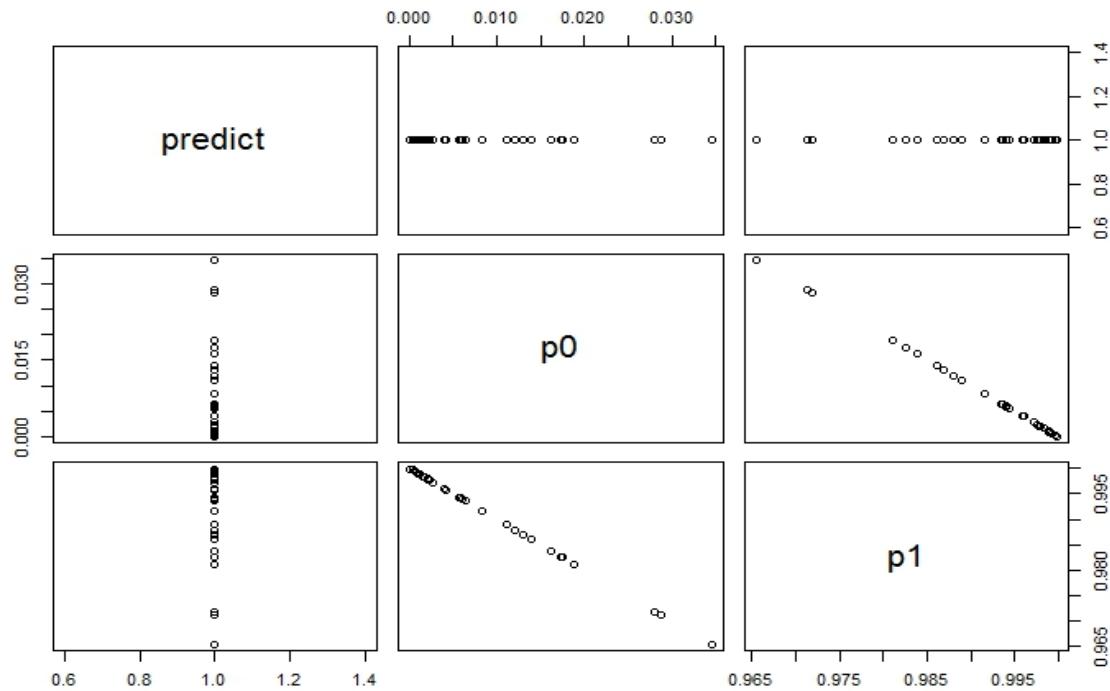


Figure.6.16 Prediction of the test data on the best model for hypertension data.

In random search, the hyper-parameters are chosen randomly rather than a best guess. Mean square error (mse) and auc of each model during training, validation, and testing on breast cancer data is illustrated in Figure 6.17 and Figure 6.18. Auc and mse of the best model identified by random grid search are 0.8835072, and 0.1436914 respectively. The performances of all the models on hypertension data using random grid search is represented in Figure 6.19 and Figure 6.20 during training, validation, and testing. The corresponding auc and mse of the best model are 0.7014199, and 0.2801003. Internal 10-fold Cross validation is also performed on the entire data to confirm these parameters. It is observed that the best models identified by the random search have better validation aucs than the previous models identified by grid search. It is also observed that the variance between validation and testing is less than the variance observed in grid search. Random search often performed well for more than four parameters by identifying best model in less time than performing exhaustive grid search. These findings confirmed that the random search worked well to find the models with the highest prediction accuracy and lowest mse for both real datasets. The optimal hyper-parameters of the best model predicted by random search are used to detect SNP interactions.

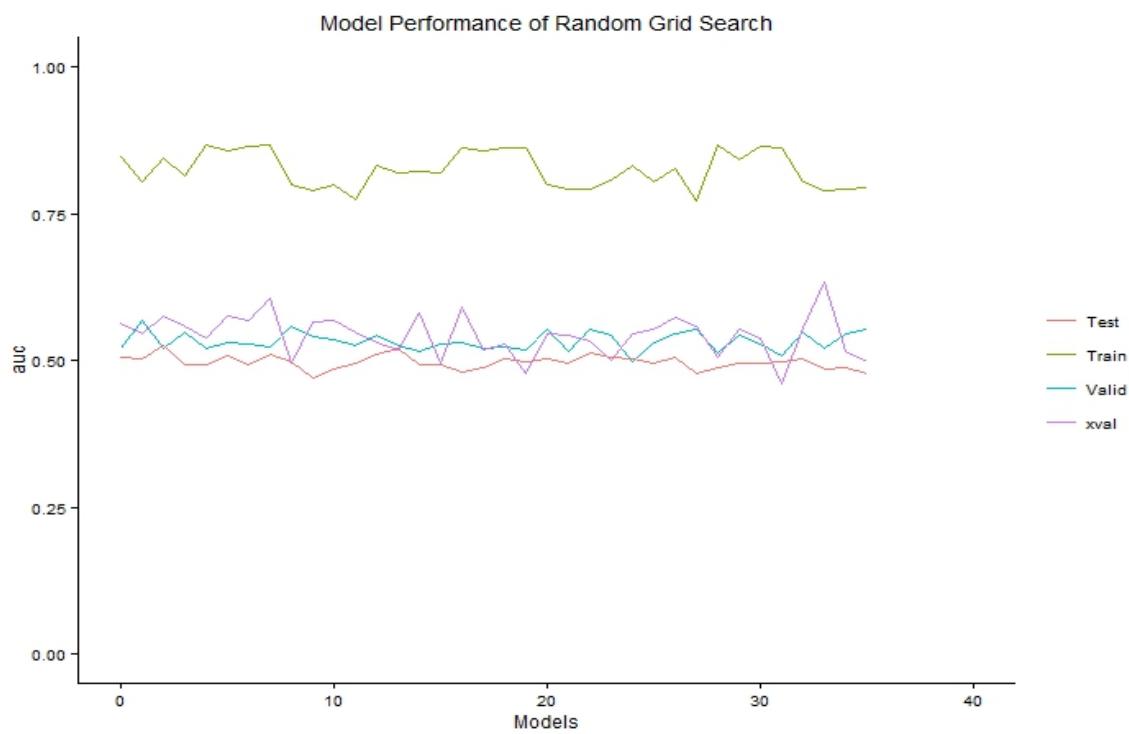


Figure.6.17 Model performance on breast cancer data (auc vs models)



Figure.6.18 Model performance on breast cancer data (mse vs models).

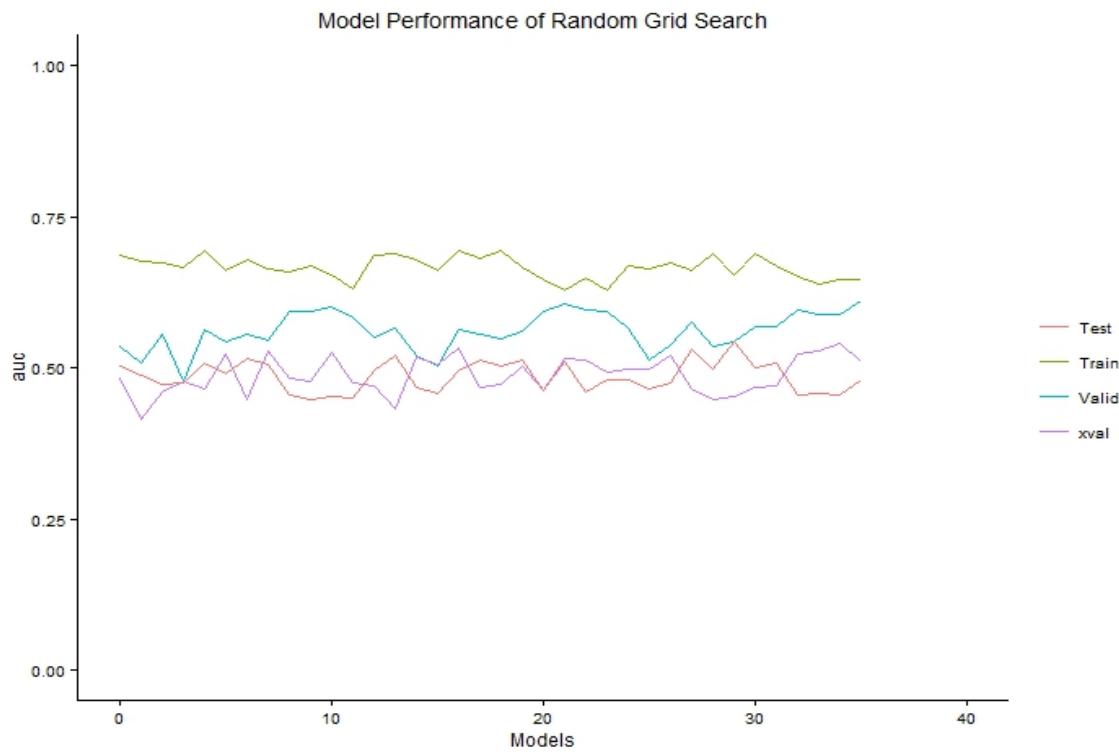


Figure.6.19 Model performance on hypertension data (auc vs models)

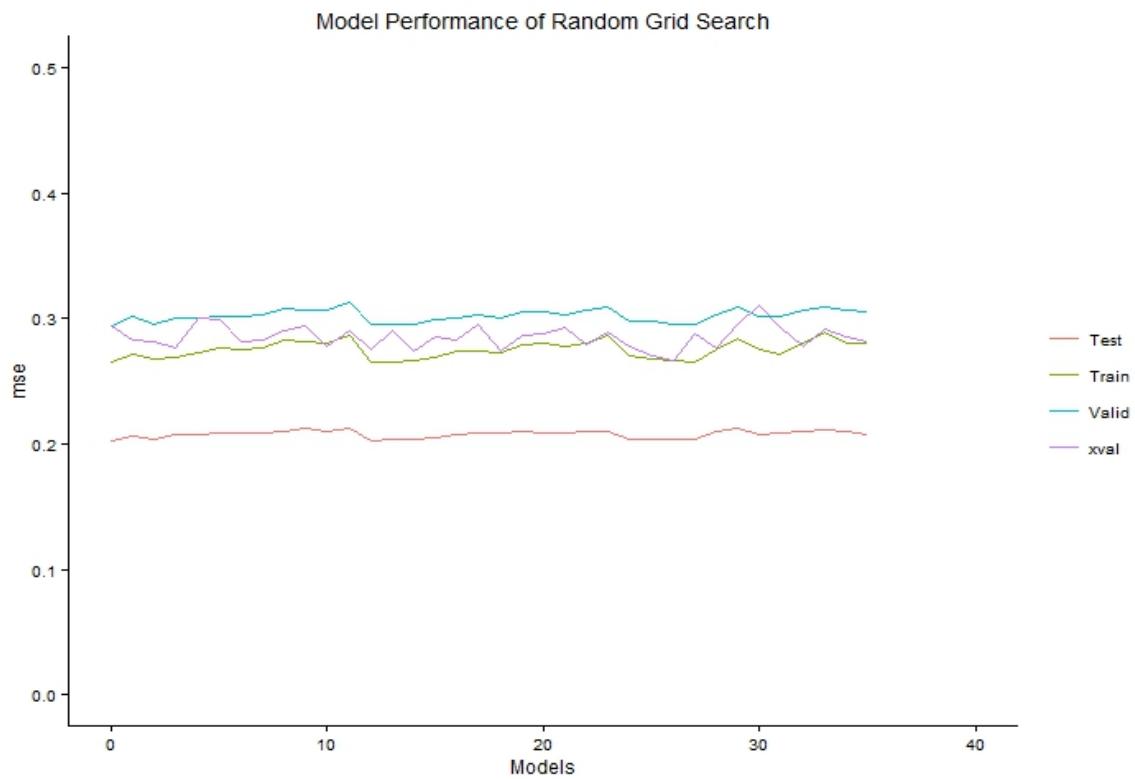


Figure.6.20 Model performance on hypertension data (mse vs models).

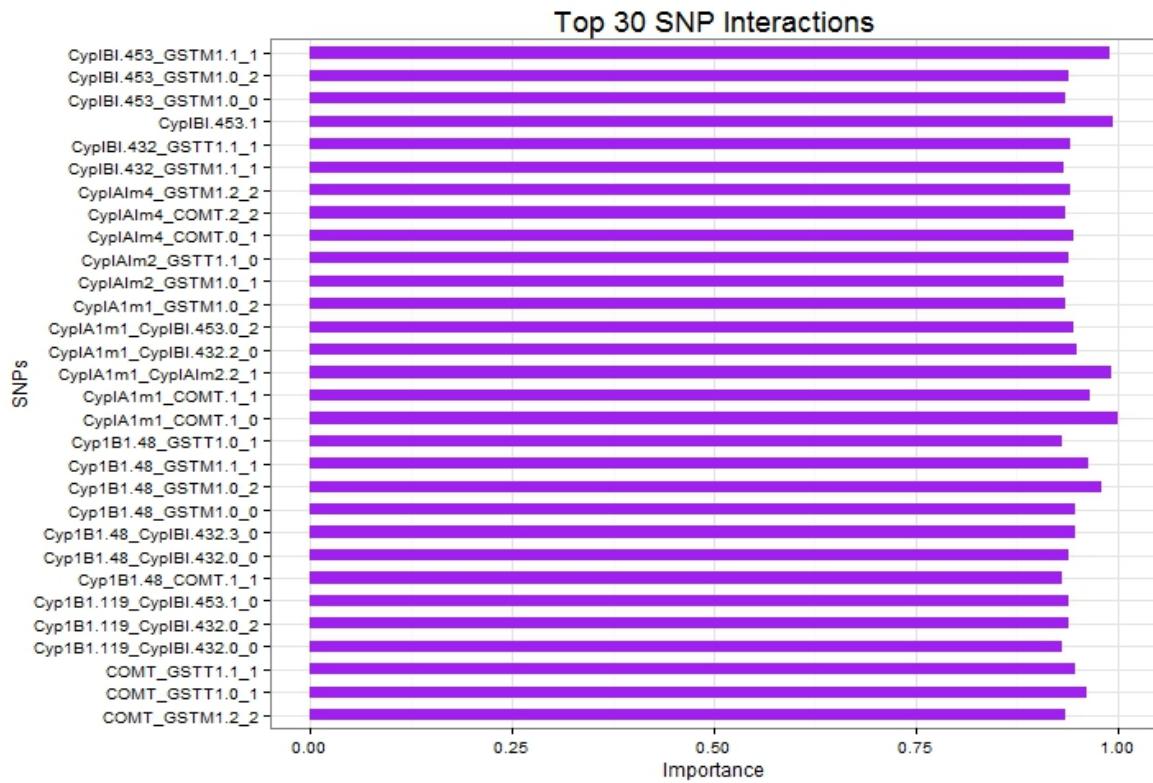


Figure.6.21 Top 30 SNP interactions of sporadic breast cancer data.

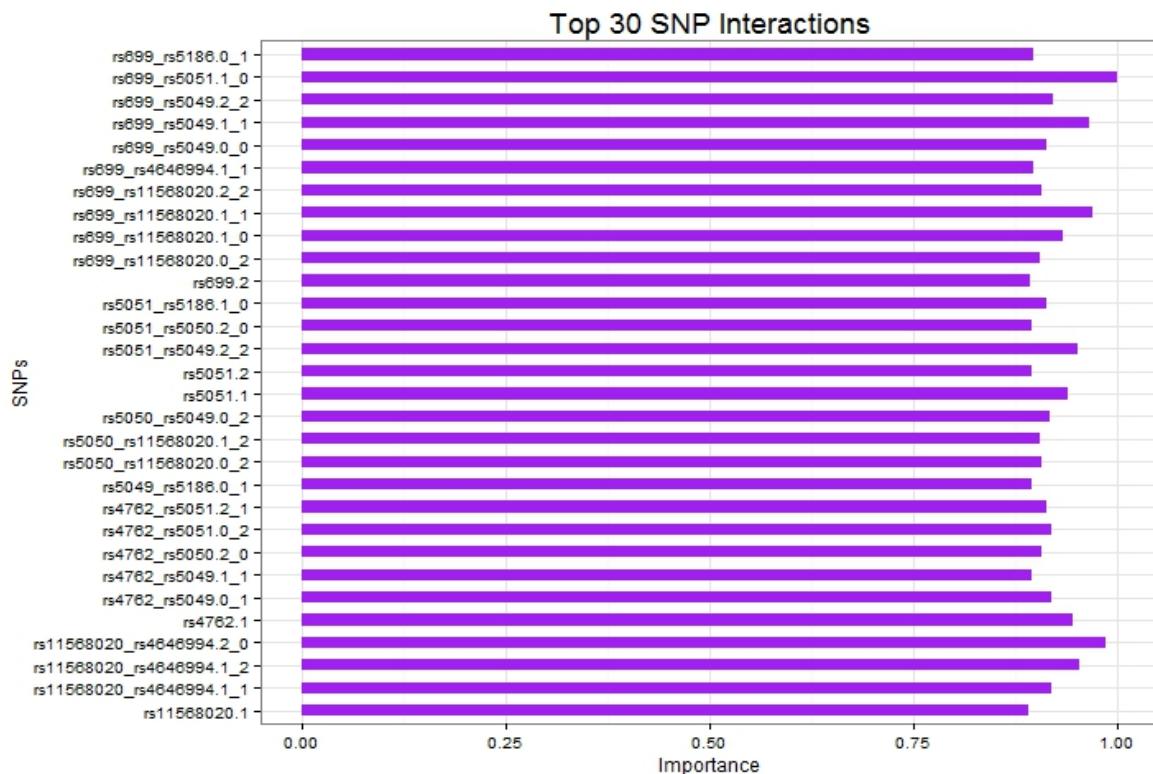


Figure.6.22 Top 30 SNP interactions of hypertension data.

Top 30 SNP interactions responsible for breast cancer and hypertension are plotted in Figure 6.21 and Figure 6.22. The best two-way SNP interaction responsible for sporadic breast cancer identified by the tuned model is CypIA1m1 (presence of AA) and COMT (presence of aa). The interaction between rs699 (presence of AA) in AGT and rs5051 (presence of aa) in AGT 5' could be the most predominant cause of hypertension. It is important to reduce the number of wrong predictions of cases or controls in reality. Hence, the extended DNN is trained efficiently such that the model does not miss out any important interacting SNPs.

## 6.5 Evaluations by improving learning

Several experimental results are demonstrated on the extended DNN method by improving the way networks learn. The improved DNN method is implemented and analysed in R [326]. The main goal of this study is to improve the identification of higher-order SNP interactions responsible for chronic kidney disease. In the preliminary evaluations, the improved DNN is evaluated by changing the activation functions (such as, rectifier, tanh, softmax, and maxout) and optimising hyperparameters (using grid and random grid search). The performance of the models is evaluated for various non-linear activation functions with and without dropouts. It is observed that *tanh* has highest prediction accuracy with low classification error. Further, a number of experiments are performed for evaluating all the possible combinations of hyperparameters (such as, hidden layers, epochs, input drop ratio, epsilon, momentum, learning rate, annealing rate, L1 and L2 penalties'). The best model with optimal hyperparameters (hidden = c(200,200,200), activation = "Tanh", epochs=1000, rate=0.01, initial\_weight\_distribution="Normal", initial\_weight\_scale=0.184837, loss="CrossEntropy", distribution="bernoulli", nesterov\_accelerated\_gradient=T, adaptive\_rate=T, rho = 0.9, epsilon = 1e-06, momentum\_start=0.5, momentum\_ramp=1e5, momentum\_stable=0.9, l2=0.0001, rate\_annealing = 1e-06) is chosen to maximise the model's accuracy. It is observed that the accuracy improved when initial weight are assigned with standard deviation  $\frac{1}{\sqrt{s_i}}$ .

The models performed well for learning rate 0.01. It is also observed that the model convergence and synchronization is decreased as the sample size increases. Computational performance is affected if training samples are too low.

Figure 6.23 illustrates the performance metrics of each model with respect to tpr vs fpr, classification error, logloss, rmse, auc, accuracy, and precision for both training and validation, and testing performance. The two-locus interaction between SNP 3 (1) – SNP 30 (2) in mitochondrial D-loop is identified by the best model with the maximum accuracy of 80.49%.

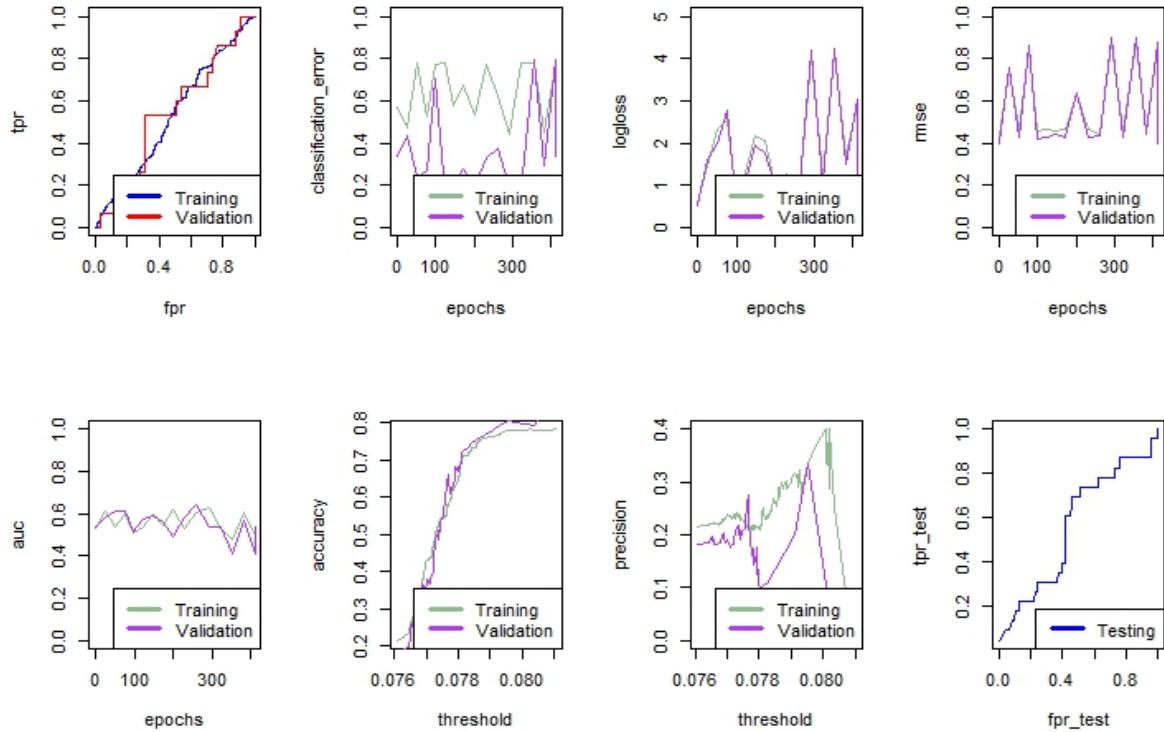


Figure 6.23 Performance analysis of improved DNN for two-locus SNP interactions. (a) True positive rate (tpr) vs false positive rate (fpr) for both training and validation, (b) scoring history of classification error at epochs, (c) scoring history of logloss at epochs, (d) scoring history of rmse at epochs, (e) scoring history of area under curve (auc) for training and validation in epochs, (f) accuracy plot at which maximum accuracy is obtained for a threshold value, (g) Precision plot at which maximum accuracy is obtained for a threshold value, and (h) Testing performance: ©2018 IEEE.

Figure 6.24 illustrates the sensitivity analysis for two-locus SNP interactions using lek profile method [339]. The function groups two-locus interactions by groupings that define the statistical property of the case-control data. Group labels define the colour of the corresponding group. The results of this analysis show the changes in the output with respect to the inputs by providing the information of sensitive SNPs, such that they could be measured more accurately. The observations urged caution in interpretation of the results.

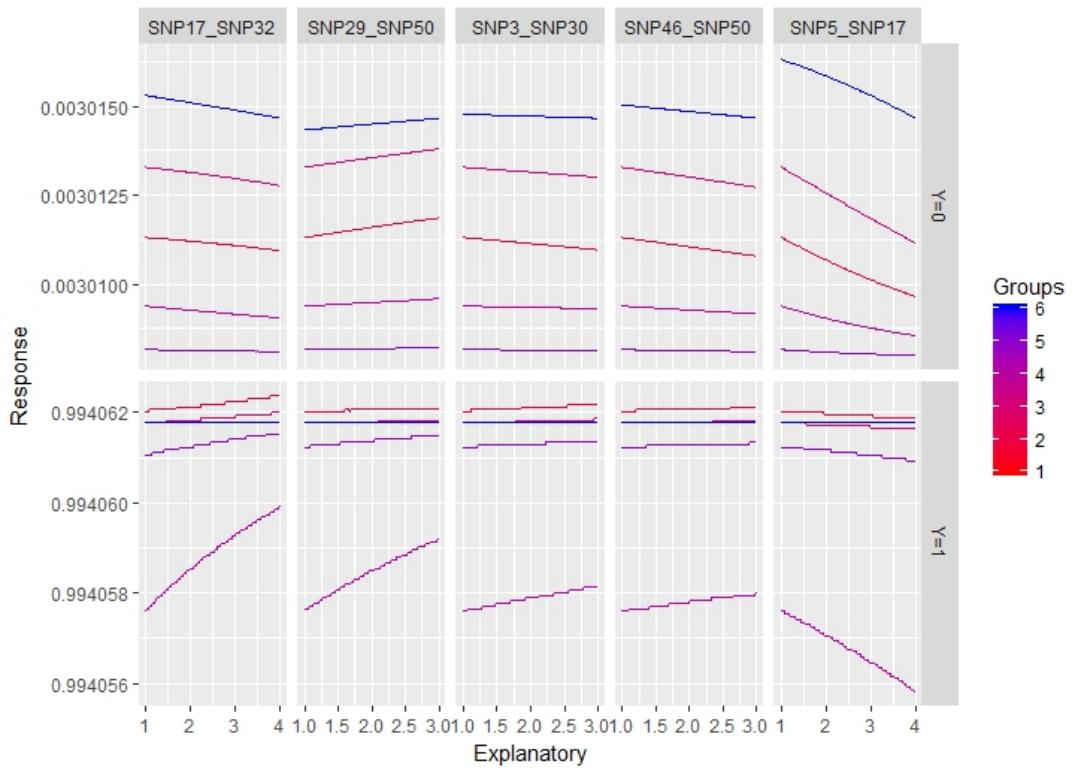


Figure.6.24 Sensitivity analysis of two-locus SNP interactions: ©2018 IEEE.

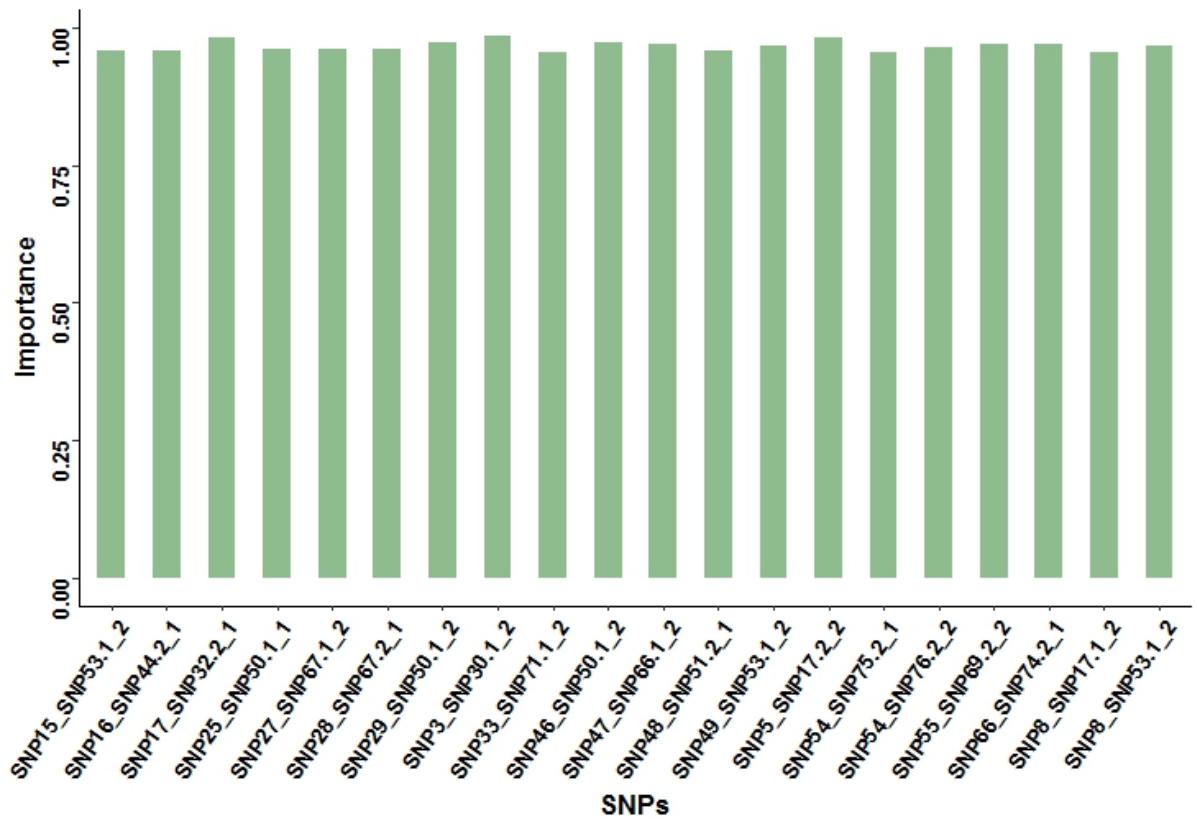


Figure.6.25 Top 20 two-locus SNP interactions: ©2018 IEEE.

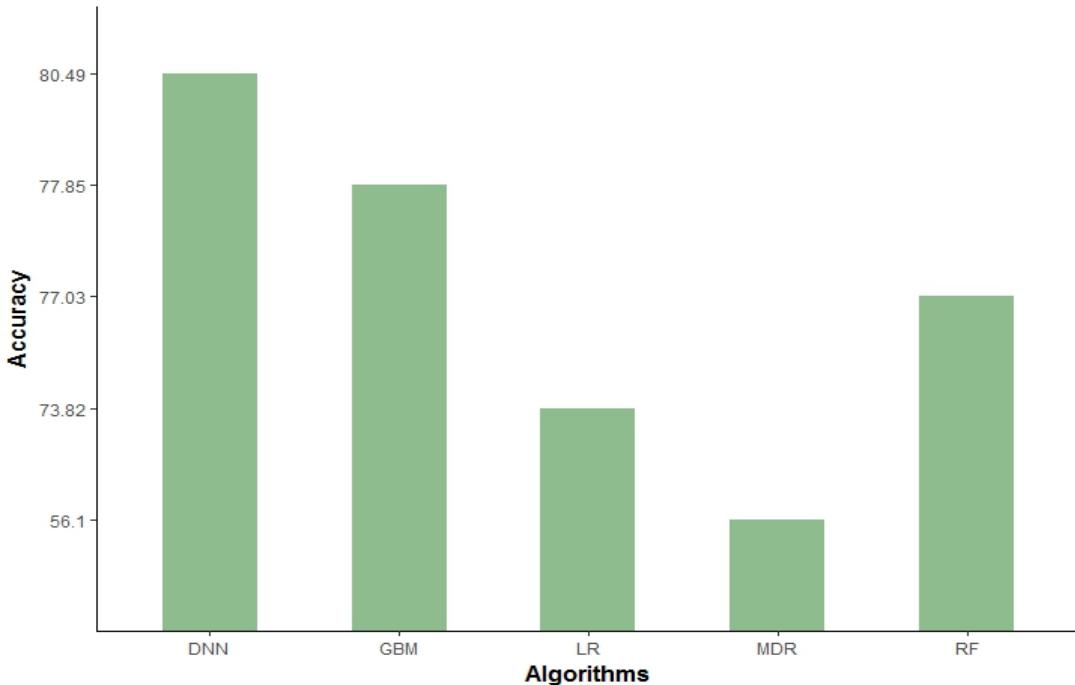


Figure 6.26 Accuracy of DNN compared previous approaches: ©2018 IEEE.

Figure 6.25 plots top 20 two-locus SNP interactions responsible for chronic kidney disease. Figure 6.26 plots the accuracy graph of improved DNN along with the previous methods from the literature. The improved method is compared with some of the commonly used approaches in machine learning (such as, LR, MDR, RF, and GBM).

## 6.6 Chapter summary

In this chapter, previously trained deep learning network is extended by implementing dimensionality reduction, and studied the behaviour of the network for detecting higher-order SNP interactions. It implemented Principal Component Analysis (PCA) to reduce the high-dimensionality multi-factor combination data to the low dimensionality data in the pre-processing step. The method was also studied for unsupervised feature learning tasks. Deep autoencoder was used for unsupervised feature learning by discovering the anomalies in the reduced representation of the original data. The studies were extended to maximize the predictive performance of the models by optimising the hyper-parameters using grid and random grid search approaches. It was observed that the random search performed more efficiently compared to the grid search. The optimal hyper-parameters of the best models were used to obtain the highest prediction accuracy with low classification error.

Further, the method was extended by improving the network learning. It was observed that the predictive performance of the models was maximized by reducing overfitting. Sensitivity analysis was also performed using Lekprofile method for interaction variables with respect to the response variable. However, the power of the extended method dropped when the noise due to GE, MS, GH, and PC are introduced. Hence, in the next chapter, a deep hybrid method is proposed to detect multi-locus SNP interactions in the presence of noise.

# Chapter 7

## A Deep Hybrid Method for the noise data

The proposed deep learning method was extended in the previous chapter by improving the network learning, and optimising hyper-parameters that can have impact on the prediction accuracy. The method was validated under real world data applications, and the performances of the models were observed. The results showed remarkable improvements in predicting higher-order SNP interactions over some of the existing methods. However, the performance of the models was poor in the presence of noise and their combined effects. From the literature, it is observed that RF is robust to noise. The performance of RF is encouraging in the presence of noise compared to other traditional machine learning approaches in GWAS. This motivated the research to be progressed by combining DNN and RF in this chapter. Hence, in this chapter, the research is further extended by proposing hybrid architecture (random forest is integrated into the deep learning method) to maximise the predictive accuracy in the presence of noise and their combined effects over the method.

This chapter is structured as follows: Section 7.1 introduces the proposed hybrid model to detect higher-order interactions in the presence of noise. Section 7.2 presents the variable importance measure applied in this method. Measuring SNP interactions are presented in Section 7.3. Evaluation methods are briefly discussed in Section 7.5. Simulated studies and real data application studies are performed in Section 7.6. Further studies are performed in the presence of MS, GE, GH, and PC, and compared with the previously proposed extended deep learning method. Finally, Section 7.7 includes the discussion.

---

This chapter is reported in the following publication:

- S. Uppu and A. Krishna, "A deep hybrid model to detect multi-locus interacting SNPs in the presence of noise," in *International Journal of Medical Informatics*, vol. 119, pp. 134-151, 2018: ©2018 Elsevier, "The original publication is available at <https://www.sciencedirect.com/science/article/pii/S1386505618303526?via%3Dihub>".

## 7.1 Deep Hybrid Method

The main goal of the proposed method is to improve the identification of SNP interactions in high-dimensional data by enriching the deep data representation learning with the capability of random forest.

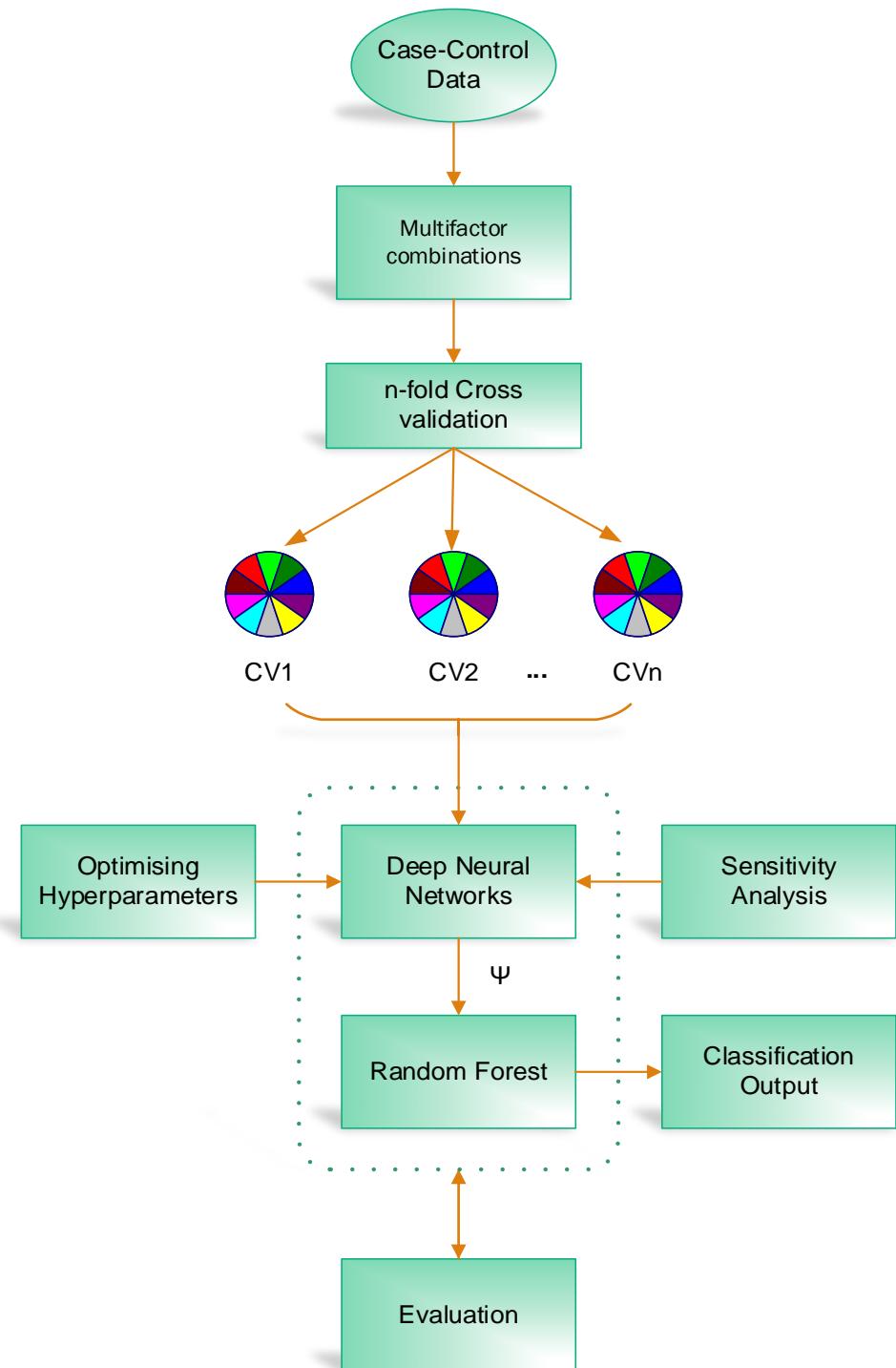


Figure.7.1 Overview of the proposed hybrid deep neural network (DNN) – Random forest (RF) method.

The block diagram of the proposed hybrid method is presented in Figure 7.1 (updated to Figure 6.1 from the previous chapter). In step one, case-control based input data are represented as n-factors, whose subjects are observed by determining their exposure to a phenotype. In multifactor combination stage, factors are combined in n-dimensional space. That is, for three loci combinations with each three genotyping, there are 27 possible three-locus genotyping combinations. In step three, the case-control data is split for training and testing as a part of cross validation. Followed by a hybrid stage where, a deep multilayered neural network is trained to learn the data representation by optimising the hyper-parameters. Subsequently, pre-trained weights and biases are parameterised into a random forest to detect interacting SNPs. As a part of an evaluation stage, the proposed method is evaluated under different simulated scenarios in the absence and presence of noise. The final findings are confirmed on a real dataset by observing the performance metrics of the models in terms of accuracy, auc, logloss, mse, and cross-validation consistency.

### 7.1.1 Parametrisation of Deep Neural Networks

A deep multilayered neural network is trained to detect higher-order SNP interactions as in our previous chapters. However, in this study, each layer is trained using autoencoder due to unsupervised feature learning instead of supervised learning [321, 340]. A multilayered neural network is trained in the proposed hybrid method and it is illustrated in Figure 7.2. It consists of an input layer  $s$ , multiple hidden layers  $h$ , and an output layer  $y$  of interconnected neurons. The definitions below are formulated based on [310, 321].

#### ***Definitions***

##### *Definition 1*

Input layer  $s$  is a three dimensional scalar vector with  $n$  input variables  $s = \{s_1, s_2, s_3, \dots, s_n\}$ ,  $s \in \{1, 2, 3\}$  where due to the duplication of a gene with 2 alleles, each input can be represented either as 1 for dominant homozygous, 2 for heterozygous, and 3 for recessive homozygous genotyping.

##### *Definition 2*

Hidden layers  $h$  in the network with  $L$  hidden layers are  $h = \{h^1, h^2, h^3, \dots, h^L\}$ , where each hidden layer with  $m$  hidden units/nodes comprises of,  $h^L = \{h_{m1}^L, h_{m2}^L, h_{m3}^L, \dots, h_{mL}^L\}$ . Bias  $b$  is a scalar vector  $b = \{b^1, b^2, b^3, \dots, b^L\}$  for  $L$  hidden layers,  $b \in \mathbb{R}^{m*1}$ .

### *Definition 3*

Consider, weighted vector of  $L$  layers connecting input and hidden layers are  $W = \{W^1, W^2, W^3, \dots, W^L\}$ , where  $W \in \mathbb{R}^{n*m}, L \geq 2$ . Each weighted layer with  $m$  nodes is defined by  $W^L = \{W_{1m}^L, W_{2m}^L, W_{3m}^L, \dots, W_m^L\}$ , where  $W^L = \mathbb{R}^{m(L-1)*mL}$ .

### *Definition 4*

Weights connecting hidden and output layer (parameterized by  $\varphi$  and fed into to the random forest to detect SNP interactions) is represented as  $W^{L+1} = \{W_1^{L+1}, W_2^{L+1}, W_3^{L+1}, \dots, W_{mL}^{L+1}\}$ , where  $W \in \mathbb{R}^{mL*1}$  and bias  $b$  for the output layer ( $L + 1$ ) is  $b^{L+1}$ .

### *Definition 5*

Output layer (classification using random forest)  $y$  is a binary variable  $y = \{0,1\}$ , where 0 is a control, and 1 is a case.

DNN receive data in the input layer and transforms non-linearly through multiple hidden layers and the output is predicted in the output layer. Each neuron in the hidden layer is the sum of all weighted inputs to the neuron along with the bias (that represents the neuron's activation threshold).

$$a = W^T S + b \quad (7.1)$$

A non-linear hyperbolic tangent activation function  $f(\cdot)$  is applied to invoke the neuron's output.

$$h = f(a) = f(W^T S + b) \quad (7.2)$$

A non-linear output of a neuron at layer  $L$  is represented by:

$$h^L = f(W^{L^T} h^{(L-1)} + b^L) \quad (7.3)$$

In practice, tanh function converges faster as its range lies in between -1 to 1.

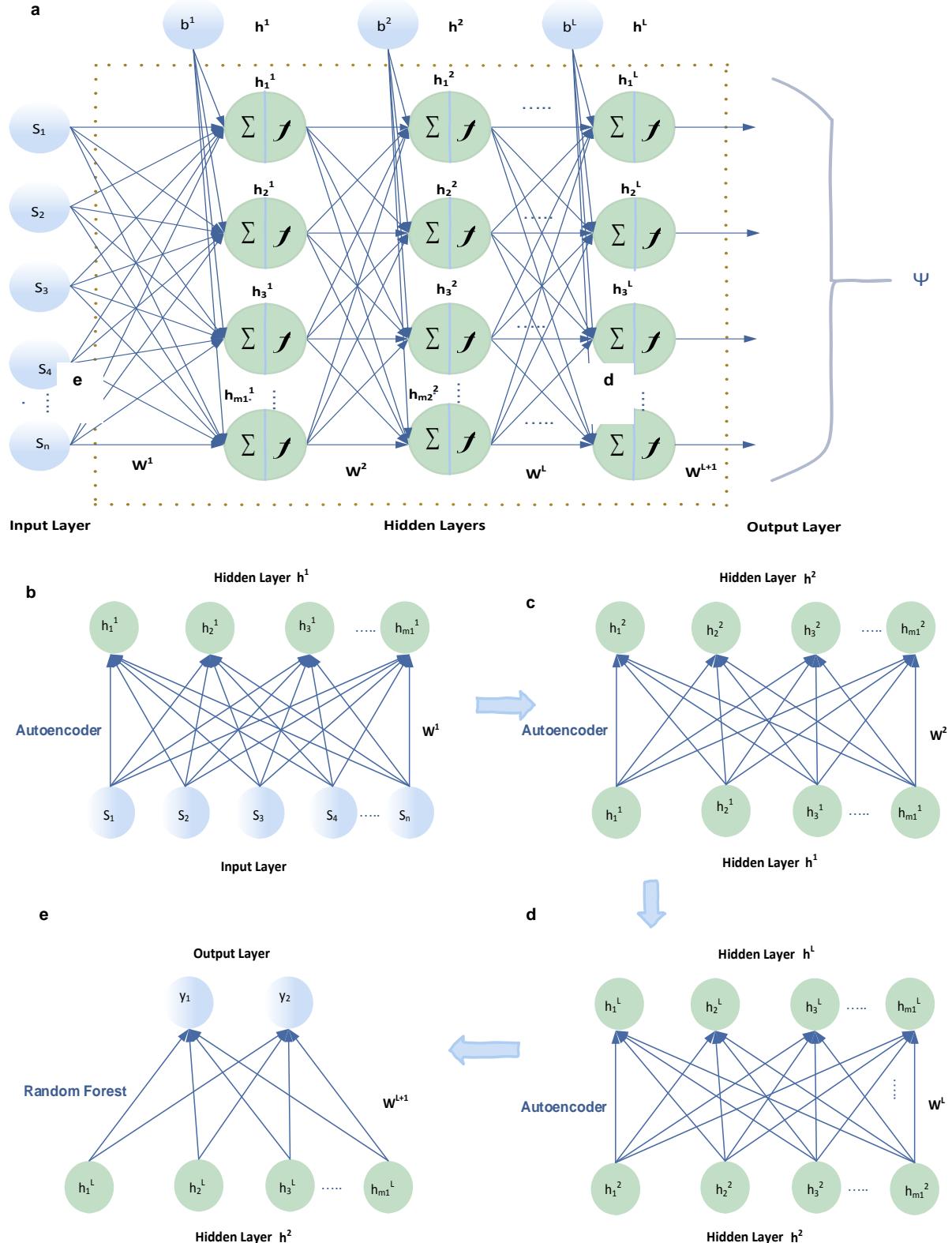


Figure.7.2 An example of Deep neural networks (DNN) with an input layer  $s$ , multiple hidden layers  $h$ , and an output layer  $y$  is parameterized by  $\varphi$ , (b) An autoencoder network used to train the initial parameters of the first layer of a deep neural network, (c) A single layered autoencoder network to train the initial parameters of the second layer, (d) An autoencoder that finds the initial parameters for  $L^{th}$  hidden layer, (e) Classification using Random forest.

$$\tanh(a) = \frac{\exp(a) - \exp(-a)}{\exp(a) + \exp(-a)} \quad (7.4)$$

where  $\tanh(a) \in (-1,1)$

Gradient of  $\tanh(a)$  is calculated as follows:

$$\nabla \tanh(a) = \left( 1 - \left( \frac{\exp(a) - \exp(-a)}{\exp(a) + \exp(-a)} \right)^2 \right) = 1 - \tanh^2(a) \quad (7.5)$$

The example of the proposed DNN shown in Figure 7.2 comprises of three layered network, whose output is parameterized by,  $\varphi$ , and fed as an input to a random forest to predict SNP interactions [340].

$$\begin{aligned} \varphi &= f \left( W^{(3)T} h^{(2)} + b^3 \right) \\ &= f \left( W^{(3)T} \cdot f \left( W^{(2)T} h^{(1)} + b^2 \right) + b^3 \right) \\ \varphi &= f \left( W^{(3)T} \cdot f \left( W^{(2)T} \cdot f \left( W^{(1)T} S + b^1 \right) + b^2 \right) + b^3 \right) \end{aligned} \quad (7.6)$$

## 7.1.2 Back Propagation

The weights and biases between the neurons of DNN layers are learned from input and output samples [341]. This learning process minimises loss function  $L(W)$ , which measures the predicted output with respect to the true output of a sample [340].

$$\{W^1, b^1, W^L, b^L\} = \underset{\{W^1, b^1, W^L, b^L\}}{\arg \min} (L(W)) \quad (7.7)$$

This loss function is minimised by using backpropagation algorithm [321, 341], which computes a gradient of loss function by using chain rule for derivatives. Stochastic gradient descent (SGD) computes derivative of each parameter with respect to the loss function. A parallel version of SDA is used in the proposed method to optimise the loss function by using the memory efficiently. The cross-entropy loss function is used in the proposed method and is given by (based on [341]):

$$L(\varphi; s, y) = -\frac{1}{S} \sum_{s=1}^S (y_s \log(y_s'(y|s; \varphi)) + ((1 - y_s) * \log(1 - y_s'(y|s; \varphi)))) \quad (7.8)$$

Where  $\emptyset_T$  the predicted output and  $y$  is the actual output,  $(s, y) \in \{S\}_{training}$  with  $S$  samples. Gradient of the loss  $L$  with respect to  $\varphi$  is  $\nabla L$ . The derivative of cross-entropy loss function  $\frac{\partial L}{\partial \varphi}(\varphi; s, y)$  is computed using chain rule. Parameter  $\varphi$  is updated with new weights and biases using SGD.

Parameter  $\varphi$  is updated with new weights and biases using SGD.

$$\Delta\varphi = \varphi - \eta * 1/(\lambda) \sum_{(s,y) \in \lambda} \frac{\partial L(\varphi; s, y)}{\partial \varphi} \quad (7.9)$$

Where  $\eta$  is a learning rate and  $\lambda$  is a mini-batch of random subset from  $\{S\}_{training}$ .

### 7.1.3 Random Forest

Random forest [122, 341] is an ensemble of multiple classification and regression trees (CARTs). Each tree is grown from bootstrap sample of the original data using a random subset of total number of predictor variables at node level, rather than considering all possible predictor variables. This results in forest of unpruned trees. Final prediction is obtained based on aggregating the majority votes represented in ensemble of trees grown using bagging (bootstrap aggregating). The observations that are not used in growing trees are used as out-of-bag instance for determining the prediction error. It can also be used to estimate the variable importance.

In summary, the trees in the random forest are grown as follows:

- (a)  $ntree$  bootstrap samples of size  $n$  are drawn from the input data (bootstrap aggregating or bagging). One third of samples are left for estimating error, which is termed as out-of-bag (OOB) data.
- (b) Unpruned CART is grown for each bootstrap sample. At each decision node,  $mtry$  variables are selected randomly for splitting. Trees are grown till each leaf has no fewer than  $nodesize$  cases.

(c) Final prediction is obtained by aggregating the majority votes obtained from  $n_{tree}$  trees grown.

(d) Prediction error is estimated using OOB data.

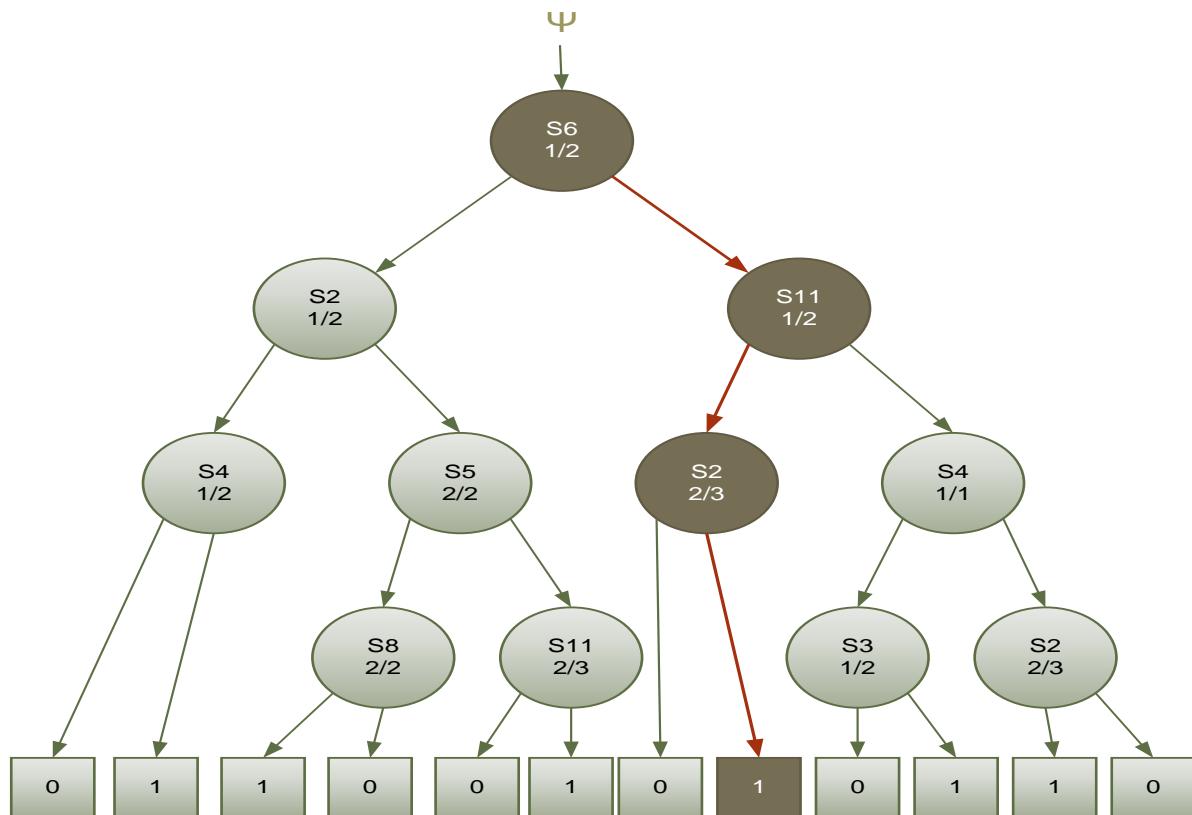


Figure 7.3 An example of a decision tree in the random forest to identify interactions between SNPs. 0 and 1 are the disease status of the subjects. 1, 2, and 3 representing dominant homozygous (AA), heterozygous (aA), and recessive homozygous (aa) genotyping of a biallelic genes.

RF is successfully outperformed many other machine learning algorithms in gene interaction studies as discussed in Chapter 2. It is further robust in the presence of noise [133]. Hence, RF is integrated into DNN for achieving the reliable detection of SNP interactions in the presence of noise and their combined effects. Figure 7.3 is an example of a decision tree with 11 decision nodes and 12 terminal nodes (leaf) to determine SNP interactions related to disease status. The shaded red coloured path flow of the figure represents an individual with interaction between SNP6 (with aa genotype), SNP11 (genotype either AA/aA), and SNP2 (genotype AA), which are more likely to have a disease (status 1). Figure 7.4 illustrates the proposed DNN-RF architecture to detect higher order multi-locus SNP interactions by relating to a classification problem.

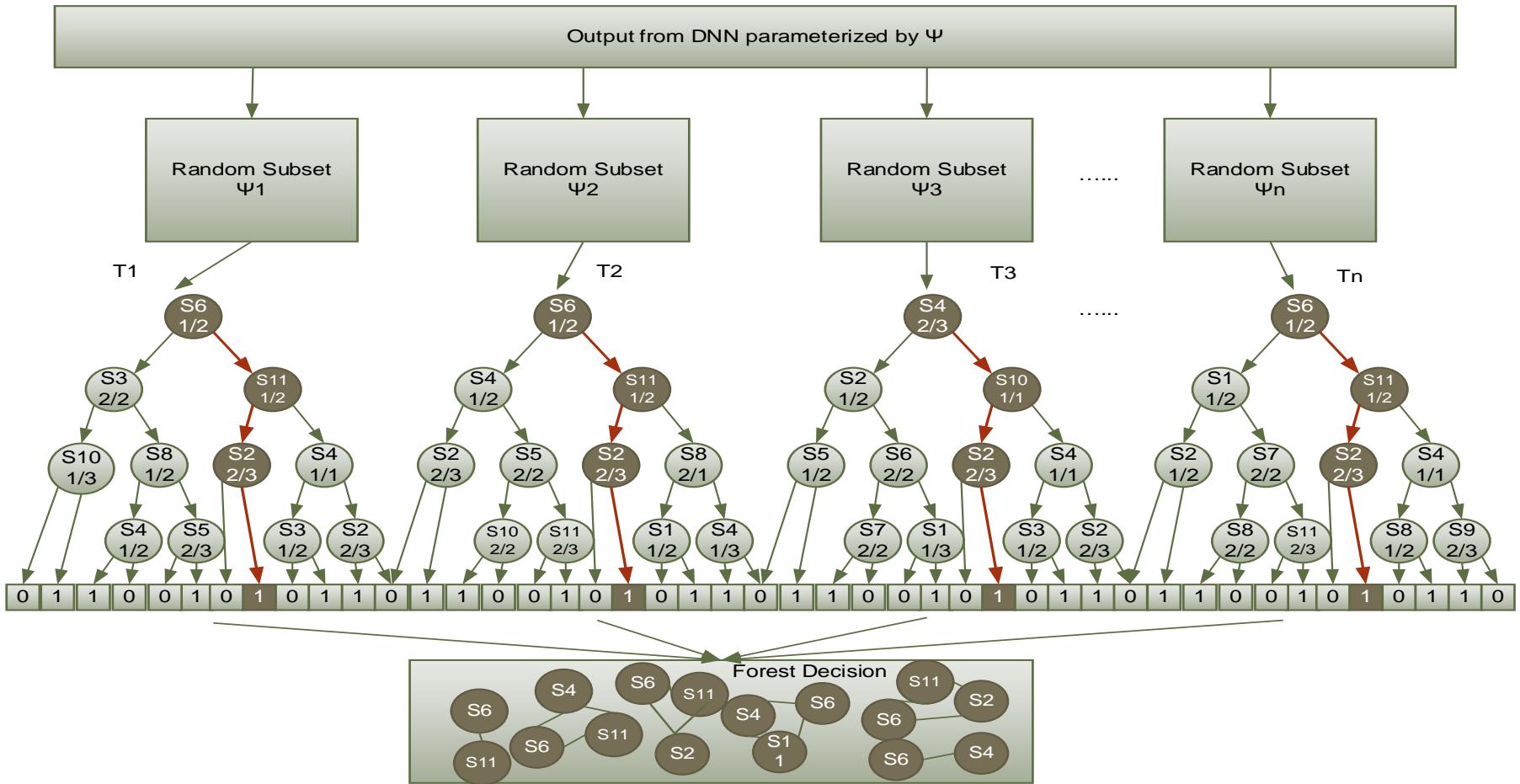


Figure 7.4 Hybrid DNN-RF model (based on [341]). Output of DNN fed into RF. The size of the input fed into RF is same as that of number of neurons in the output of the hidden layer ( $L$ ). RF constructs  $n$  number of trees using random subset of variables. Each decision node  $n$  is Parameterised by  $\varphi$ , whose function is binary and routing is deterministic either to left or right of the subtree based on output of decision function  $\theta_n(\cdot; \varphi)$ . Once the sample reaches prediction node (leaf or terminal node)  $\ell$ , the related prediction of tree holds the probability distribution over class-label. Final decision of the forest is obtained by aggregating the majority votes in ensemble of trees grown.

The output of DNN are fed to RF, whose decision trees are parameterized by  $\varphi$  [340, 341]. The final prediction is obtained by aggregating the majority votes of ensemble of trees grown in the forest. The definitions below are formulated as reported in [122, 341].

### **Definitions**

#### *Definition 6*

Each decision node is parameterized by  $\varphi$  and splits to either right or left nodes with decision function defined by  $\theta(s; \varphi)$ , where  $s \in \{1, 2, 3\}$  due to biallele SNPs and  $\theta$  is binary, whose routing is deterministic by ending at leaf node (terminal node)  $\ell$ . Sample  $s_n$  parameterised by  $\varphi_m$  reaches to decision node  $\theta_d$ , where it is sent to either left or right of the child node, and ends at a leaf node  $l_m$ . Where  $\theta = \{\theta_1, \theta_2, \theta_3, \dots, \theta_d\}$  decision nodes, and  $\ell = \{l_1, l_2, l_3, \dots, l_m\}$  leaf nodes in a tree.

#### *Definition 7*

A random forest is an ensemble of decision trees  $F = \{T_1, T_2, T_3, \dots, T_n\}$ , where  $n$  trees are grown in a forest using random subset of variables. Final prediction of a sample  $s$  in the random forest is determined by averaging the prediction output of each tree and is represented by:

$$\emptyset_F[y|s] = \frac{1}{n} \sum_{u=1}^n \emptyset_{T_u}[y|s] \quad (7.10)$$

## **7.2 Variable importance**

Variable importance is calculated based on [27, 182]. The important feature of RF is to measure importance of each predictor variables. Gini importance and mean decrease in accuracy (MDA) are some of the popular approaches used in the literature. Although, Gini importance is easy to compute, it shows bias in selecting the variables with different categories [342]. Hence, MDA measure is used to compute the importance of SNP interactions identified by the models [27]. The unscaled MDA estimation for each tree  $T$  in overall trees of the forest  $F$  is calculated as reported by Schwarz:

$$\omega = \frac{1}{|F|} \sum_{T=1}^F \rho - \rho' \quad (7.11)$$

Where  $\rho$  is the prediction accuracy computed using OOB sample in each tree, and  $\rho'$  is the prediction accuracy computed using OOB sample in each tree after randomly permuted. Scaled MDA (often known as z score) can also be used to measure variable importance as reported by [27].

$$\omega_{scaled} = \frac{\omega}{\sqrt{\frac{\varepsilon^2}{|F|}}} \quad (7.12)$$

Where  $\varepsilon^2$  is the standard error across all trees in the forest  $F$  [27].

$$\varepsilon^2 = \frac{1}{|F|} \sum_{T=1}^F N_{OOB}(\rho - \rho')^2 - \omega \quad (7.13)$$

Substituting equation (7.14) into equation (7.16),

$$\varepsilon^2 = \frac{1}{|F|} \sum_{T=1}^F (N_{OOB}(\rho - \rho')^2 + (\rho' - \rho)) \quad (7.14)$$

## 7.3 SNP Interactions

Interactions are detected based on [182]. Variable importance is used to rank the higher-order interacting SNPs. Highly ranked interactions are considered to be highly associated with the disease. However, variable importance measures in RF do not specify the effect of variables in the tree either due to main or interaction effect. Hence, performance of the RF is calculated with respect to the heritability to identify main and interaction effects [182]. Heritability is a common measure of variance in a phenotype that attributes to genetic variation [343]. It can also measure the effect of a locus on a disease. Winham considered  $C$  is a complex disease which is controlled by loci  $A$  and  $B$ , with genotypes X and Y at each locus are 1, 2, and 3. The total heritability  $\alpha^2$  due to two loci  $A$  and  $B$  is defined by Winham as:

$$\alpha^2 = \frac{1}{\sigma(C)(1 - \sigma(C))} \sum_{X=1}^3 \sum_{Y=1}^3 \sigma(G_{XY}) (\sigma(C|G_{XY}) - \sigma(C))^2 \quad (7.15)$$

Where,  $\sigma(C)$  is the prevalence of the disease,  $\sigma(G_{XY})$  is the frequency of genotype combination, and  $\sigma(C|G_{XY})$  is the penetrance of the disease [344].

Heritability  $\beta^2$  due to marginal effect of SNP at locus A is given by:

$$\beta^2 = \frac{1}{\sigma(C)(1 - \sigma(C))} \sum_{X=1}^3 \left( \sum_{Y=1}^3 \sigma(G_{XY}) \right) * \left( \sum_{Y=1}^3 \sigma(G_{XY}) (\sigma(C|G_{XY}) - \sigma(C)) \right)^2 \quad (7.16)$$

Heritability  $\gamma^2$  due to marginal effect of SNP at locus B is given by:

$$\gamma^2 = \frac{1}{\sigma(C)(1 - \sigma(C))} \sum_{Y=1}^3 \left( \sum_{X=1}^3 \sigma(G_{XY}) \right) * \left( \sum_{X=1}^3 \sigma(G_{XY}) (\sigma(C|G_{XY}) - \sigma(C)) \right)^2 \quad (7.17)$$

Heritability due to the interaction effect between SNPs at locus A and locus B on a disease  $\mu_I^2$  (conditional dependency between SNPs at both loci on a phenotype) is given by:

$$\mu_I^2 = \alpha^2 - \beta^2 - \gamma^2 \quad (7.18)$$

From the above four definitions reported by [182], it is inferred that SNP at locus A possess main effect if  $\beta^2 > 0$ , and SNP at locus B possess main effect if  $\gamma^2 > 0$ , such that  $\mu_I^2 = 0$  for main effect only. Interaction effect exist between SNPs at loci A and B if  $\mu_I^2 > 0$ .

## 7.4 Simulated datasets

### 7.4.1 Simulated datasets 1

Six two-locus epistasis models with various penetrance functions and allele frequencies ( $p$ , and  $q$ ) are generated based on [36] using GAMETES tool [36, 221]. Where,  $p$  is a minor allele frequency (MAF) of 0.5, 0.25, or 0.1, and  $q$  is a common allele frequency of 0.5, 0.75, or 0.9. Case-control based datasets are generated similarly for all six epistasis models under various scenarios using GAMETES tool [36, 221]. Genotypes are generated according to Hardy-Weinberg proportions. Case-control based datasets are generated similarly for all six epistasis models under various scenarios as presented in the Section 3.2.1 in the absence of noise.

For each model, 16 scenarios were developed to simulate the datasets in the absence and the presence of noise, due to genotyping error (GE), missing data (MS), genetic heterogeneity (GH), and phenocopy (PC) [36]. First scenario is generated in the absence of noise, and remaining fifteen scenarios are generated in presence of noise (GE, MS, GH, and PC), and their combinations (GE-GH, GE-GH-MS, GE-GH-PC, GE-GH-PC-MS, GE-MS, GE-PC, GE-PC-MS, GH-MS, GH-PC, GH-PC-MS, PC-MS). For each scenario, 100 case-control based datasets (200 cases and 200 controls) are simulated with two functional SNPs for two-locus models. Hence, 1600 datasets are generated for each model. In total, 9,600 datasets are simulated according to Hardy-Weinberg proportions.

### 7.4.2 Simulated datasets 2

Table 7.1 presents 26 two-locus epistasis models generated using latest version of GAMETES tool [221]. GAMETES generate pure, strict, complex biallelic SNP disease models with random architecture. The noisy data is generated with respect to heritability (the less heritability, the more noise). This noise is generated randomly from any source for the simulation purposes. The datasets in this study were generated based on the studies performed by [108] [345] for MAFs (0.2, and 0.4). Further, this study is extended by generating models for MAFs (0.1, 0.2, 0.3, and 0.4) to observe performance of the proposed method in various simulated scenarios. The datasets are simulated by varying heritability (0.01, 0.025, 0.05, 0.1, 0.1, 0.2, 0.3, 0.4), MAFs (0.1, 0.2, 0.3, and 0.4), sample size (200, 400, 800, and 1600), and case-control ratio (1:1, 1:2, 1:4). For each simulated settings, 50 datasets are generated with two functional SNPs. Models for MAF 0.1 with heritability 0.3 and 0.4 could not be generated by GAMETES due to low likelihood of finding models with higher heritability. Hence, in total, 15600 datasets are generated based on Hardy-Weinberg proportions.

**Table 7.1:** Epistasis models generated using various penetrance functions, allele frequencies p and q, and heritability H using GAMETES tool [221]

Model 7 p=0.1,q=0.9, H=0.01	Model 8 p=0.1,q=0.9, H=0.025	Model 9 p=0.1,q=0.9, H=0.05
0.036	0.017	0.028
0.017	0.102	0.003
0.022	0.03	0.919

Model 10	Model 11	Model 12

p=0.1,q=0.9, H=0.1

0.826	0.964	0.866
0.965	0.341	0.82
0.856	0.865	0.199

p=0.1,q=0.9, H=0.2

0.773	0.987	0.922
0.992	0.033	0.326
0.832	0.733	0.756

p=0.2,q=0.8, H=0.01

0.874	0.824	0.884
0.827	0.92	0.854
0.856	0.909	0.48

Model 13

p=0.2,q=0.8, H=0.025

0.075	0.124	0.038
0.123	0.019	0.102
0.043	0.092	0.807

Model 14

p=0.2,q=0.8, H=0.05

0.824	0.937	0.848
0.927	0.714	0.978
0.922	0.828	0.136

Model 15

p=0.2,q=0.8, H=0.1

0.241	0.056	0.179
0.058	0.431	0.11
0.165	0.139	0.742

Model 16

p=0.2,q=0.8, H=0.2

0.621	0.93	0.699
0.925	0.305	0.838
0.739	0.759	0.182

Model 17

p=0.2,q=0.8, H=0.3

0.532	0.931	0.688
0.936	0.127	0.656
0.648	0.737	0.389

Model 18

p=0.2,q=0.8, H=0.4

0.472	0.959	0.792
0.944	0.072	0.347
0.919	0.093	0.571

Model 19

p=0.3,q=0.7, H=0.01

0.474	0.449	0.464
0.499	0.4	0.554
0.429	0.519	0.384

Model 20

p=0.3,q=0.7, H=0.025

0.553	0.528	0.579
0.478	0.649	0.419
0.6	0.459	0.646

Model 21

p=0.3,q=0.7, H=0.05

0.394	0.636	0.353
0.622	0.352	0.438
0.399	0.585	0.564

Model 22

p=0.3,q=0.7, H=0.1

0.599	0.327	0.491
0.294	0.681	0.497
0.644	0.317	0.288

Model 23

p=0.3,q=0.7, H=0.2

0.578	0.359	0.259
0.209	0.752	0.436
0.649	0.223	0.512

Model 24

p=0.3,q=0.7, H=0.3

0.449	0.773	0.157
0.876	0.251	0.276
0.308	0.784	0.876

Model 25

p=0.3,q=0.7, H=0.4

0.498	0.425	0.567
0.107	0.811	0.888
0.782	0.192	0.101

Model 26

p=0.4,q=0.6, H=0.01

0.443	0.381	0.314
0.339	0.426	0.411
0.437	0.318	0.514

Model 27

p=0.4,q=0.6, H=0.025

0.641	0.433	0.438
0.455	0.531	0.56
0.371	0.61	0.511

Model 28

p=0.4,q=0.6, H=0.05

0.389	0.67	0.697
0.666	0.545	0.452
0.71	0.442	0.659

Model 29

p=0.4,q=0.6, H=0.1

0.254	0.598	0.22
0.438	0.325	0.627
0.702	0.266	0.211

Model 30

p=0.4,q=0.6, H=0.2

0.079	0.394	0.193
0.457	0.161	0.047
0.006	0.187	0.982

Model 31

p=0.4,q=0.6, H=0.3

0.836	0.149	0.057
0.149	0.533	0.448

Model 32

p=0.4,q=0.6, H=0.4

0.017	0.804	0.986
0.918	0.392	0.196

0.054	0.45	0.912	0.644	0.453	0.629
-------	------	-------	-------	-------	-------

## 7.5 Evaluations

The performance of the models during training, validation, and testing are evaluated by determining model's metrics. The performance of the models in the presence of noise due to GE, PC, GH, and MS are also evaluated based on [36]. Training speed and time to execute the models are evaluated by varying width and depth of the network, along with various activation functions. The overall best model with highest prediction accuracy and lowest logloss along with the highest cross validation consistency (CVC) is selected. The final results are statistically evaluated by w-test, whose p-values are compared with 0.05 in determining the statistical significance of the findings.

## 7.6 Experimental Results

Number of simulated datasets in the presence and absence of noise due to GE, GH, PC, and MS, and their combined effects are evaluated on the DNN-RF method [36]. 5% of GE is generated with overrepresentation of one allele. GH is simulated for 50% of the data with two different two-locus combinations (SNP5-SNP10, and SNP3-SNP4) to increase the risk of a disease. PC is generated for 50% of the cases, which are considered to have low risk genotypes according to the epistasis models. The cases are assumed to be affected due to environmental factors. 5% of the MS data is simulated by removing genotypes randomly. Several experimental results are demonstrated on the DNN-RF hybrid method to improve the identification of two-locus SNP interactions responsible for a disease. On an average, the accuracy of the proposed model is much higher than the previous methods. The experimental findings are further confirmed over a real world dataset [337].

### 7.6.1 Simulated Studies

#### 7.6.1.1 Simulation Study 1

In this study, the power of the proposed DNN-RF method is compared with the previously trained DNN [45] in the absence and the presence of noise. Power of DNN-RF

is the estimation of number of times the method correctly detects the functional SNPs (SNP5, and SNP 10) among 100 datasets for each model. Table 7.2 summarises the power of DNN to detect two-locus SNP interactions in the absence, and presence of different forms of noise. Power is represented in the form of three values; first one represents the number of times the functional SNPs are detected with highest ranking (two-locus SNP interactions are ranked by using MDA variable importance measure) among 100 datasets, second value represents the functional SNPs occurrence in top 10 ranks, and third value represents functional SNP in top 20 ranks. Similarly, Table 7.3 shows the power of DNN-RF. On an average, it is observed that the power of DNN-RF outperformed in all the simulated scenarios of six epistasis models compared with DNN.

The power of DNN-RF is 100% for all six epistasis models in the absence of noise and in the presence of 5% of MS. In the presence of 5% of GE, the power is almost equal to 100% for all the models. The power of the method in the presence of 50% of PC is almost 100% for model 1 ( $p=0.5$ ,  $q=0.5$ ,  $H=0.053$ ) and model 2 by gradually reducing in Model 4 ( $p=0.25$ ,  $q=0.75$ ,  $H=0.033$ ) (38%), Model 3 ( $p=0.25$ ,  $q=0.75$ ,  $H=0.016$ ) (47%), Model 5 ( $p=0.1$ ,  $q=0.9$ ,  $H=0.02$ ) (53%), and Model 6 ( $p=0.1$ ,  $q=0.9$ ,  $H=0.015$ ) (72%). The power of the method in the presence of 50% of GH falls to 45%, 51%, 48%, 38%, 38%, and 49% for models 1-6 respectively. The combined effect of PC-MS performs well for model 1 ( $p=0.5$ ,  $q=0.5$ ,  $H=0.053$ ) and model 2. However, power dropped for models 3-6 by taking values of 54%, 48%, 43%, and 57%, slightly less than the impact of individual effect on these models. The combined effect of GE-MS also performed almost well for all six epistasis models. The power of GE-PC and GE-PC-MS is about average 95% for models 1 and 2 along with fall of about 40% on an average for all the other models 3-6. The power of the method for combined effect of GE-GH, GE-GH-MS, and GH-MS falls about 50%. It is observed that the combined effect of GE-GH-PC, GE-GH-PC-MS, GH-PC, and GH-PC-MS has the dramatic fall in the power. The power drastically reduces in all the six models due to the combined effect of GH-PC and has the greatest impact on the method.

Figure 7.5 compares the power of previously proposed DNN presented in chapter 6 (represented in Table 7.2) and the proposed DNN-RF (represented in Table 7.3) methods. It is clearly observed that the power of DNN-RF is consistently high in the presence of all types of noise compared with the previously proposed DNN.

**Table 7.2:** Power of DNN in various simulated scenarios among six models to detect two-locus SNP interactions.

Noise	Model1		Model2		Model3		Model4		Model5		Model6	
	Power	Importance	Power	Importance	Power	Importance	Power	Importance	Power	Importance	Power	Importance
GE	10\30\40	0.9185232	8\37\44	0.92027436	2\10\21	0.8908237	1\11\19	0.8826586	1\6\11	0.8592594	2\10\19	0.8731572
GE+GH	0\12\22	0.8865167	0\10\21	0.88580856	0\6\23	0.8843263	2\14\20	0.8826038	0\6\15	0.8608146	0\6\13	0.8546887
GE+GH+MS	2\8\13	0.8796005	4\15\25	0.88958859	0\7\17	0.8805794	0\6\15	0.8778808	1\6\12	0.8612048	1\10\13	0.8569457
GE+GH+PC	0\11\17	0.884614	1\13\17	0.88238141	0\10\17	0.8884239	0\8\21	0.8827364	0\6\10	0.8568603	1\9\11	0.8539244
GE+GH+PC+MS	2\11\18	0.8824771	3\12\25	0.88956878	1\8\16	0.8752033	0\6\18	0.877409	1\5\10	0.856791	0\6\10	0.8508098
GE+MS	4\23\38	0.9113822	13\29\37	0.91288119	1\12\21	0.885192	2\8\18	0.8837899	2\11\17	0.8619942	0\8\14	0.8630082
GE+PC	2\10\17	0.8865794	0\8\19	0.88436952	1\13\24	0.8881893	1\10\20	0.8866258	1\8\13	0.8560701	0\6\12	0.8525487
GE+PC+MS	1\10\19	0.8850328	2\14\23	0.89026255	0\10\24	0.8835444	2\8\16	0.8798108	0\7\17	0.8601303	0\5\14	0.867672
GH	0\5\19	0.8858407	0\9\18	0.88871727	1\12\22	0.8836247	1\8\22	0.8781621	0\6\12	0.8543147	0\6\12	0.8577496
GH+MS	1\13\27	0.8881948	3\12\16	0.88168822	1\14\22	0.8845436	0\13\17	0.8828589	1\10\15	0.8639469	0\10\20	0.8670642
GH+PC	1\6\15	0.8779375	1\10\16	0.88568725	1\10\16	0.8822872	0\10\23	0.8853507	0\10\20	0.8711614	0\7\15	0.8615859
GH+PC+MS	3\8\15	0.883542	0\10\19	0.88283407	1\5\19	0.8798247	0\11\17	0.8820321	0\4\9	0.8529785	1\5\9	0.8607697
MS	5\30\42	0.9126057	5\28\43	0.9159232	2\12\27	0.8854451	0\6\13	0.8727755	1\3\11	0.859998	0\8\16	0.8661739
No error	11\37\48	0.9264184	5\27\41	0.90886191	2\18\29	0.8886084	1\9\17	0.8797939	0\10\14	0.8601424	2\11\19	0.8717422
PC	1\6\14	0.8848343	3\15\24	0.89197047	3\9\22	0.8848929	1\8\23	0.8908746	0\7\13	0.8608917	0\7\17	0.8582902
PC+MS	1\10\20	0.8890396	0\13\20	0.88373868	0\8\14	0.877729	3\8\12	0.8847574	1\6\18	0.8656147	0\8\16	0.8590068

**Table 7.3:** Power of DNN-RF in various simulated scenarios among six models to detect two-locus SNP interactions.

Noise	Model1		Model2		Model3		Model4		Model5		Model6	
	Power	Importance										
GE	90\90\90	1	99\99\99	1	98\98\98	1	97\99\100	0.9949635	99\100\100	0.9981631	99\100\100	0.9990714
GE+GH	48\90\90	0.8824058	50\100\100	0.8993395	33\87\95	0.8410114	38\83\91	0.8659039	36\80\90	0.7831425	40\86\93	0.8319533
GE+GH+MS	52\89\90	0.8918551	33\85\94	0.85887749	37\85\92	0.8483482	42\83\91	0.8706476	37\81\88	0.800231	36\92\97	0.8379415
GE+GH+PC	16\57\66	0.7876617	37\86\94	0.86528047	15\41\58	0.745041	8\44\68	0.7338524	13\48\73	0.689669	15\57\76	0.7102578
GE+GH+PC+MS	26\67\78	0.8201348	33\85\94	0.85887749	7\36\62	0.7021804	8\41\63	0.7315057	5\45\70	0.6490605	8\55\71	0.6825867
GE+MS	90\90\90	1	100\100\100	1	100\100\100	1	98\100\100	0.9984826	93\99\100	0.9882789	100\100\100	1
GE+PC	85\92\92	0.9896108	100\100\100	1	55\95\99	0.9365313	53\83\89	0.9056961	60\89\93	0.9083858	65\93\98	0.9412304
GE+PC+MS	95\100\100	0.9961153	99\100\100	0.99991752	53\90\94	0.9146774	56\80\90	0.9030355	44\83\91	0.8660377	64\91\95	0.9141585
GH	45\100\100	0.8915172	51\100\100	0.89740342	48\92\96	0.8646593	38\83\90	0.8622593	38\81\92	0.8021247	49\95\98	0.8669313
GH+MS	43\100\100	0.8587725	47\100\100	0.87683438	36\86\91	0.855682	33\83\96	0.8572148	36\85\92	0.7907713	45\91\94	0.8234406
GH+PC	19\63\83	0.7928737	39\85\92	0.8689831	11\42\62	0.7414824	5\31\57	0.7081456	14\53\73	0.7106355	14\52\72	0.7194238
GH+PC+MS	24\66\81	0.8034188	35\85\91	0.82834247	11\33\58	0.7206934	9\43\63	0.7397768	16\60\78	0.7125411	23\63\74	0.739219
MS	100\100\100	1	100\100\100	1	100\100\100	1	100\100\100	1	95\100\100	0.988724	100\100\100	1
No error	100\100\100	1	100\100\100	1	100\100\100	1	99\100\100	0.9968939	100\100\100	1	100\100\100	1
PC	93\100\100	0.9892198	100\100\100	1	47\83\91	0.901953	38\74\84	0.8687428	53\88\94	0.8863391	72\92\97	0.9276377
PC+MS	94\100\100	0.9995187	100\100\100	1	54\84\94	0.9185419	48\83\91	0.8888536	43\82\91	0.8380772	57\88\95	0.8932816

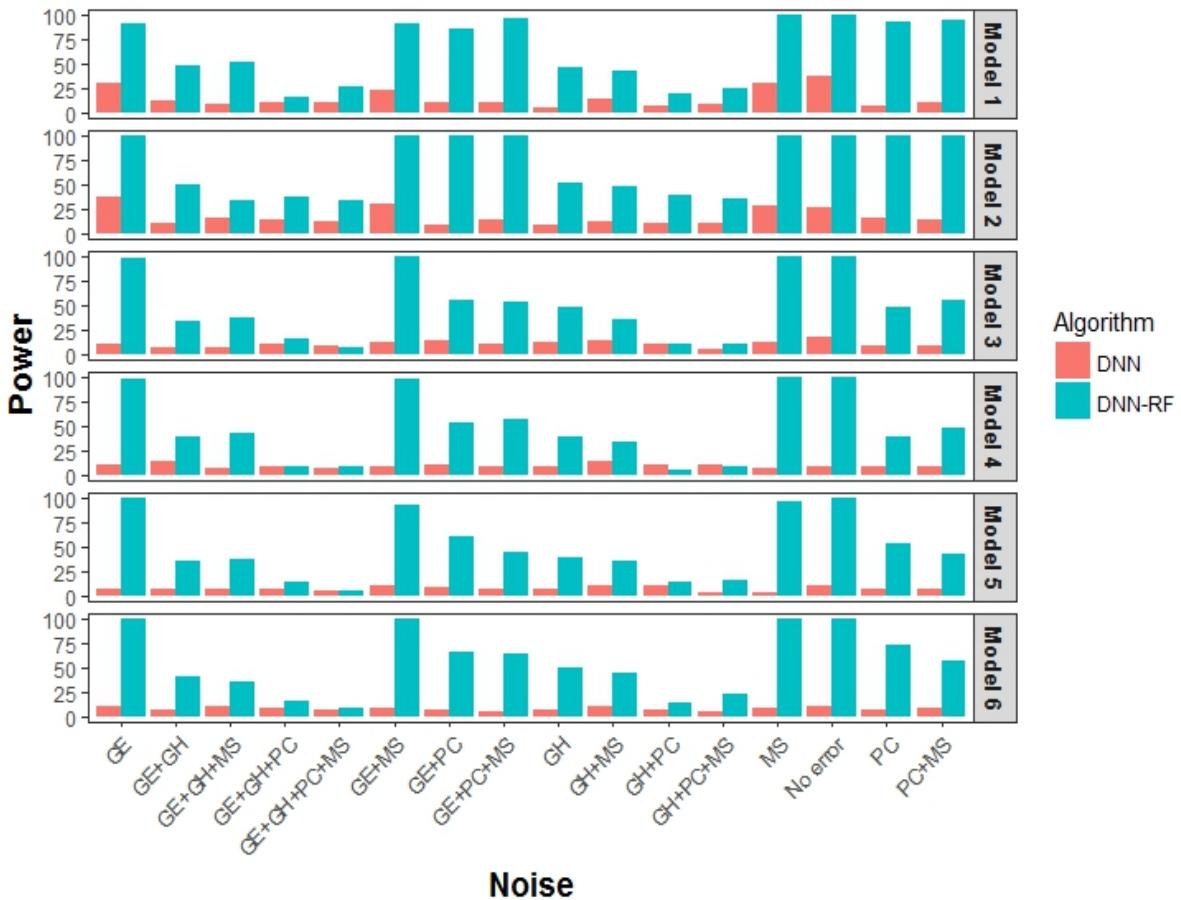


Figure 7.5 Comparison of power of DNN (Table 7.2) and DNN-RF (Table 7.3) in the presence of noise for the balanced datasets.

### 7.6.1.2 Simulation Study 2

Several experiments were performed for simulation study 2 under various scenarios for both balanced (case-control ratio 1:1) and imbalanced (case-control ratios 1:2 and 1:4) datasets. The power of the DNN-RF is evaluated for all the 26 models. The number of times method correctly identifies known two-locus interaction from 50 datasets of each simulated scenario is tabulated in Table 7.4 as ready reference.

**Table 7.4:** Power of DNN-RF in various simulated scenarios among 24 models to detect two-locus SNP interactions

Models	MAF	Heritability	Sample Size	Power		
				Ratio 1:1	Ratio 1:2	Ratio 1:4
Model 7	0.1	0.01	200	2	2	0
Model 8	0.1	0.025	200	6	7	4
Model 9	0.1	0.05	200	8	7	8
Model 10	0.1	0.1	200	18	13	12

Model 11	0.1	0.2	200	22	29	24
Model 7	0.1	0.01	400	2	0	0
Model 8	0.1	0.025	400	17	8	7
Model 9	0.1	0.05	400	20	19	11
Model 10	0.1	0.1	400	33	27	25
Model 11	0.1	0.2	400	36	43	43
Model 7	0.1	0.01	800	10	12	4
Model 8	0.1	0.025	800	30	24	11
Model 9	0.1	0.05	800	36	32	21
Model 10	0.1	0.1	800	42	44	39
Model 11	0.1	0.2	800	46	48	46
Model 7	0.1	0.01	1600	19	12	11
Model 8	0.1	0.025	1600	43	34	36
Model 9	0.1	0.05	1600	49	43	40
Model 10	0.1	0.1	1600	48	48	45
Model 11	0.1	0.2	1600	50	48	50
Model 12	0.2	0.01	200	1	0	0
Model 13	0.2	0.025	200	7	2	1
Model 14	0.2	0.05	200	12	8	7
Model 15	0.2	0.1	200	26	18	10
Model 16	0.2	0.2	200	36	33	23
Model 17	0.2	0.3	200	34	38	39
Model 18	0.2	0.4	200	38	41	35
Model 12	0.2	0.01	400	3	2	0
Model 13	0.2	0.025	400	8	4	4
Model 14	0.2	0.05	400	24	15	12
Model 15	0.2	0.1	400	33	32	25
Model 16	0.2	0.2	400	42	46	41
Model 17	0.2	0.3	400	45	44	43
Model 18	0.2	0.4	400	50	42	44
Model 12	0.2	0.01	800	3	1	1
Model 13	0.2	0.025	800	16	15	11
Model 14	0.2	0.05	800	41	42	27
Model 15	0.2	0.1	800	46	46	40
Model 16	0.2	0.2	800	47	49	47
Model 17	0.2	0.3	800	49	47	48
Model 18	0.2	0.4	800	49	50	49
Model 12	0.2	0.01	1600	9	3	1
Model 13	0.2	0.025	1600	35	30	20
Model 14	0.2	0.05	1600	44	42	41
Model 15	0.2	0.1	1600	50	47	48
Model 16	0.2	0.2	1600	48	49	49
Model 17	0.2	0.3	1600	49	49	49
Model 18	0.2	0.4	1600	50	49	48
Model 19	0.3	0.01	200	1	0	0

Model 20	0.3	0.025	200	5	4	3
Model 21	0.3	0.05	200	4	5	4
Model 22	0.3	0.1	200	13	20	8
Model 23	0.3	0.2	200	30	30	18
Model 24	0.3	0.3	200	35	34	33
Model 25	0.3	0.4	200	41	38	33
Model 19	0.3	0.01	400	2	2	1
Model 20	0.3	0.025	400	12	3	3
Model 21	0.3	0.05	400	21	20	4
Model 22	0.3	0.1	400	34	22	17
Model 23	0.3	0.2	400	43	43	36
Model 24	0.3	0.3	400	43	45	47
Model 25	0.3	0.4	400	47	47	45
Model 19	0.3	0.01	800	1	2	5
Model 20	0.3	0.025	800	19	8	5
Model 21	0.3	0.05	800	26	21	9
Model 22	0.3	0.1	800	44	41	36
Model 23	0.3	0.2	800	50	45	47
Model 24	0.3	0.3	800	47	50	50
Model 25	0.3	0.4	800	49	50	47
Model 19	0.3	0.01	1600	2	5	3
Model 20	0.3	0.025	1600	19	18	13
Model 21	0.3	0.05	1600	35	39	21
Model 22	0.3	0.1	1600	49	48	48
Model 23	0.3	0.2	1600	50	49	49
Model 24	0.3	0.3	1600	49	50	47
Model 25	0.3	0.4	1600	50	47	50
Model 26	0.4	0.01	200	2	2	2
Model 27	0.4	0.025	200	6	2	2
Model 28	0.4	0.05	200	9	5	5
Model 29	0.4	0.1	200	18	19	14
Model 30	0.4	0.2	200	29	24	18
Model 31	0.4	0.3	200	27	30	23
Model 32	0.4	0.4	200	39	38	31
Model 26	0.4	0.01	400	3	0	0
Model 27	0.4	0.025	400	9	5	2
Model 28	0.4	0.05	400	10	11	7
Model 29	0.4	0.1	400	34	21	18
Model 30	0.4	0.2	400	44	39	35
Model 31	0.4	0.3	400	45	43	39
Model 32	0.4	0.4	400	48	47	47
Model 26	0.4	0.01	800	4	4	2
Model 27	0.4	0.025	800	10	12	9
Model 28	0.4	0.05	800	23	26	22
Model 29	0.4	0.1	800	41	39	33

Model 30	0.4	0.2	800	46	47	44
Model 31	0.4	0.3	800	50	49	48
Model 32	0.4	0.4	800	49	49	49
Model 26	0.4	0.01	1600	6	5	2
Model 27	0.4	0.025	1600	23	23	27
Model 28	0.4	0.05	1600	40	32	10
Model 29	0.4	0.1	1600	45	46	43
Model 30	0.4	0.2	1600	50	49	49
Model 31	0.4	0.3	1600	50	47	48
Model 32	0.4	0.4	1600	48	50	49

Figure 7.6 shows the power of DNN-RF for balanced datasets. The power of the proposed method is high for all the models, when  $H=0.4$ . It is observed that the power of DNN-RF is gradually reduced as the proportion of noise increased. At  $H=0.01$ , the method performed poorly for all the value of MAFs. The performance of the method improved slightly by increasing the sample size. Power is almost 100% for heritability 0.3, and 0.4 with MAF values 0.3 and 0.4 for sample size 800 and 1600. Power is nearly 90% when  $H=0.2$  and  $MAF = 0.2$  for sample size 400. The model performed to the maximum when MAF is 0.1 and  $H$  values are 0.05, 0.1, and 0.2 for sample size 1600. On an average, the power of the proposed method gradually increased by increasing sample size and MAF values for all the balanced datasets.

Figure 7.7 illustrates the power of DNN-RF when the case-control ratio is 1:2. As expected the power of the method is improved as the sample size is increased. However, the power is dropped when compared with the same simulated settings in balanced datasets. Power is almost equal to 100% when heritability and MAFs values are 0.3, and 0.4 for sample size 800 and 1600 respectively. Figure 7.8 shows the results of power analysis of DNN-RF when the case-control ratio is 1:4. Power is less than 5% when heritability is 0.01 for all the MAF values. Unexpectedly, power did not improve by increasing the sample size. Power drastically fell to 10% at  $H=0.05$ , and  $MAF = 0.4$ . Table 7.5 summarises the average power of DNN-RF with respect to 100% for case-control ratios 1:1, 1:2, and 1:4 by varying sample size (200, 400, 800 and 1600). It also summarises the classification accuracy of the method for various sample size.

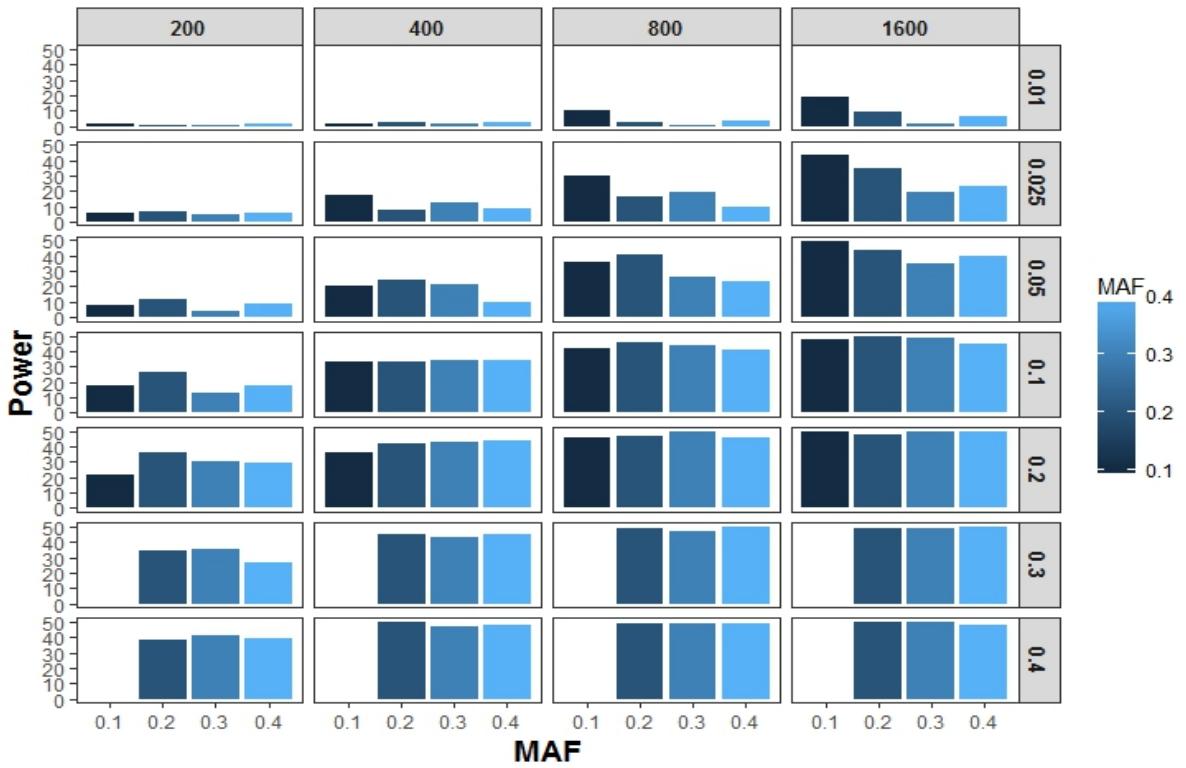


Figure.7.6 Power of DNN-RF for ratio 1:1 for various minor allele frequencies, heritability, and sample size.

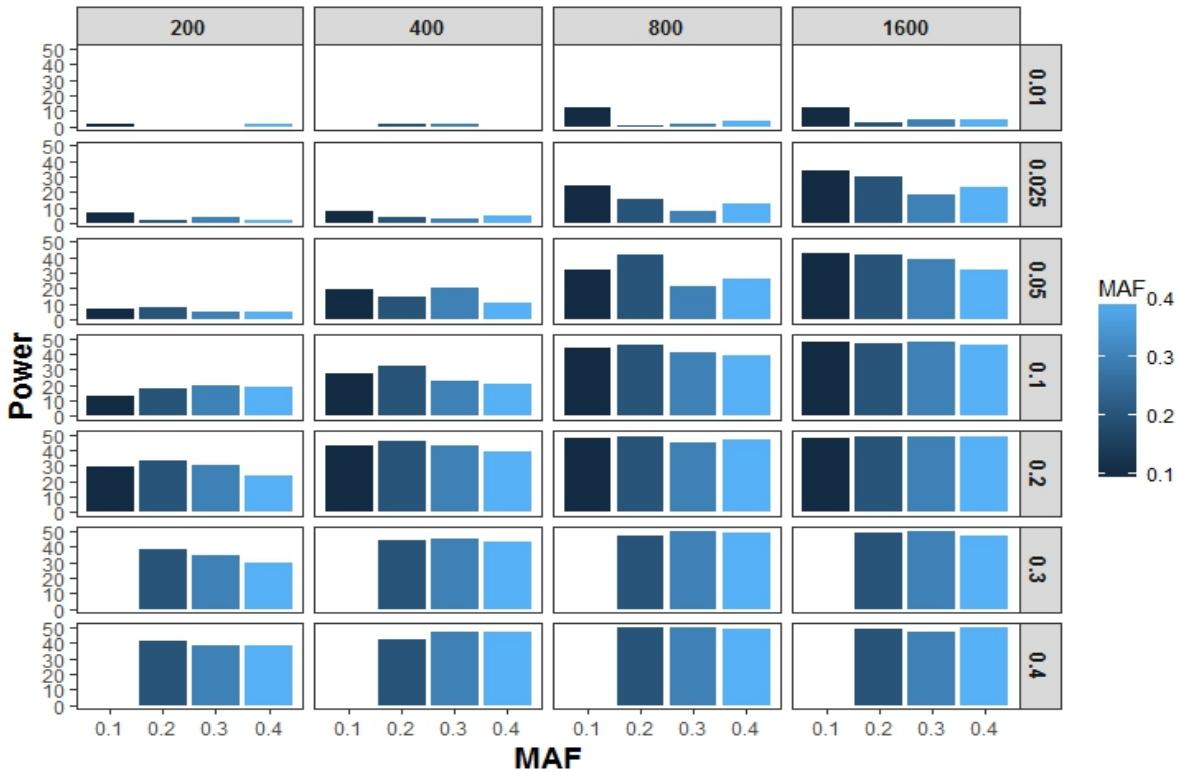


Figure.7.7 Power of DNN-RF for ratio 1:2 for various minor allele frequencies, heritability, and sample size.

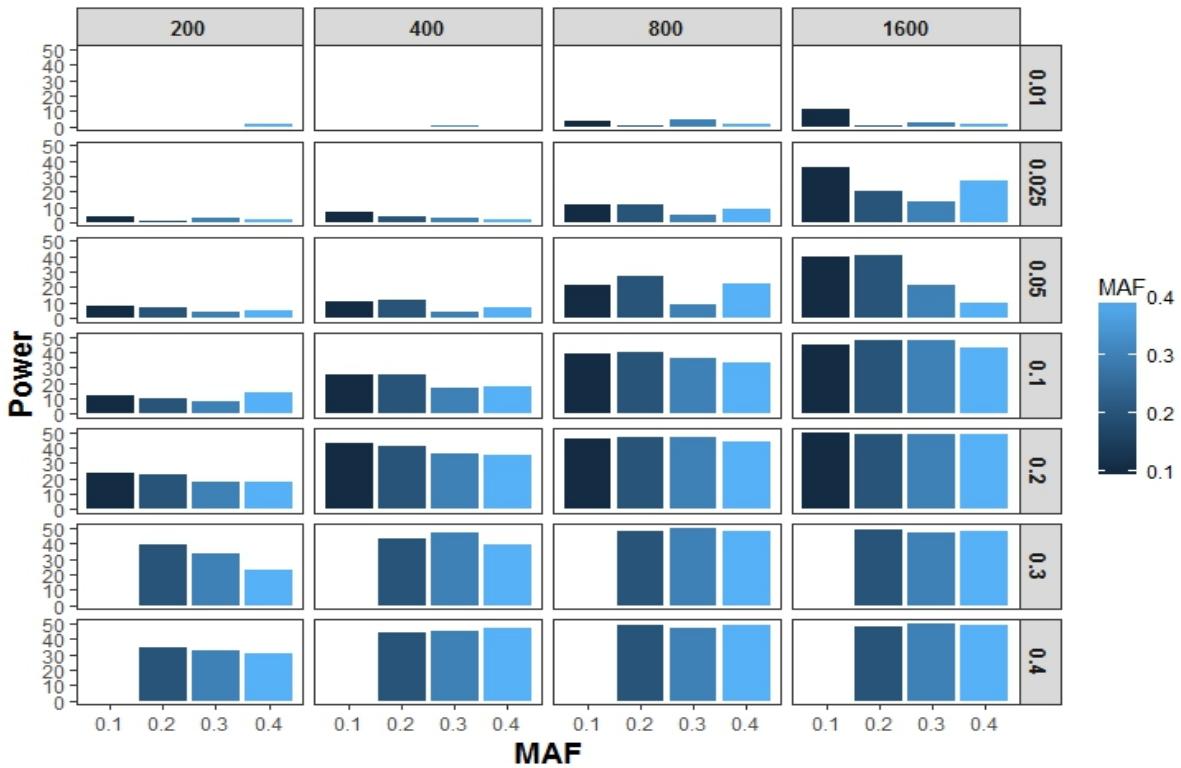


Figure.7.8 Power of DNN-RF for ratio 1:4 for various minor allele frequencies, heritability, and sample size.

**Table 7.5:** Summary of power of DNN-RF to detect known two-locus SNP interactions.

Sample Size	Ratio	Power (%)	Classification Accuracy
200	1:1	35.1	74.36
	1:2	34.43	63.03
	1:4	26.87	57.63
400	1:1	52.56	74.873
	1:2	47.77	60.77
	1:4	42.17	55.23
800	1:1	67.11	75.08
	1:2	65.5	61.25
	1:4	57.03	55.38
1600	1:1	78.11	75.27
	1:2	74	63.55
	1:4	69.27	55.8

Power and classification accuracy of the method increases by increasing the sample size, and they are affected by case-control ratios. On an average, it is observed from the number of experiments that power of the method is high for higher values of MAF and

heritability. Even though, the proposed method performed better by increasing the sample size, still further investigations required to improve the performance of the method at low values of heritability.

## 7.6.2 Real world data application

Chronic dialysis patients' dataset (presented in Section 6.1.5) is applied on the proposed DNN-RF method to evaluate performance of the models for detecting two-locus SNP interactions responsible for disease manifestation. The method is built and trained in R [326]. As per preliminary evaluations, the method is evaluated by changing the activation functions (such as, rectifier, tanh, softmax, and maxout along with dropouts) [45] and optimising hyper-parameters (using grid and random grid search) [47]. Number of experiments is performed for evaluating all the possible combinations of hyper-parameters (such as, hidden layers, epochs, input drop ratio, epsilon, momentum, learning rate, annealing rate, L1 and L2 penalties'). The best model with optimal hyper-parameters is chosen to maximise the model's accuracy. It is observed that the model convergence and synchronization is decreased as the sample size increases. Computational performance is affected if training samples are too low. Figure 7.9 illustrates the performance of metrics of each model with respect to accuracy, precision, auc, logloss, mse, and classification error for both training and validation. The two-locus interactions between SNP 21 (2) – SNP 28 (2) is identified by the best model with the maximum accuracy of 78.1337. Figure 7.10 plots top 30 two-locus SNP interactions responsible for chronic dialysis.

Figure 7.11 plots the accuracy graph of DNN-RF along with previous methods from the literature. The proposed model is compared with some of the commonly used approaches in machine learning (such as, LR, MDR, RF, GBM, and DNN). The study is evaluated on chronic dialysis patients' data. The prediction accuracy of DNN is 77.99. A best two-way SNP interaction identified by the model is SNP 22 – SNP 26. MDR identified interaction between SNP 40 – SNP 56 with the accuracy of 56.1 along with highest CVC 10\10. LR detected SNP21 – SNP52 interaction along with the accuracy of 73.82. Similarly, the best two-locus interaction identified by RF is SNP 9 – SNP 64 with the accuracy of 77.03. Finally, the GBM identifies two-locus SNP interaction (SNP 55 –

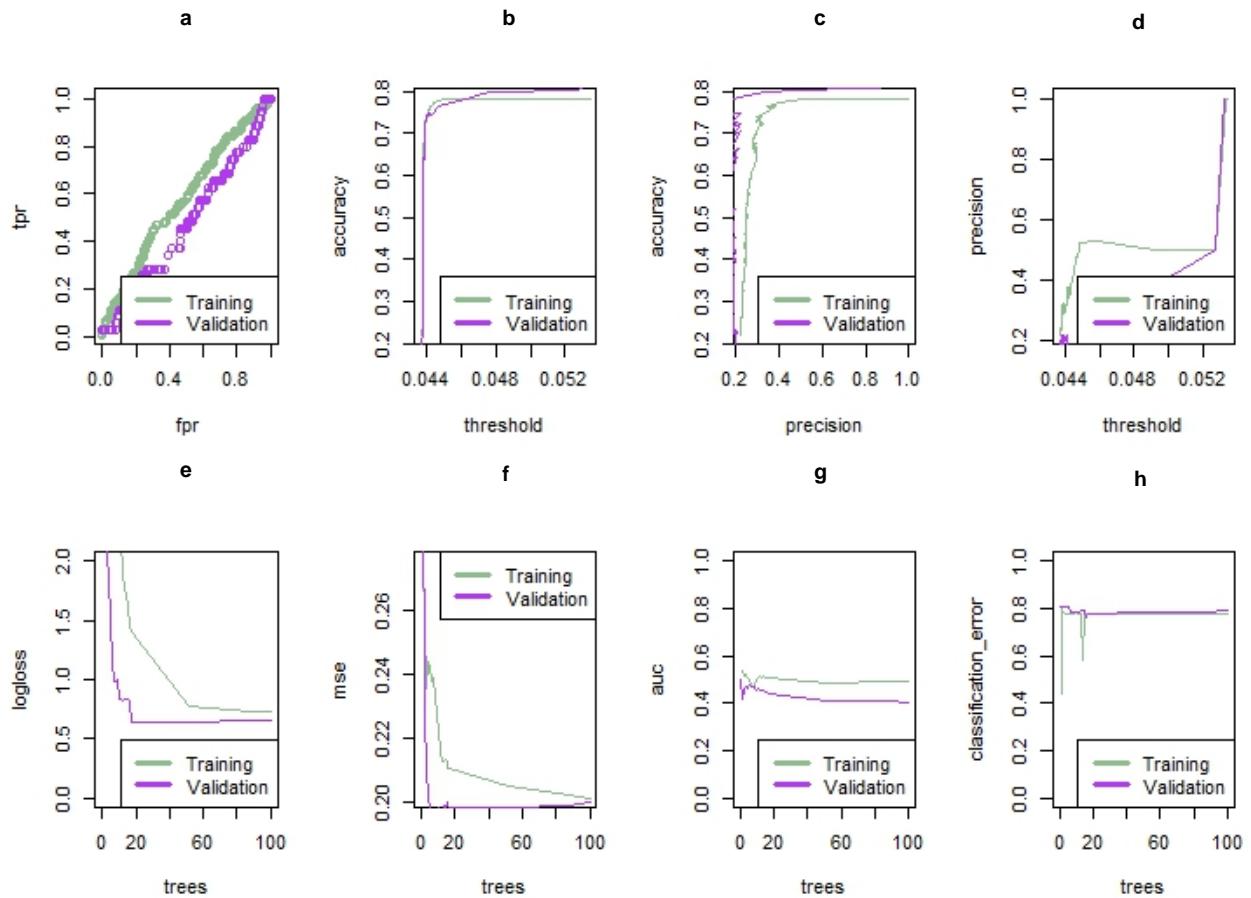


Figure 7.9 Performance analysis of DNN-RF hybrid method for two-locus SNP interactions. **(a)** True positive rate (tpr) vs false positive rate (fpr) for both training and validation, **(b)** accuracy plot at which maximum accuracy is obtained for a threshold value, **(c)** accuracy vs precision plot during training and validation, **(d)** Precision plot at which maximum accuracy is obtained for a threshold value, **(e)** scoring history of logloss at n trees, **(f)** scoring history of mse at n trees, **(g)** scoring history of area under curve (auc) for training and validation in n trees, and **(h)** scoring history of classification error.

SNP64) with the highest cross-validation accuracy (77.86) when compared with other approaches from the literature.

It is observed that different methods detect different two-locus interactions. These observations urged caution in interpretation of the results obtained from different methods. Despite accuracy being cited as one of the criteria to evaluate the performance of different methods, W-test is performed to measure pairwise SNP interactions [346]. W-test is a model free statistical test that follows scaled Chi-squared distribution as a function of two parameters (scalar  $h$  and degree of freedom  $f$ ) using bootstrapping. The  $h$  and  $f$  values are adjusted for distributional bias in the presence of population stratification.

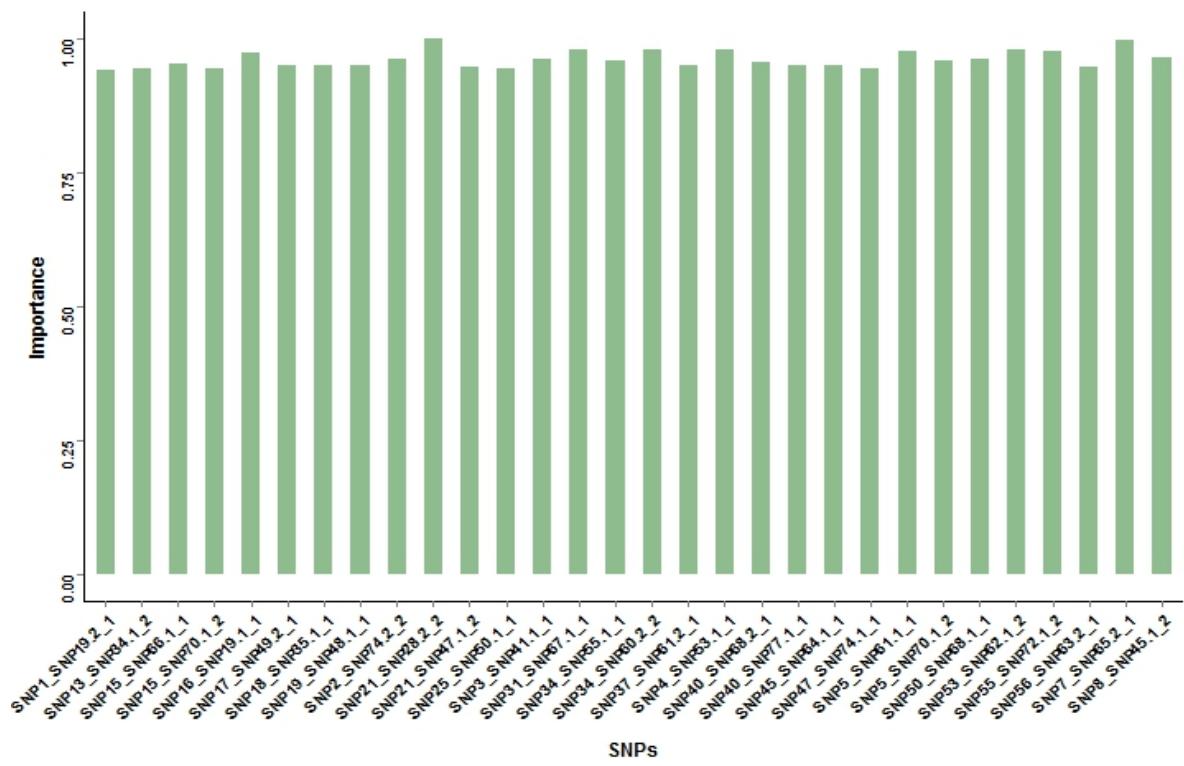


Figure.7.10 Top 30 two-way SNP interactions identified by DNN-RF model on Chronic dialysis patients' dataset.

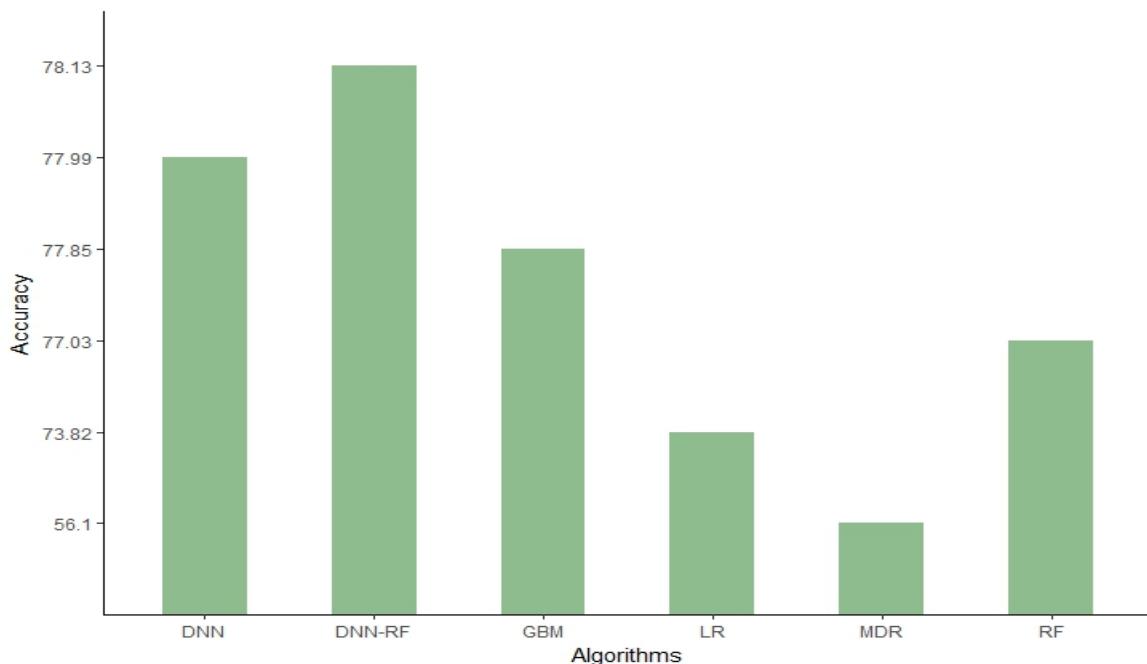


Figure.7.11 Accuracy of hybrid DNN-RF model compared with some of the previous algorithms such as, multilayered feedforward neural networks (DNN), gradient boosted machines (GBM), multifactor dimensionality reduction (MDR), logistic regression (LR), and random forest (RF).

Table 7.6 shows w-value, and p-value of the top 5 predicted SNP interactions for different methods. It is observed that p-values of all the top five pairwise SNPs of the proposed method showed statistical significance of p-value < 0.05. Previous studies show that SNP 64, SNP65, SNP 21, and SNP 34 have higher incidence in D-loop of CRS of cases rather than controls [337]. Hence, there is no reason for suspecting the findings of the proposed method. Four pairwise SNP interactions predicted by DNN and RF are statistically significant ( $p<0.05$ ). Only one p-value of RF (SNP21\_SNP56) showed deviation from statistical significance. This could be due to false positive, where SNP 56 is independently associated with the high risk of chronic dialysis. Similarly, three pairwise interactions predicted by the GBM showed statistically insignificant results due to false positives. This may be caused due to marker dependencies rather than real interaction effects.

In MDR, only two pairwise interactions are statistically significant. This may be due to lower cross validation consistency when the MAFs are low. This may lead to lower p-values. It may also be due to extreme sensitivity to the choice of random seed number. Hence, results are interpreted carefully based on the optimal choice of random seed. In LR, only one pairwise interaction is statistically significant with p-value <0.05. These unexpected results may be due to linear relationship with the interaction term. Many useful non-linear relationships may be missed in co-dominant genetic models.

**Table 7.6:** W-test for predicted two-locus SNP interactions.

Methods	Top 5 SNP Interactions	W values	p-value
DNN-RF	SNP21_SNP28	8.846399	0.03376429
	SNP7_SNP65	7.585945	0.02797846
	SNP31_SNP67	10.17918	0.01848273
	SNP34_SNP60	12.39451	0.00669231
	SNP16_SNP19	12.52843	0.00629073
	SNP4_SNP63	14.5787	0.00242609
DNN	SNP22_SNP26	11.43058	0.00424467
	SNP1_SNP37	8.100954	0.02175054
	SNP8_SNP24	11.08434	0.01222723
	SNP1_SNP3	6.072052	0.05852792
	SNP23_SNP70	10.626	0.01507808
RF	SNP9_SNP64	7.386538	0.03084088
	SNP21_SNP56	0.2957758	0.9651625
	SNP10_SNP64	7.021056	0.03686393

	SNP64_SNP77	16.12623	0.0011751
	SNP54_SNP64	9.687092	0.00999816
GBM	SNP55_SNP64	9.687092	0.00999816
	SNP8_SNP31	3.42325	0.3449625
	SNP3_SNP64	14.71695	0.00227437
	SNP29_SNP35	4.514217	0.1245242
	SNP21_SNP52	1.107926	0.7886552
MDR	SNP40_SNP56	2.576055	0.4777909
	SNP21_SNP64	9.315519	0.02733535
	SNP47_SNP56	6.009799	0.1178916
	SNP45_SNP56	1.854882	0.6193778
	SNP56_SNP64	9.264284	0.02797464
LR	SNP21_SNP52	1.107926	0.7886552
	SNP35_SNP56	4.407371	0.231819
	SNP8_SNP39	4.794594	0.1974254
	SNP55_SNP64	12.82208	0.00213938
	SNP21_SNP40	2.908923	0.4213296

However, to determine the best method, computer simulations are generally performed in the literature with known interactions [36, 108]. Simulation studies identifies that different methods perform well under different true models with different scenarios. Further, different methods often assume different definitions for interaction effects [347]. All these factors may also determine the prediction of the methods.

Further preliminary studies are performed on the chronic dialysis patients' dataset to observe the performance of the proposed method on higher-order interactions. Figure 7.12 illustrates the top 30 three-way interactions identified by the proposed method. We could observe that the proposed method can be easily extended for higher-order interactions. However, the execution time drastically increased when compared with detection for two-way interactions. This observation leads us to urge caution to investigate some of the filtering and parallel computational approaches implemented into the method.

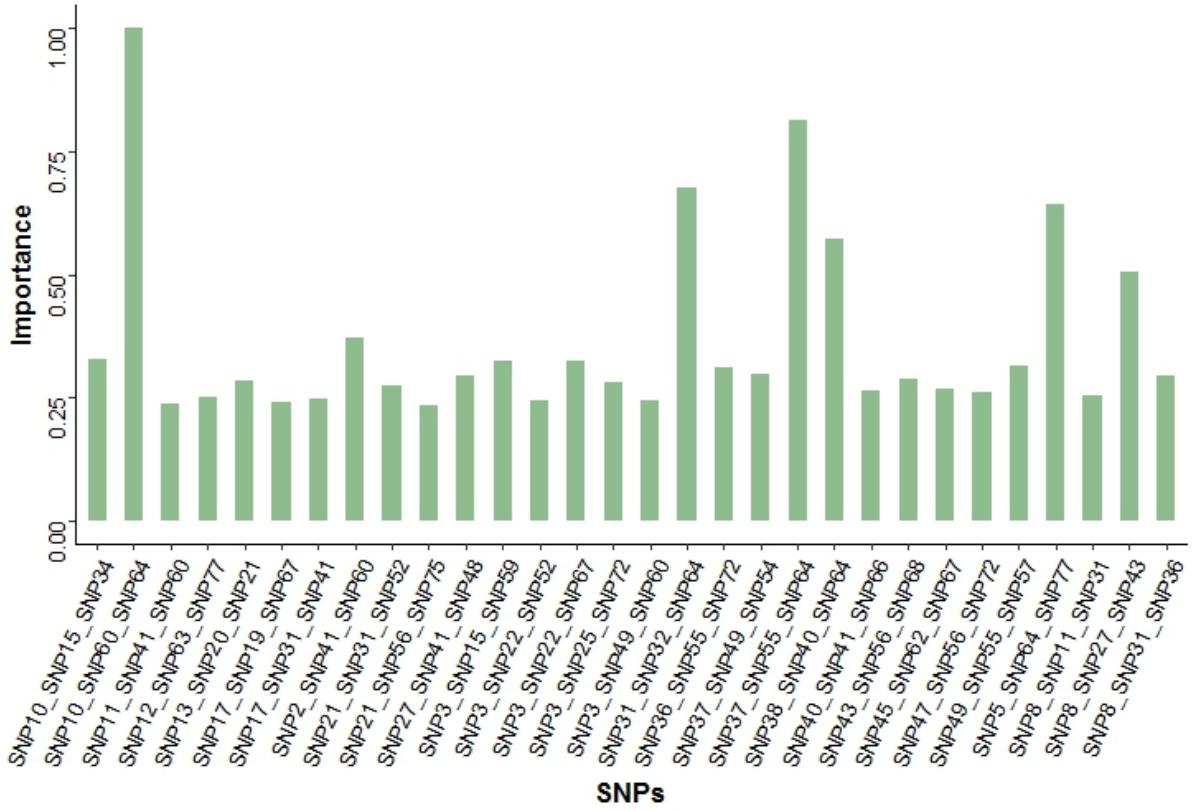


Figure.7.12 Top 30 three-way SNP interactions identified by DNN-RF model on Chronic dialysis patients' dataset.

## 7.7 Discussions

From the simulated studies, it is observed that, the power of the proposed method in presence of GE and MS, and their combined effects are 90-100% for all six epistasis models. In the presence of PC (variation of a phenotype that is caused due to environmental factors, but it resembles as due to inheritance of genetic factors), and their combined effect with MS and GE, the power of the method also performed equally well with 90-100% for model 1 ( $p=0.5$ ,  $q=0.5$ ,  $H=0.053$ ) and model 2 ( $p=0.5$ ,  $q=0.5$ ,  $H=0.051$ ). These are due to the simplest nature of the models, which do not exhibit any independent main effects [36]. That is, Model 1 ( $p=0.5$ ,  $q=0.5$ ,  $H=0.053$ ) has four high risk cells with the function of XOR, which cannot be separated linearly. Model 2 ( $p=0.5$ ,  $q=0.5$ ,  $H=0.051$ ) has only three high risk cells in the presence of aaBB, AaBb, and AAAb, when compared with the other models with four high risk cells. The power of method in the presence of PC can be improved by including environmental factors rather than any methodological changes to the DNN-RF hybrid method [36]. Encompassing the

environmental factors into the analysis is another challenge to be considered. AntEpiseeker [92], BOOST [22], and SNPRuler [191] are some of the approaches that are robust to the phenocopy.

Genetic heterogeneity can be either allelic or locus based in which, multiple genes are associated for causing a phenotype. Locus heterogeneity occurs due to the SNPs at different locus in contrast with allele heterogeneity, which occurs due to SNPs in the same locus. It is observed that the power is between 24-56% in the presence of GH, and their combined effect with MS, GE, and PC for all six models. However, the power of the method reduced drastically low between 0-20% for model 3 – 6 in the presence of combined effect of GH-PC. This is due to complex nature of the models along with reduced levels in the frequencies of rare allele  $p = 0.25$  and  $p = 0.1$  [36]. These observations suggests us the space in which more sophisticated algorithms can be integrated to improve the power of the method to detect SNP interactions in the presence of GH, and their combined effect with PC. It is believed that GH plays very important role in the insight of complex diseases [348]. However, the power of GH, and their combined effects are comparatively high when compared with other approaches such as MDR, and DNN. This is due to the reason of incorporating RF into analysis to improve overall predictive accuracy of the method.

## 7.8 Chapter Summary

In this chapter, a deep neural network was unified with a random forest by forming a hybrid method, for achieving reliable detection of multi-locus interactions between SNPs. The proposed method was evaluated on various simulated scenarios in the absence of main effects. Further, the performance of the method in the presence of noise due to MS, GE, GH, and PC, and their combined effects were evaluated. The power of the method was compared with the previously proposed extended deep learning method in the presence and absence of noise. On an average, the power of the proposed hybrid method was much higher than the previous methods for all simulated scenarios in the presence of noise. However, the power of the hybrid method in the presence of PC, and GH, and their combined effects is still low due to the complex nature of the models. Finally, these findings were confirmed on a chronical dialysis patient's data. It was observed that the two-locus interaction in the mitochondrial D-loop has the highest risk

for the disease manifestation. These findings will pave the way for further exploration of improving the performance of the method.

The next chapter presents a summary of the contributions of this thesis and discusses how the research objectives were achieved. Furthermore, the chapter will discuss several ideas for the future work based on the experimental observations in this chapter and the previous chapters.

# **Chapter 8**

## **Conclusions and Future works**

With the growing amount of high-throughput genotyping data, many researchers have made great efforts to examine the interaction effects between the multi-locus SNPs for discriminating the status of complex diseases. Detecting these interactions in high-dimensional genome is difficult, due to the growing number of genetic variants in human genetics. A number of efficient and less expensive methodologies and computational techniques have been incorporated by the researchers in the current literature. However, the current conventional techniques are still left with several caveats. It was evident from the limitations of the current methodologies that they still require further developments for a better understanding of interaction studies. Hence, the main focus of this thesis was to propose the methods to detect higher-order interactions in GWIS based on associative classification and deep learning methods.

Many researchers have shown that AC is more accurate than traditional classifiers. The rules generated in AC can be stored and can provide reasoning to the classification. AC is also suitable for both categorical and discrete data. Mapping SNP-SNP interactions to disease can be improved by integrating association rules and classification. Deep learning is a new breed of machine learning that elucidates the hidden structure of the raw data by transforming it into multiple high levels of abstractions, using the power of parallel and distributed computing. It promises empirical success in a number of applications including bioinformatics to provide insights into biological complexities. The deep learning in the multi-locus interaction studies is yet to begin to meet its potential achievements. These explorations have motivated the need to address the series of problems in GWIS that focus on improving the detection of real causal interactions responsible for disease manifestations. In this chapter, the work that has been carried out in this thesis is summarised in Section 8.1. The chapter also introduces some of the possible future avenues for improving the proposed methods in Section 8.2.

## **8.1 Summary of Contributions**

The main objective of this research was to implement methods to detect higher-order SNP interactions in high-dimensional genome data. Chapter 1 discussed several research objectives that would be explored in this thesis. Chapter 2 provided the background for formulating these research objectives, and explored many questions to be answered even in well-established methods. These objectives are summarised below with respect to the contributions presented in the previous chapters.

### **Objective 1: Develop methods for search of two-locus SNP interactions**

The main goal of this research was to develop efficient methods for detecting true causal subset of SNP interactions in GWAS. In Chapter 3, a rule based approach was implemented to detect interaction effects more accurately than traditional approaches. This method has shown some promising ability of searching for two-locus interactions in imbalanced datasets. However, a huge number of rules were generated as the SNP combinations increase. This was computationally demanding. Hence, a MDR based AC was proposed in Chapter 4 by reducing the dimensionality of the data. This method has shown improved accuracy in both balanced and imbalanced datasets. The major challenge for the MDRAC was that the power was reduced in the presence of noise. These observations provided new clues towards exploration of deep learning to overcome this inherent problem. Chapter 5 trained a deep neural network, and showed potential results compared with other two methods. The power of this method in the presence of noise was improved by integrating RF with DNN (DNN-RF) in Chapter 7.

### **Objective 2: Extend and improve the performance of the proposed methods**

Rule based approach and MDRAC were extended for identifying three-locus to six-locus interactions in Chapter 3 and Chapter 4 respectively. These methods were successfully demonstrated over both balanced and imbalanced case-control datasets. In Chapter 6, the deep learning method was extended and evaluated for one-locus to ten-locus interactions. The combined effect of higher-order interactions was also observed for the proposed DNN method. The method was evaluated for unsupervised tasks, such as; high-dimensional data was reduced to low-dimensional data by incorporating PCA, discovered anomalies in the reduced data representation using deep auto-encoder.

Sensitivity analysis was further conducted to observe the behaviour of the network for change in input parameters. The proposed method showed promising ability for searching important higher-order interactions. However, further improvements in optimisation techniques, and learning algorithms improved the accuracy of the models.

### **Objective 3: Evaluating the proposed methods on both simulated and real data**

The four methods proposed were evaluated for various simulated scenarios and real data application studies in Chapters 3 to 7. The simulated datasets were generated by varying minor allele frequency, heritability with various penetrance functions, various case-control ratios, and sample size. These simulated datasets were generated in the presence and absence of noise using GAMETES tool. The methods were studied in detail in the presence of noise due to GE, MS, GH, PC and their combined effects. Although the power of the models to search for subset of interesting SNPs was high in the presence of GE, and MS, it drastically dropped in the presence of GH, PC and their combined effects. Further experimental findings were validated on four real datasets (sporadic breast cancer, hypertension, chronic kidney dialysis patients' data, and data obtained from whole genome). Top 20 highly ranked higher-order SNP interactions were identified by the proposed methods in Chapter 5 to 7.

### **Objective 4: Comparing the performance of the proposed methods with the previous approaches**

The performances of the proposed methods were compared with some of the conventional methods for both simulated and real datasets. It was observed that the extended deep learning method potentially performed well compared with other previous approaches. The best models had the highest prediction accuracy with low classification errors. However, the power of the previous methods was much reduced in the presence of noise and their combined effects, when compared with the power of DNN-RF. This was due to the incorporation of RF into analysis to improve the overall predictive performance of the models. The method has demonstrated better accuracy in identifying subsets of important SNPs by reducing false associations. Despite accuracy being cited as one of the criteria to evaluate the performance of the methods, W-test is also performed to measure the statistical significance of the interactions identified by the previous approaches.

## 8.2 Limitations and Future work

This research can be further improved by addressing some of the potential limitations, where further research and extensions can be incorporated. These suggestions are partially presented in the published papers [49, 51].

### 8.2.1 Population Stratification

An important confounding factor in the case-control datasets was population stratification, which lead to spurious associations between SNPs. The studies presented in this thesis were restricted to homogenous populations. Improving the ability of the proposed methods to handle population stratification could be a major aspect to be considered. In GWAS, a number of approaches have been successfully implemented to detect main effects by controlling the population stratification [349]. However, only a few methods are available in interaction analysis to handle population stratification, such as, principal component analysis based method [350], MDR in structured population [351], and w-test for pairwise interactions [346]. Even though our preliminary results remain a promising option to adjust for population stratification, test showed sensitivity to some of the main effects included in the models. Romero [352] showed how to experimentally handle population stratification by proposing diet networks using deep learning. It reduced the number of free parameters by implementing a multi-tasking architecture that can handle very high-dimensional input. It showed better generalization with respect to misclassification errors. This investigation can be implementation into the proposed methods to handle population stratification. This can improve the understanding of etiology of a disease at the population level.

### 8.2.2 Power calculations

Another major challenge of interaction analysis of the proposed methods was power calculations. Power calculations in the interaction analysis of the methods were performed on simulated studies with different data parameters (such as, sample size, allele frequency, and heritability), and tested experimentally to show the power of the method to detect known interactions. In existing literature, simulated models are

generated to fit the specific needs of the method by noting the power to detect higher-order interaction effects with different models/data simulation settings [36, 78, 108, 353]. Urbanowicz described a simulation method, and a metric to estimate relative model detection difficulty that strongly correlated with detection power [221, 345]. Further, they presented a case for a simulation study design, which takes model architecture into account by identifying the point where methods may or may not be empowered to detect the underlying interactions. Considering these recommendations from the existing literature, simulated studies can be performed to evaluate the power of the proposed methods in various simulated settings.

### 8.2.3 Training data

It was observed that huge training data was required due to optimisation of higher number of weights and biases of the deep learning methods. It was also observed that the data with imbalanced class becomes difficult to train the DNN as performance metrics showed bias towards the majority class. Hence, the imbalanced datasets can be analyzed using oversampling or under sampling or by adjusting the threshold values of the models. Over sampling randomly re-samples the underrepresented class of samples until the number of cases and controls are equal. Under sampling randomly removes samples from an overrepresented class until the number of cases and controls are equal. Even though this technique was used in the literature [108], there may be false associations due to the samples that are oversampled or under sampled. This could also provide a false sense of high power [301]. Further, the performance of the proposed deep learning method was not expected to perform well if the input is highly sparse and irregular. Hence, several normalized optimization techniques, and implementation of stacked autoencoders to the proposed DNN model can be explored.

### 8.2.4 Activation Function

The activation function  $tanh$  used in the deep learning method maps the data to the [-1, 1] domain resulting in minimal learning. This can be improved by using learnable activation functions that approximates to underlying data distribution. Implementing W-test to measure pairwise interactions into the model can reduce the distributional bias occur due to sparse data and small sample size [346]. W-test copes well in sparse

data by using data dependent bootstrap estimation of  $h$  and  $f$  parameters. Further investigations of incorporating the w-test estimations into the methods will be conducted for higher-order interaction testing.

### **8.2.5 Huge network parameters**

Training DNNs for the multi-locus genome data became tedious and challenging due to huge number of network parameters. Convolutional neural network (CNN) can be explored for the current problem (preliminary studies are explored [51]). CNNs are inspired by visual cortex of the human brain. In the visual cortex, there are simple neurons that respond to primitive patterns in the visual field, and complex neurons that respond to large intricate forms. CNNs leverages the idea of sparse interactions, parameter sharing and equivariant representations to improve machine learning [321]. They uses convolution, a mathematical linear operation, instead of matrix multiplication at least in one of their hidden layers. The main features of CNN are to share the weights and extract useful features with its trained weights. Hence, CNNs are considered to be less complex and uses less memory compared with DNNs. These features of CNN can suggest incorporating it into the method for improving learning efficiently.

### **8.2.6 Presence of Noise**

The power of DNN-RF is reduced in the presence of PC and GH, and their combined effects. GH played important role in the insight of complex diseases. These observations suggest that more sophisticated algorithms can be integrated into the method to improve the power of the method in the presence of GH, and their combined effect with PC. The power of the method in the presence of PC can be improved by including environmental factors rather than any methodological changes to the DNN-RF hybrid method [36]. Encompassing the environmental factors into the analysis is another challenge to be considered.

### **8.2.7 Other future directions**

Unexpectedly, the current variable measures available in RF, fails to capture the interactions in high-dimensional settings. Hence, further research strategies can be explored for accurate variable selection measures in DNN-RF. Choosing the best model

from large combinations of hyper-parameters becomes tedious when the parameters of the DNN-RF are increased, and reproducing these results also becomes more complex. Hence, other optimising search algorithms can also be explored to maximise the predictive accuracy of the method.

Further advances in computing power and optimising algorithms will play a prominent role in analysing the interactions among the SNPs. The etiology of complex diseases involves genetic and environmental factors, and their interaction effects. Hence, the contribution of environmental factors over the interaction effects should not be underestimated. It is believed that exposing these interactions in detail will provide new insights into the genetic architecture of a number of complex diseases. These findings will pave the way for further exploration of improving the performance of the method by implementing the multimodal, and the accelerating deep learning techniques into the architecture. Although, the suggested approaches are complementary to the current deep learning methods, continuous improvements with more sophisticated algorithms can be applied to modelling high-dimensional SNP interactions.

# Bibliography

- [1] I. H. Consortium, "A second generation human haplotype map of over 3.1 million SNPs," *Nature*, vol. 449, p. 851, 2007.
- [2] L. Padyukov, *Between the Lines of Genetic Code: Genetic Interactions in Understanding Disease and Complex Phenotypes*: Academic Press, 2013.
- [3] W.-H. Wei, G. Hemani, and C. S. Haley, "Detecting epistasis in human complex traits," *Nature Reviews Genetics*, 2014.
- [4] E. S. Gusareva, M. M. Carrasquillo, C. Bellenguez, E. Cuyvers, S. Colon, N. R. Graff-Radford, R. C. Petersen, D. W. Dickson, J. M. M. John, and K. Bessonov, "Genome-wide association interaction analysis for Alzheimer's disease," *Neurobiology of aging*, vol. 35, pp. 2436-2443, 2014.
- [5] J. H. Moore and S. M. Williams, "Traversing the conceptual divide between biological and statistical epistasis: systems biology and a more modern synthesis," *Bioessays*, vol. 27, pp. 637-646, 2005.
- [6] H. J. Cordell, "Detecting gene–gene interactions that underlie human diseases," *Nature Reviews Genetics*, vol. 10, pp. 392-404, 2009.
- [7] C. L. Koo, M. J. Liew, M. S. Mohamad, and A. H. Mohamed Salleh, "A review for detecting gene–gene interactions using machine learning methods in genetic epidemiology," *BioMed research international*, vol. 2013, 2013.
- [8] C. K. I. Ruczinski, M. L. LeBlanc, and L. Hsu, "Sequence analysis using logic regression," *Genetic epidemiology*, vol. 21, pp. S626-S631, 2001.
- [9] M. Y. Park and T. Hastie, "Penalized logistic regression for detecting gene interactions," *Biostatistics*, vol. 9, pp. 30-50, 2008.
- [10] W. Y. Loh, "Classification and regression trees," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 1, pp. 14-23, 2011.
- [11] N. R. Cook, R. Y. Zee, and P. M. Ridker, "Tree and spline based association analysis of gene–gene interaction models for ischemic stroke," *Statistics in medicine*, vol. 23, pp. 1439-1453, 2004.
- [12] J. Millstein, D. V. Conti, F. D. Gilliland, and W. J. Gauderman, "A testing framework for identifying susceptibility genes in the presence of epistasis," *The American Journal of Human Genetics*, vol. 78, pp. 15-27, 2006.
- [13] N. Tahri-Daizadeh, D.-A. Tregouet, V. Nicaud, N. Manuel, F. Cambien, and L. Tiret, "Automated detection of informative combined effects in genetic association studies of complex traits," *Genome Research*, vol. 13, pp. 1952-1960, 2003.
- [14] M. Nelson, S. Kardia, R. Ferrell, and C. Sing, "A combinatorial partitioning method to identify multilocus genotypic partitions that predict quantitative trait variation," *Genome Research*, vol. 11, pp. 458-470, 2001.
- [15] R. Culverhouse, T. Klein, and W. Shannon, "Detecting epistatic interactions contributing to quantitative traits," *Genetic epidemiology*, vol. 27, pp. 141-152, 2004.
- [16] A. Wille, J. Hoh, and J. Ott, "Sum statistics for the joint detection of multiple disease loci in case-control association studies with SNP markers," *Genetic epidemiology*, vol. 25, pp. 350-359, 2003.

- [17] M. D. Ritchie, L. W. Hahn, N. Roodi, L. R. Bailey, W. D. Dupont, F. F. Parl, and J. H. Moore, "Multifactor-dimensionality reduction reveals high-order interactions among estrogen-metabolism genes in sporadic breast cancer," *The American Journal of Human Genetics*, vol. 69, pp. 138-147, 2001.
- [18] L. Kaufman and P. J. Rousseeuw, *Finding groups in data: an introduction to cluster analysis* vol. 344: John Wiley & Sons, 2009.
- [19] R. Upstill-Goddard, D. Eccles, J. Fliege, and A. Collins, "Machine learning approaches for the discovery of gene–gene interactions in disease data," *Briefings in bioinformatics*, vol. 14, pp. 251-260, 2013.
- [20] T. Kohonen, *Self-organizing maps* vol. 30: Springer, 2001.
- [21] X. Zhang, S. Huang, F. Zou, and W. Wang, "TEAM: efficient two-locus epistasis tests in human genome-wide association study," *Bioinformatics*, vol. 26, pp. i217-i227, 2010.
- [22] X. Wan, C. Yang, Q. Yang, H. Xue, X. Fan, N. L. Tang, and W. Yu, "BOOST: A fast approach to detecting gene-gene interactions in genome-wide case-control studies," *The American Journal of Human Genetics*, vol. 87, pp. 325-340, 2010.
- [23] L. S. Yung, C. Yang, X. Wan, and W. Yu, "GBOOST: a GPU-based tool for detecting gene–gene interactions in genome-wide case control studies," *Bioinformatics*, vol. 27, pp. 1309-1310, 2011.
- [24] C. C. Chen, H. Schwender, J. Keith, R. Nunkesser, K. Mengersen, and P. Macrossan, "Methods for identifying SNP interactions: a review on variations of Logic Regression, Random Forest and Bayesian logistic regression," *Computational Biology and Bioinformatics, IEEE/ACM Transactions on*, vol. 8, pp. 1580-1591, 2011.
- [25] R. Jiang, W. Tang, X. Wu, and W. Fu, "A random forest approach to the detection of epistatic interactions in case-control studies," *BMC bioinformatics*, vol. 10, p. S65, 2009.
- [26] M. Yoshida and A. Koike, "SNPInterForest: a new method for detecting epistatic interactions," *BMC bioinformatics*, vol. 12, p. 469, 2011.
- [27] D. F. Schwarz, I. R. König, and A. Ziegler, "On safari to Random Jungle: a fast implementation of Random Forests for high-dimensional data," *Bioinformatics*, vol. 26, pp. 1752-1758, 2010.
- [28] X. Chen, C.-T. Liu, M. Zhang, and H. Zhang, "A forest-based approach to identifying gene and gene–gene interactions," *Proceedings of the National Academy of Sciences*, vol. 104, pp. 19199-19203, 2007.
- [29] Y. Zhang and J. S. Liu, "Bayesian inference of epistatic interactions in case-control studies," *Nature genetics*, vol. 39, pp. 1167-1173, 2007.
- [30] B. Han, X.-w. Chen, and Z. Talebizadeh, "FEPI-MB: identifying SNPs-disease association using a Markov Blanket-based approach," *BMC bioinformatics*, vol. 12, p. S3, 2011.
- [31] B. Han and X.-w. Chen, "bNEAT: a Bayesian network method for detecting epistatic interactions in genome-wide association studies," *BMC genomics*, vol. 12, p. S9, 2011.
- [32] X. Wan, C. Yang, Q. Yang, H. Xue, N. L. Tang, and W. Yu, "MegaSNPHunter: a learning approach to detect disease predisposition SNPs and high level interactions in genome wide association study," *BMC bioinformatics*, vol. 10, p. 13, 2009.
- [33] M. S. Cunningham, M. S. Koref, B. M. Mayosi, J. Burn, and B. Keavney, "Chromosome 9p21 SNPs associated with multiple disease phenotypes correlate with ANRIL expression," *PLoS genetics*, vol. 6, p. e1000899, 2010.

- [34] P. Yang, J. W. Ho, A. Y. Zomaya, and B. B. Zhou, "A genetic ensemble approach for gene-gene interaction identification," *BMC bioinformatics*, vol. 11, p. 524, 2010.
- [35] K. Van Steen, "Travelling the world of gene–gene interactions," *Briefings in bioinformatics*, vol. 13, pp. 1-19, 2012.
- [36] M. D. Ritchie, L. W. Hahn, and J. H. Moore, "Power of multifactor dimensionality reduction for detecting gene-gene interactions in the presence of genotyping error, missing data, phenocopy, and genetic heterogeneity," *Genetic epidemiology*, vol. 24, pp. 150-157, 2003.
- [37] T. Cattaert, M. L. Calle, S. M. Dudek, J. M. Mahachie John, F. Van Lishout, V. Urrea, M. D. Ritchie, and K. Van Steen, "Model-Based Multifactor Dimensionality Reduction for detecting epistasis in case-control data in the presence of noise," *Annals of human genetics*, vol. 75, pp. 78-89, 2011.
- [38] S. Uppu, A. Krishna, and R. P. Gopalan, "An Associative Classification Based Approach for Detecting SNP-SNP Interactions in High Dimensional Genome," in *Bioinformatics and Bioengineering (BIBE), 2014 IEEE International Conference on*, 2014, pp. 329-333.
- [39] S. Uppu, A. Krishna, and R. Gopalan, "Detecting SNP interactions in balanced and imbalanced datasets using associative classification," *Australian Journal of Intelligent Information Processing Systems*, vol. 14, pp. 7-18, 2014.
- [40] S. Uppu, A. Krishna, and R. P. Gopalan, "Rule-based analysis for detecting epistasis using associative classification mining," *Network Modeling Analysis in Health Informatics and Bioinformatics*, vol. 4, pp. 1-19, 2015.
- [41] S. Uppu, A. Krishna, and R. P. Gopalan, "A Multifactor Dimensionality Reduction Based Associative Classification for Detecting SNP Interactions," in *Neural Information Processing*, 2015, pp. 328-336.
- [42] S. Uppu, A. Krishna, and R. Gopalan, "Combining associative classification with multifactor dimensionality reduction for predicting higher-order SNP interactions in case-control studies," *International Journal of Computational Intelligence in Bioinformatics and Systems Biology*, accepted on 22/10/2017.
- [43] S. Uppu and A. Krishna, "Evaluation of associative classification-based multifactor dimensionality reduction in the presence of noise," *Network Modeling Analysis in Health Informatics and Bioinformatics*, vol. 5, pp. 1-9, 2016.
- [44] S. Uppu, A. Krishna, and R. P. Gopalan, "Towards Deep Learning in genome-Wide Association Interaction studies," in *PACIS*, 2016, p. 20.
- [45] S. Uppu, A. Krishna, and R. P. Gopalan, "A Deep Learning Approach to Detect SNP Interactions," *JSW*, vol. 11, pp. 965-975, 2016.
- [46] S. Uppu and A. Krishna, "Improving strategy for discovering interacting genetic variants in association studies," in *International Conference on Neural Information Processing*, 2016, pp. 461-469.
- [47] S. Uppu and A. Krishna, "Tuning Hyperparameters for Gene Interaction Models in Genome-Wide Association Studies," in *International Conference on Neural Information Processing*, 2017, pp. 791-801.
- [48] S. Uppu and A. Krishna, "[Regular Paper] An Intensive Search for Higher-Order Gene-Gene Interactions by Improving Deep Learning Model," in *2018 IEEE 18th International Conference on Bioinformatics and Bioengineering (BIBE)*, 2018, pp. 104-109.
- [49] S. Uppu and A. Krishna, "A deep hybrid model to detect multi-locus interacting SNPs in the presence of noise," *International journal of medical informatics*, vol. 119, pp. 134-151, 2018.

- [50] S. Uppu, A. Krishna, and R. Gopalan, "A review on methods for detecting SNP interactions in high-dimensional genomic data," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, 2018.
- [51] S. Uppu and A. Krishna, "Convolutional Model for Predicting SNP Interactions," in *International Conference on Neural Information Processing*, 2018, pp. 127-137.
- [52] K. C. Koenen, "Genetics of posttraumatic stress disorder: review and recommendations for future studies," *Journal of traumatic stress*, vol. 20, pp. 737-750, 2007.
- [53] W. S. Bush and J. H. Moore, "Genome-wide association studies," *PLoS Comput Biol*, vol. 8, p. e1002822, 2012.
- [54] S. Wernicke, *On the algorithmic tractability of single nucleotide polymorphism (SNP) analysis and related problems*: diplom. de, 2014.
- [55] N. L. o. M. (US). (2016). *Genetics Home Reference [Internet]*. Available: Available from: <http://ghr.nlm.nih.gov/>
- [56] O. L. Griffith, S. B. Montgomery, B. Bernier, B. Chu, K. Kasaian, S. Aerts, S. Mahony, M. C. Sleumer, M. Bilenky, and M. Haeussler, "ORegAnno: an open-access community-driven resource for regulatory annotation," *Nucleic acids research*, vol. 36, pp. D107-D113, 2008.
- [57] B.-s. Kerem, "Identification of the cystic fibrosis gene: genetic analysis," *TRENDS in Genetics*, vol. 5, p. 363, 1989.
- [58] R. Guy, "Machine Learning for Biostatisticians: A Hypothesis Driven Approach," Wake Forest University, 2010.
- [59] A. Antoniades, "Discovering disease associated gene-gene interactions: A two SNP interaction analysis framework," 2011.
- [60] B. Maher, "The case of the missing heritability," *Nature*, vol. 456, pp. 18-21, 2008.
- [61] X. Wang, R. C. Elston, and X. Zhu, "The meaning of interaction," *Human heredity*, vol. 70, pp. 269-277, 2010.
- [62] T. A. Manolio, J. E. Bailey-Wilson, and F. S. Collins, "Genes, environment and the value of prospective cohort studies," *Nature Reviews Genetics*, vol. 7, pp. 812-820, 2006.
- [63] A. Garrod, "The incidence of alkaptonuria: a study in chemical individuality," *The Lancet*, vol. 160, pp. 1616-1620, 1902.
- [64] D. J. Hunter, "Gene-environment interactions in human diseases," *Nature Reviews Genetics*, vol. 6, pp. 287-298, 2005.
- [65] D. Thomas, "Gene-environment-wide association studies: emerging approaches," *Nature Reviews Genetics*, vol. 11, pp. 259-272, 2010.
- [66] S. Ahmad, T. V. Varga, and P. W. Franks, "Genex environment interactions in obesity: the state of the evidence," *Human heredity*, vol. 75, pp. 106-115, 2013.
- [67] T. Huang and F. B. Hu, "Gene-environment interactions and obesity: recent developments and future directions," *BMC medical genomics*, vol. 8, p. 1, 2015.
- [68] W. Bateson, "Mendel's principles of heredity. ,," *Cambridge: Cambridge University Press*, 1909.
- [69] R. A. Fisher, "XV.—The Correlation between Relatives on the Supposition of Mendelian Inheritance," *Transactions of the royal society of Edinburgh*, vol. 52, pp. 399-433, 1919.
- [70] C. C. Cockerham, "An extension of the concept of partitioning hereditary variance for analysis of covariances among relatives when epistasis is present," *Genetics*, vol. 39, p. 859, 1954.

- [71] P. Li, M. Guo, C. Wang, X. Liu, and Q. Zou, "An overview of SNP interactions in genome-wide association studies," *Briefings in functional genomics*, p. elu036, 2014.
- [72] P. C. Phillips, "The language of gene interaction," *Genetics*, vol. 149, pp. 1167-1171, 1998.
- [73] B. A. McKinney, D. M. Reif, M. D. Ritchie, and J. H. Moore, "Machine learning for detecting gene-gene interactions," *Applied bioinformatics*, vol. 5, pp. 77-88, 2006.
- [74] J. H. Moore, "The ubiquitous nature of epistasis in determining susceptibility to common human diseases," *Human heredity*, vol. 56, pp. 73-82, 2003.
- [75] J. H. Moore and K. J. Mitchell, "The role of genetic interactions in neurodevelopmental disorders," *The Genetics of Neurodevelopmental Disorders*, p. 69, 2015.
- [76] K. J. Mitchell, *The Genetics of Neurodevelopmental Disorders*: John Wiley & Sons, 2015.
- [77] J. H. Moore, F. W. Asselbergs, and S. M. Williams, "Bioinformatics challenges for genome-wide association studies," *Bioinformatics*, vol. 26, pp. 445-455, 2010.
- [78] M. L. Calle, V. Urrea, N. Malatsi Riera, and K. Van Steen, "MB-MDR: model-based multifactor dimensionality reduction for detecting interactions in high-dimensional genomic data," 2008.
- [79] T. Hu and J. H. Moore, "Network modeling of statistical epistasis," *Biological Knowledge Discovery Handbook: Preprocessing, Mining, and Postprocessing of Biological Data*, eds. M. Elloumi and AY Zomaya (Wiley, 2013) pp, pp. 175-190, 2013.
- [80] S. F. Sheet, "Human genome project," *US Department of Energy genome Program's biological and environmental research information system (BERIS)*. (Cited on 2010 July 28) available from: [http://www.ornl.gov/sci/techresources/Human\\_Genome/](http://www.ornl.gov/sci/techresources/Human_Genome/).
- [81] I. Guyon, S. Gunn, M. Nikravesh, and L. Zadeh, "Feature extraction," *Foundations and applications*, 2006.
- [82] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *The Journal of Machine Learning Research*, vol. 3, pp. 1157-1182, 2003.
- [83] Y. Saeys, I. Inza, and P. Larrañaga, "A review of feature selection techniques in bioinformatics," *Bioinformatics*, vol. 23, pp. 2507-2517, 2007.
- [84] C. Dong, X. Chu, Y. Wang, Y. Wang, L. Jin, T. Shi, W. Huang, and Y. Li, "Exploration of gene-gene interaction effects using entropy-based methods," *European journal of human genetics*, vol. 16, pp. 229-235, 2007.
- [85] V. Varadan, D. M. Miller, and D. Anastassiou, "Computational inference of the molecular logic for synaptic connectivity in *C. elegans*," *Bioinformatics*, vol. 22, pp. e497-e506, 2006.
- [86] I. Kononenko, "Estimating attributes: analysis and extensions of RELIEF," in *Machine Learning: ECML-94*, 1994, pp. 171-182.
- [87] J. H. Moore and B. C. White, "Tuning ReliefF for genome-wide genetic analysis," in *Evolutionary computation, machine learning and data mining in bioinformatics*, ed: Springer, 2007, pp. 166-175.
- [88] C. S. Greene, N. M. Penrod, J. Kiralis, and J. H. Moore, "Spatially Uniform ReliefF (SURF) for computationally-efficient filtering of gene-gene interactions," *BioData mining*, vol. 2, pp. 1-9, 2009.
- [89] B. A. McKinney, D. M. Reif, B. C. White, J. Crowe, and J. H. Moore, "Evaporative cooling feature selection for genotypic data involving interactions," *Bioinformatics*, vol. 23, pp. 2113-2120, 2007.

- [90] R. Nunkesser, T. Bernholt, H. Schwender, K. Ickstadt, and I. Wegener, "Detecting high-order interactions of single nucleotide polymorphisms using genetic programming," *Bioinformatics*, vol. 23, pp. 3280-3288, 2007.
- [91] C. S. Greene, B. C. White, and J. H. Moore, "Ant colony optimization for genome-wide genetic analysis," in *Ant Colony Optimization and Swarm Intelligence*, ed: Springer, 2008, pp. 37-47.
- [92] Y. Wang, X. Liu, K. Robbins, and R. Rekaya, "AntEpiSeeker: detecting epistatic interactions for case-control studies using a two-stage ant colony optimization algorithm," *BMC research notes*, vol. 3, p. 117, 2010.
- [93] H. Zhang and G. Bonney, "Use of classification trees for association studies," *Genetic epidemiology*, vol. 19, pp. 323-332, 2000.
- [94] D. V. Zaykin and S. S. Young, "Large recursive partitioning analysis of complex disease pharmacogenetic studies. II. Statistical considerations," 2005.
- [95] P. Yang, Y. Hwa Yang, B. B Zhou, and A. Y Zomaya, "A review of ensemble methods in bioinformatics," *Current Bioinformatics*, vol. 5, pp. 296-308, 2010.
- [96] H. Schwender and K. Ickstadt, "Identification of SNP interactions using logic regression," *Biostatistics*, vol. 9, pp. 187-198, 2008.
- [97] R. Bellman, R. E. Bellman, R. E. Bellman, and R. E. Bellman, *Adaptive control processes: a guided tour* vol. 4: Princeton University Press Princeton, 1961.
- [98] P. Good, *Permutation tests*: Springer, 2000.
- [99] T. Hastie, R. Tibshirani, J. Friedman, and J. Franklin, "The elements of statistical learning: data mining, inference and prediction," *The Mathematical Intelligencer*, vol. 27, pp. 83-85, 2005.
- [100] M. Reich, T. Liefeld, J. Gould, J. Lerner, P. Tamayo, and J. P. Mesirov, "GenePattern 2.0," *Nature genetics*, vol. 38, pp. 500-501, 2006.
- [101] D. Reif, "Exploratory Visual Analysis of Pharmacogenomic Results DM Reif, SM Dudek, CM Shaffer, J. Wang, and JH Moore Pacific Symposium on Biocomputing 10: 296-307 (2005)," in *Pacific Symposium on Biocomputing*, 2005, pp. 296-307.
- [102] L. C. Tsoi, M. Boehnke, R. L. Klein, and W. J. Zheng, "Evaluation of genome-wide association study results through development of ontology fingerprints," *Bioinformatics*, vol. 25, pp. 1314-1320, 2009.
- [103] W. Yu, A. Wulf, T. Liu, M. J. Khoury, and M. Gwinn, "Gene Prospector: an evidence gateway for evaluating potential susceptibility genes and interacting risk factors for human diseases," *BMC bioinformatics*, vol. 9, p. 528, 2008.
- [104] D. L. Aylor and Z.-B. Zeng, "From classical genetics to quantitative genetics to systems biology: modeling epistasis," *PLoS genetics*, vol. 4, p. e1000029, 2008.
- [105] N. Sepúlveda, C. D. Paulino, and C. Penha-Gonçalves, "Bayesian analysis of allelic penetrance models for complex binary traits," *Computational Statistics & Data Analysis*, vol. 53, pp. 1271-1283, 2009.
- [106] V. Gayatonde, "Genome-wide association studies," University of Agricultural Sciences, Department of Genetics and Plant Breeding, 2013.
- [107] L. Su, G. Liu, H. Wang, Y. Tian, Z. Zhou, L. Han, and L. Yan, "Research on Single Nucleotide Polymorphisms Interaction Detection from Network Perspective," *PloS one*, vol. 10, p. e0119146, 2015.

- [108] D. R. Velez, B. C. White, A. A. Motsinger, W. S. Bush, M. D. Ritchie, S. M. Williams, and J. H. Moore, "A balanced accuracy function for epistasis modeling in imbalanced datasets using multifactor dimensionality reduction," *Genetic epidemiology*, vol. 31, pp. 306-315, 2007.
- [109] L. W. Hahn, M. D. Ritchie, and J. H. Moore, "Multifactor dimensionality reduction software for detecting gene–gene and gene–environment interactions," *Bioinformatics*, vol. 19, pp. 376-382, 2003.
- [110] S. Y. Lee, Y. Chung, R. C. Elston, Y. Kim, and T. Park, "Log-linear model-based multifactor dimensionality reduction method to detect gene–gene interactions," *Bioinformatics*, vol. 23, pp. 2589-2595, 2007.
- [111] K. A. Pattin and J. H. Moore, "Exploiting the proteome to improve the genome-wide genetic analysis of epistasis in common human diseases," *Human genetics*, vol. 124, pp. 19-29, 2008.
- [112] J. Namkung, K. Kim, S. Yi, W. Chung, M.-S. Kwon, and T. Park, "New evaluation measures for multifactor dimensionality reduction classifiers in gene–gene interaction analysis," *Bioinformatics*, vol. 25, pp. 338-345, 2009.
- [113] Y. Chung, S. Y. Lee, R. C. Elston, and T. Park, "Odds ratio based multifactor-dimensionality reduction method for detecting gene–gene interactions," *Bioinformatics*, vol. 23, pp. 71-76, 2007.
- [114] X.-Y. Lou, G.-B. Chen, L. Yan, J. Z. Ma, J. Zhu, R. C. Elston, and M. D. Li, "A generalized combinatorial approach for detecting gene-by-gene and gene-by-environment interactions with application to nicotine dependence," *The American Journal of Human Genetics*, vol. 80, pp. 1125-1137, 2007.
- [115] M. Calle, V. Urrea, G. Vellalta, N. Malats, and K. Steen, "Improving strategies for detecting genetic patterns of disease susceptibility in association studies," *Statistics in medicine*, vol. 27, pp. 6532-6546, 2008.
- [116] J. Gui, A. S. Andrew, P. Andrews, H. M. Nelson, K. T. Kelsey, M. R. Karagas, and J. H. Moore, "A robust multifactor dimensionality reduction method for detecting gene–gene interactions with application to the genetic analysis of bladder cancer susceptibility," *Annals of human genetics*, vol. 75, pp. 20-28, 2011.
- [117] A. A. Motsinger-Reif, "The effect of alternative permutation testing strategies on the performance of multifactor dimensionality reduction," *BMC research notes*, vol. 1, p. 1, 2008.
- [118] J. J. Gory, H. C. Sweeney, D. M. Reif, and A. A. Motsinger-Reif, "A comparison of internal model validation methods for multifactor dimensionality reduction in the case of genetic heterogeneity," *BMC research notes*, vol. 5, p. 623, 2012.
- [119] F. Van Lishout, J. M. M. John, E. S. Gusareva, V. Urrea, I. Cleynen, E. Théâtre, B. Charlotteaux, M. L. Calle, L. Wehenkel, and K. Van Steen, "An efficient algorithm to perform multiple testing in epistasis screening," *BMC bioinformatics*, vol. 14, p. 1, 2013.
- [120] D. Gola, J. M. Mahachie John, K. Van Steen, and I. R. König, "A roadmap to multifactor dimensionality reduction methods," *Briefings in bioinformatics*, vol. 17, pp. 293-308, 2015.
- [121] J. Han, M. Kamber, and J. Pei, *Data mining: concepts and techniques*: Morgan kaufmann, 2006.
- [122] L. Breiman, "Random forests," *Machine learning*, vol. 45, pp. 5-32, 2001.
- [123] Y. Qi, "Random Forest for Bioinformatics," in *Ensemble Machine Learning*, ed: Springer, 2012, pp. 307-323.

- [124] A. Liaw and M. Wiener, "Classification and regression by randomForest," *R news*, vol. 2, pp. 18-22, 2002.
- [125] X. Liu, K. Tang, J. R. Buhrman, and H. Cheng, "An agent-based framework for collaborative data mining optimization," in *Collaborative Technologies and Systems (CTS), 2010 International Symposium on*, 2010, pp. 295-301.
- [126] E. Frank, M. Hall, L. Trigg, G. Holmes, and I. H. Witten, "Data mining in bioinformatics using Weka," *Bioinformatics*, vol. 20, pp. 2479-2481, 2004.
- [127] H. Zhang, M. Wang, and X. Chen, "Willows: a memory efficient tree and forest construction package," *BMC bioinformatics*, vol. 10, p. 130, 2009.
- [128] L. De Lobel, P. Geurts, G. Baele, F. Castro-Giner, M. Kogevinas, and K. Van Steen, "A screening methodology based on random forests to improve the detection of gene–gene interactions," *European journal of human genetics*, vol. 18, pp. 1127-1132, 2010.
- [129] C. Yang, Z. He, X. Wan, Q. Yang, H. Xue, and W. Yu, "SNPHarvester: a filtering-based approach for detecting epistatic interactions in genome-wide association studies," *Bioinformatics*, vol. 25, pp. 504-511, 2009.
- [130] Q. Wu, Y. Ye, Y. Liu, and M. K. Ng, "SNP selection and classification of genome-wide SNP data using stratified sampling random forests," *NanoBioscience, IEEE Transactions on*, vol. 11, pp. 216-227, 2012.
- [131] H. Y. Lin, Y. Ann Chen, Y. Y. Tsai, X. Qu, T. S. Tseng, and J. Y. Park, "TRM: A Powerful Two-Stage Machine Learning Approach for Identifying SNP-SNP Interactions," *Annals of human genetics*, vol. 76, pp. 53-62, 2012.
- [132] Q. Pan, T. Hu, J. D. Malley, A. S. Andrew, M. R. Karagas, and J. H. Moore, *Supervising random forest using attribute interaction networks*: Springer, 2013.
- [133] A. Bureau, J. Dupuis, K. Falls, K. L. Lunetta, B. Hayward, T. P. Keith, and P. Van Eerdewegh, "Identifying SNPs predictive of phenotype using random forests," *Genetic epidemiology*, vol. 28, pp. 171-182, 2005.
- [134] A. A. Motsinger-Reif, S. M. Dudek, L. W. Hahn, and M. D. Ritchie, "Comparison of approaches for machine-learning optimization of neural networks for detecting gene-gene interactions in genetic epidemiology," *Genetic epidemiology*, vol. 32, pp. 325-340, 2008.
- [135] Y. Tomita, S. Tomida, Y. Hasegawa, Y. Suzuki, T. Shirakawa, T. Kobayashi, and H. Honda, "Artificial neural network approach for selection of susceptible single nucleotide polymorphisms and construction of prediction model on childhood allergic asthma," *BMC bioinformatics*, vol. 5, p. 120, 2004.
- [136] E. Keedwell and A. Narayanan, "Discovering gene networks with a neural-genetic hybrid," *Computational Biology and Bioinformatics, IEEE/ACM Transactions on*, vol. 2, pp. 231-242, 2005.
- [137] M. D. Ritchie, A. A. Motsinger, W. S. Bush, C. S. Coffey, and J. H. Moore, "Genetic programming neural networks: A powerful bioinformatics tool for human genetics," *Applied Soft Computing*, vol. 7, pp. 471-479, 2007.
- [138] A. A. Motsinger, S. M. Dudek, L. W. Hahn, and M. D. Ritchie, "Comparison of neural network optimization approaches for studies of human genetics," in *Applications of Evolutionary Computing*, ed: Springer, 2006, pp. 103-114.
- [139] N. E. Hardison and A. A. Motsinger-Reif, "The power of quantitative grammatical evolution neural networks to detect gene-gene interactions," in *Proceedings of the 13th annual conference on Genetic and evolutionary computation*, 2011, pp. 299-306.

- [140] S. D. Turner, S. M. Dudek, and M. D. Ritchie, "ATHENA: A knowledge-based hybrid backpropagation-grammatical evolution neural network algorithm for discovering epistasis among quantitative trait Loci," *BioData mining*, vol. 3, p. 5, 2010.
- [141] Y. Shen, Z. Liu, and J. Ott, "Detecting gene-gene interactions using support vector machines with L 1 penalty," in *Bioinformatics and Biomedicine Workshops (BIBMW), 2010 IEEE International Conference on*, 2010, pp. 309-311.
- [142] S. H. Chen, J. Sun, L. Dimitrov, A. R. Turner, T. S. Adams, D. A. Meyers, B. L. Chang, S. L. Zheng, H. Grönberg, and J. Xu, "A support vector machine approach for detecting gene-gene interaction," *Genetic epidemiology*, vol. 32, pp. 152-167, 2008.
- [143] N. Matchenko-Shimko and M.-P. Dube, "Gene-gene interaction tests using SVM and neural network modeling," in *Computational Intelligence and Bioinformatics and Computational Biology, 2007. CIBCB'07. IEEE Symposium on*, 2007, pp. 90-97.
- [144] A. Özgür, T. Vu, G. Erkan, and D. R. Radev, "Identifying gene-disease associations using centrality on a literature mined gene-interaction network," *Bioinformatics*, vol. 24, pp. i277-i285, 2008.
- [145] Y. Shen, Z. Liu, and J. Ott, "Support Vector Machines with L 1 penalty for detecting gene-gene interactions," *International journal of data mining and bioinformatics*, vol. 6, pp. 463-470, 2012.
- [146] Y. H. Fang and Y. F. Chiu, "SVM-Based Generalized Multifactor Dimensionality Reduction Approaches for Detecting Gene-Gene Interactions in Family Studies," *Genetic epidemiology*, vol. 36, pp. 88-98, 2012.
- [147] S. Marvel and A. Motsinger-Reif, "Grammatical evolution support vector machines for predicting human genetic disease association," in *Proceedings of the 14th annual conference companion on Genetic and evolutionary computation*, 2012, pp. 595-598.
- [148] H. Zhang, H. Wang, Z. Dai, M.-s. Chen, and Z. Yuan, "Improving accuracy for cancer classification with a new algorithm for genes selection," *BMC bioinformatics*, vol. 13, p. 298, 2012.
- [149] A. S. Andrew, M. R. Karagas, H. H. Nelson, S. Guerrera, S. Polidoro, S. Gamberini, C. Sacerdote, J. H. Moore, K. T. Kelsey, and E. Demidenko, "DNA repair polymorphisms modify bladder cancer risk: a multi-factor analytic strategy," *Human heredity*, vol. 65, pp. 105-118, 2007.
- [150] A. Fritsch and K. Ickstadt, "Comparing logic regression based methods for identifying SNP interactions," in *Bioinformatics research and development*, ed: Springer, 2007, pp. 90-103.
- [151] B. Atik, T. A. Skwor, R. P. Kandel, B. Sharma, H. K. Adhikari, L. Steiner, H. Erlich, and D. Dean, "Identification of novel single nucleotide polymorphisms in inflammatory genes as risk factors associated with trachomatous trichiasis," *PloS one*, vol. 3, p. e3600, 2008.
- [152] C. Kooperberg, J. C. Bis, K. D. Marciante, S. R. Heckbert, T. Lumley, and B. M. Psaty, "Logic regression for analysis of the association between genetic variation in the renin-angiotensin system and myocardial infarction or stroke," *American journal of epidemiology*, vol. 165, pp. 334-343, 2007.
- [153] S. Purcell, B. Neale, K. Todd-Brown, L. Thomas, M. A. Ferreira, D. Bender, J. Maller, P. Sklar, P. I. De Bakker, and M. J. Daly, "PLINK: a tool set for whole-genome association and population-based linkage analyses," *The American Journal of Human Genetics*, vol. 81, pp. 559-575, 2007.

- [154] L. Trotta, I. Guella, G. Soldà, F. Sironi, S. Tesei, M. Canesi, G. Pezzoli, S. Goldwurm, S. Duga, and R. Asselta, "SNCA and MAPT genes: Independent and joint effects in Parkinson disease in the Italian population," *Parkinsonism & related disorders*, vol. 18, pp. 257-262, 2012.
- [155] C. Kooperberg and I. Ruczinski, "Identifying interacting SNPs using Monte Carlo logic regression," *Genetic epidemiology*, vol. 28, pp. 157-170, 2005.
- [156] D. J. Lunn, J. C. Whittaker, and N. Best, "A Bayesian toolkit for genetic association studies," *Genetic epidemiology*, vol. 30, pp. 231-247, 2006.
- [157] C. F. Aliferis, I. Tsamardinos, and A. Statnikov, "HITON: a novel Markov Blanket algorithm for optimal variable selection," in *AMIA Annual Symposium Proceedings*, 2003, p. 21.
- [158] M. Dorigo and V. Maniezzo, "Colorni (1991) A positive feedback as a search strategy," Technical report 91-016, Politecnico di Milano, Italy.
- [159] P.-J. Jing and H.-B. Shen, "MACOED: a multi-objective ant colony optimization algorithm for SNP epistasis detection in genome-wide association studies," *Bioinformatics*, p. btu702, 2014.
- [160] R. Rekaya and K. Robbins, "Ant colony algorithm for analysis of gene interaction in high-dimensional association data," *Revista Brasileira de Zootecnia*, vol. 38, pp. 93-97, 2009.
- [161] E. Sapin, E. Keedwell, and T. Frayling, "Subset-based ant colony optimisation for the discovery of gene-gene interactions in genome wide association studies," in *Proceedings of the 15th annual conference on Genetic and evolutionary computation*, 2013, pp. 295-302.
- [162] E. Sapin, E. Keedwell, and T. Frayling, "Ant colony optimisation of decision trees for the detection of gene-gene interactions," in *Bioinformatics and Biomedicine (BIBM), 2014 IEEE International Conference on*, 2014, pp. 57-61.
- [163] C. S. Greene, J. M. Gilmore, J. Kiralis, P. C. Andrews, and J. H. Moore, "Optimal use of expert knowledge in ant colony optimization for the analysis of epistasis in human disease," in *Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics*, ed: Springer, 2009, pp. 92-103.
- [164] X. Zhang, F. Zou, and W. Wang, "Fastanova: an efficient algorithm for genome-wide association study," in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2008, pp. 821-829.
- [165] X. Zhang, F. Pan, Y. Xie, F. Zou, and W. Wang, "COE: a general approach for efficient genome-wide two-locus epistasis test in disease association study," in *Annual International Conference on Research in Computational Molecular Biology*, 2009, pp. 253-269.
- [166] B. Goudy, D. Rawlinson, Q. Wang, F. Shi, H. Ferra, R. M. Campbell, L. Stern, M. T. Inouye, C. S. Ong, and A. Kowalczyk, "GWIS-model-free, fast and exhaustive search for epistatic interactions in case-control GWAS," *BMC genomics*, vol. 14, p. 1, 2013.
- [167] L. Zou, Q. Huang, A. Li, and M. Wang, "A genome-wide association study of Alzheimer's disease using random forests and enrichment analysis," *Science China Life Sciences*, vol. 55, pp. 618-625, 2012.
- [168] B. A. Goldstein, A. E. Hubbard, A. Cutler, and L. F. Barcellos, "An application of Random Forests to a genome-wide association dataset: methodological considerations & new findings," *BMC genetics*, vol. 11, p. 49, 2010.
- [169] A. Staiano, M. D. Di Taranto, E. Bloise, M. N. D'Agostino, A. D'Angelo, G. Marotta, M. Gentile, F. Jossa, A. Iannuzzi, and P. Rubba, "Investigation of single nucleotide polymorphisms associated to familial combined Hyperlipidemia with random forests," in *Neural Nets and Surroundings*, ed: Springer, 2013, pp. 169-178.

- [170] X. Chen and H. Ishwaran, "Pathway hunting by random survival forests," *Bioinformatics*, vol. 29, pp. 99-105, 2013.
- [171] C. Liu, H. H. Ackerman, and J. P. Carulli, "A genome-wide screen of gene–gene interactions for rheumatoid arthritis susceptibility," *Human genetics*, vol. 129, pp. 473-485, 2011.
- [172] A. A. Motsinger, S. L. Lee, G. Mellick, and M. D. Ritchie, "GPNN: Power studies and applications of a neural network method for detecting gene-gene interactions in studies of human disease," *BMC bioinformatics*, vol. 7, p. 39, 2006.
- [173] H.-J. Ban, J. Y. Heo, K.-S. Oh, and K.-J. Park, "Identification of type 2 diabetes-associated combination of SNPs using support vector machine," *BMC genetics*, vol. 11, p. 26, 2010.
- [174] R. J. Klein, C. Zeiss, E. Y. Chew, J.-Y. Tsai, R. S. Sackler, C. Haynes, A. K. Henning, J. P. SanGiovanni, S. M. Mane, and S. T. Mayne, "Complement factor H polymorphism in age-related macular degeneration," *Science*, vol. 308, pp. 385-389, 2005.
- [175] W. T. C. C. Consortium, "Genome-wide association study of copy number variation in 16,000 cases of eight common diseases and 3,000 shared controls," *Nature*, vol. 464, p. 713, 2010.
- [176] J. Shang, J. Zhang, Y. Sun, D. Liu, D. Ye, and Y. Yin, "Performance analysis of novel methods for detecting epistasis," *BMC bioinformatics*, vol. 12, p. 475, 2011.
- [177] J. Marchini, P. Donnelly, and L. R. Cardon, "Genome-wide strategies for detecting multiple loci that influence complex diseases," *Nature genetics*, vol. 37, pp. 413-417, 2005.
- [178] T. L. Edwards, S. D. Turner, E. S. Torstenson, S. M. Dudek, E. R. Martin, and M. D. Ritchie, "A general framework for formal tests of interaction after exhaustive search methods with applications to MDR and MDR-PDT," *PLoS one*, vol. 5, p. e9363, 2010.
- [179] T. Zheng, H. Wang, and S.-H. Lo, "Backward genotype-trait association (BGTA)-based dissection of complex traits in case-control designs," *Human heredity*, vol. 62, pp. 196-212, 2006.
- [180] X. Zhang, F. Zou, and W. Wang, "FastChi: an efficient algorithm for analyzing gene-gene interactions," in *Biocomputing 2009*, ed: World Scientific, 2009, pp. 528-539.
- [181] X. Jiang, M. M. Barmada, and S. Visweswaran, "Identifying genetic interactions in genome-wide data using Bayesian networks," *Genetic epidemiology*, vol. 34, pp. 575-581, 2010.
- [182] S. J. Winham, C. L. Colby, R. R. Freimuth, X. Wang, M. De Andrade, M. Huebner, and J. M. Biernacka, "SNP interaction detection with random forests in high-dimensional genetic data," *BMC bioinformatics*, vol. 13, p. 164, 2012.
- [183] W. Tang, X. Wu, R. Jiang, and Y. Li, "Epistatic module detection for case-control studies: a Bayesian model with a Gibbs sampling strategy," *PLoS Genet*, vol. 5, p. e1000464, 2009.
- [184] J. Hoh, A. Wille, and J. Ott, "Trimming, weighting, and grouping SNPs in human case-control association studies," *Genome Research*, vol. 11, pp. 2115-2119, 2001.
- [185] D. M. Evans, J. Marchini, A. P. Morris, and L. R. Cardon, "Two-stage two-locus models in genome-wide association," *PLoS genetics*, vol. 2, p. e157, 2006.
- [186] C. Herold, M. Steffens, F. F. Brockschmidt, M. P. Baur, and T. Becker, "INTERSNP: genome-wide interaction analysis guided by a priori information," *Bioinformatics*, vol. 25, pp. 3275-3281, 2009.
- [187] J. R. Kilpatrick, "Methods for detecting multi-locus genotype-phenotype association," Rice University, 2010.

- [188] D. J. Miller, Y. Zhang, G. Yu, Y. Liu, L. Chen, C. D. Langefeld, D. Herrington, and Y. Wang, "An algorithm for learning maximum entropy probability models of disease risk that efficiently searches and sparingly encodes multilocus genomic interactions," *Bioinformatics*, vol. 25, pp. 2478-2485, 2009.
- [189] G. Fang, M. Haznadar, W. Wang, H. Yu, M. Steinbach, T. R. Church, W. S. Oetting, B. Van Ness, and V. Kumar, "High-order SNP combinations associated with complex diseases: efficient discovery, statistical power and functional interactions," *Plos one*, vol. 7, p. e33531, 2012.
- [190] S. Leem, H.-h. Jeong, J. Lee, K. Wee, and K.-A. Sohn, "Fast detection of high-order epistatic interactions in genome-wide association studies using information theoretic measure," *Computational biology and chemistry*, vol. 50, pp. 19-28, 2014.
- [191] X. Wan, C. Yang, Q. Yang, H. Xue, N. L. Tang, and W. Yu, "Predictive rule inference for epistatic interaction detection in genome-wide association studies," *Bioinformatics*, vol. 26, pp. 30-37, 2010.
- [192] X. Yuan, D. J. Miller, J. Zhang, D. Herrington, and Y. Wang, "An overview of population genetic data simulation," *Journal of Computational Biology*, vol. 19, pp. 42-54, 2012.
- [193] A. Carvajal-Rodríguez, "Simulation of genomes: a review," *Current genomics*, vol. 9, p. 155, 2008.
- [194] C. J. Hoggart, M. Chadeau-Hyam, T. G. Clark, R. Lampariello, J. C. Whittaker, M. De Iorio, and D. J. Balding, "Sequence-level population simulations over large genomic regions," *Genetics*, vol. 177, pp. 1725-1731, 2007.
- [195] S. Hoban, G. Bertorelle, and O. E. Gaggiotti, "Computer simulations: tools for population and evolutionary genetics," *Nature Reviews Genetics*, vol. 13, pp. 110-122, 2012.
- [196] J. F. C. Kingman, "The coalescent," *Stochastic processes and their applications*, vol. 13, pp. 235-248, 1982.
- [197] J. F. Kingman, "Origins of the coalescent: 1974-1982," *Genetics*, vol. 156, pp. 1461-1463, 2000.
- [198] G. K. Chen, P. Marjoram, and J. D. Wall, "Fast and flexible simulation of DNA sequence data," *Genome Research*, vol. 19, pp. 136-142, 2009.
- [199] L. Liang, S. Zöllner, and G. R. Abecasis, "GENOME: a rapid coalescent-based whole genome simulator," *Bioinformatics*, vol. 23, pp. 1565-1567, 2007.
- [200] T. Mailund, M. H. Schierup, C. N. Pedersen, P. J. Mechlenborg, J. N. Madsen, and L. Schauser, "CoaSim: a flexible environment for simulating genetic data under coalescent models," *BMC bioinformatics*, vol. 6, p. 252, 2005.
- [201] J. Shang, J. Zhang, X. Lei, W. Zhao, and Y. Dong, "EpiSIM: simulation of multiple epistasis, linkage disequilibrium patterns and haplotype blocks for genome-wide interaction analysis," *Genes & Genomics*, vol. 35, pp. 305-316, 2013.
- [202] A. Antoniades, "Discovering disease associated gene–gene interactions: A two snp interaction analysis framework," Ph. D. dissertation, University of Cyprus, Cyprus, 2011.
- [203] Y.-X. Fu and W.-H. Li, "Coalescing into the 21st century: an overview and prospects of coalescent theory," *Theoretical Population Biology*, vol. 56, pp. 1-10, 1999.
- [204] T. Günther, I. Gawenda, and K. J. Schmid, "phenosim-A software to simulate phenotypes for testing in genome-wide association studies," *BMC bioinformatics*, vol. 12, p. 1, 2011.

- [205] T. L. Edwards, W. S. Bush, S. D. Turner, S. M. Dudek, E. S. Torstenson, M. Schmidt, E. Martin, and M. D. Ritchie, "Generating linkage disequilibrium patterns in data simulations using genomeSIMLA," in *Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics*, ed: Springer, 2008, pp. 24-35.
- [206] F. Guillaume and J. Rougemont, "Nemo: an evolutionary and population genetics programming framework," *Bioinformatics*, vol. 22, pp. 2556-2557, 2006.
- [207] S. Neuenschwander, F. Guillaume, and J. Goudet, "quantiNemo: an individual-based program to simulate quantitative traits with explicit genetic architecture in a dynamic metapopulation," *Bioinformatics*, vol. 24, pp. 1552-1553, 2008.
- [208] F. Rebaudo, A. Rouzic, S. Dupas, J. F. Silvain, M. Harry, and O. Dangles, "SimAdapt: an individual-based genetic model for simulating landscape management impacts on populations," *Methods in Ecology and Evolution*, vol. 4, pp. 595-600, 2013.
- [209] B. Peng and C. I. Amos, "Forward-time simulation of realistic samples for genome-wide association studies," *BMC bioinformatics*, vol. 11, p. 442, 2010.
- [210] B. Peng and C. I. Amos, "Forward-time simulations of non-random mating populations using simuPOP," *Bioinformatics*, vol. 24, pp. 1408-1409, 2008.
- [211] B. O'Fallon, "TreesimJ: a flexible, forward time population genetic simulator," *Bioinformatics*, vol. 26, pp. 2200-2201, 2010.
- [212] C. Li and M. Li, "GWAsimulator: a rapid whole-genome simulation program," *Bioinformatics*, vol. 24, pp. 140-142, 2008.
- [213] Software Resource List Available: <https://popmodels.cancercontrol.cancer.gov>.
- [214] L. Excoffier and M. Foll, "Fastsimcoal: a continuous-time coalescent simulator of genomic diversity under arbitrarily complex evolutionary scenarios," *Bioinformatics*, vol. 27, pp. 1332-1334, 2011.
- [215] S. E. Ramos-Onsins and T. Mitchell-Olds, "MLcoalsim: multilocus coalescent simulations," *Evolutionary bioinformatics online*, vol. 3, p. 41, 2007.
- [216] G. Ewing and J. Hermisson, "MSMS: a coalescent simulation program including recombination, demographic structure and selection at a single locus," *Bioinformatics*, vol. 26, pp. 2064-2065, 2010.
- [217] G. Laval and L. Excoffier, "SIMCOAL 2.0: a program to simulate genomic diversity over large recombining regions in a subdivided population with a complex history," *Bioinformatics*, vol. 20, pp. 2485-2487, 2004.
- [218] B. Li, G. Wang, and S. M. Leal, "SimRare: a program to generate and analyze sequence-based data for association studies of quantitative and qualitative traits," *Bioinformatics*, vol. 28, pp. 2703-2704, 2012.
- [219] Z. Su, J. Marchini, and P. Donnelly, "HAPGEN2: simulation of multiple disease SNPs," *Bioinformatics*, vol. 27, pp. 2304-2305, 2011.
- [220] F. A. Wright, H. Huang, X. Guan, K. Gamiel, C. Jeffries, W. T. Barry, F. P.-M. de Villena, P. F. Sullivan, K. C. Wilhelmsen, and F. Zou, "Simulating association studies: a data-based resampling method for candidate regions or whole genome scans," *Bioinformatics*, vol. 23, pp. 2581-2588, 2007.
- [221] R. J. Urbanowicz, J. Kiralis, N. A. Sinnott-Armstrong, T. Heberling, J. M. Fisher, and J. H. Moore, "GAMETES: a fast, direct algorithm for generating pure, strict, epistatic models with random architectures," *BioData mining*, vol. 5, pp. 1-14, 2012.

- [222] P. R. Burton, D. G. Clayton, L. R. Cardon, N. Craddock, P. Deloukas, A. Duncanson, D. P. Kwiatkowski, M. I. McCarthy, W. H. Ouwehand, and N. J. Samani, "Genome-wide association study of 14,000 cases of seven common diseases and 3,000 shared controls," *Nature*, vol. 447, pp. 661-678, 2007.
- [223] J. N. Weinstein, E. A. Collisson, G. B. Mills, K. R. M. Shaw, B. A. Ozenberger, K. Ellrott, I. Shmulevich, C. Sander, J. M. Stuart, and C. G. A. R. Network, "The cancer genome atlas pan-cancer analysis project," *Nature genetics*, vol. 45, pp. 1113-1120, 2013.
- [224] TCGA, "The cancer genome atlas (TCGA)," <http://cancergenome.nih.gov/>, ed.
- [225] Gpd. *Gene pattern datasets*  
Available: <http://software.broadinstitute.org/cancer/software/genepattern/datasets/>
- [226] D. Gilbert, "Biomolecular interaction network database," *Briefings in bioinformatics*, vol. 6, pp. 194-198, 2005.
- [227] A. Chatr-Aryamontri, A. Ceol, L. M. Palazzi, G. Nardelli, M. V. Schneider, L. Castagnoli, and G. Cesareni, "MINT: the Molecular INTERaction database," *Nucleic acids research*, vol. 35, pp. D572-D574, 2007.
- [228] T. Keshava Prasad, R. Goel, K. Kandasamy, S. Keerthikumar, S. Kumar, S. Mathivanan, D. Telikicherla, R. Raju, B. Shafreen, and A. Venugopal, "Human protein reference database—2009 update," *Nucleic acids research*, vol. 37, pp. D767-D772, 2008.
- [229] L. Salwinski, C. S. Miller, A. J. Smith, F. K. Pettit, J. U. Bowie, and D. Eisenberg, "The database of interacting proteins: 2004 update," *Nucleic acids research*, vol. 32, pp. D449-D451, 2004.
- [230] C. Stark, B.-J. Breitkreutz, T. Reguly, L. Boucher, A. Breitkreutz, and M. Tyers, "BioGRID: a general repository for interaction datasets," *Nucleic acids research*, vol. 34, pp. D535-D539, 2006.
- [231] G. Joshi-Tope, M. Gillespie, I. Vastrik, P. D'Eustachio, E. Schmidt, B. de Bono, B. Jassal, G. Gopinath, G. Wu, and L. Matthews, "Reactome: a knowledgebase of biological pathways," *Nucleic acids research*, vol. 33, pp. D428-D432, 2005.
- [232] K. A. Pattin and J. H. Moore, "Role for protein–protein interaction databases in human genetics," *Expert review of proteomics*, vol. 6, pp. 647-659, 2009.
- [233] W. S. Bush, S. M. Dudek, and M. D. Ritchie, "Biofilter: a knowledge-integration system for the multi-locus analysis of genome-wide association studies," in *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, 2009, p. 368.
- [234] M. D. Ritchie, "Using Biological Knowledge to Uncover the Mystery in the Search for Epistasis in Genome-Wide Association Studies," *Annals of human genetics*, vol. 75, pp. 172-182, 2011.
- [235] I. Foster, Y. Zhao, I. Raicu, and S. Lu, "Cloud computing and grid computing 360-degree compared," in *Grid Computing Environments Workshop, 2008. GCE'08*, 2008, pp. 1-10.
- [236] T. Schüpbach, I. Xenarios, S. Bergmann, and K. Kapur, "FastEpistasis: a high performance computing solution for quantitative trait epistasis," *Bioinformatics*, vol. 26, pp. 1468-1469, 2010.
- [237] L. Ma, H. B. Runesha, D. Dvorkin, J. R. Garbe, and Y. Da, "Parallel and serial computing tools for testing single-locus and epistatic SNP effects of quantitative traits in genome-wide association studies," *BMC bioinformatics*, vol. 9, p. 315, 2008.
- [238] O. Kempthorne, "The correlation between relatives in a random mating population," *Proceedings of the Royal Society of London. Series B-Biological Sciences*, vol. 143, pp. 103-113, 1954.

- [239] C. S. Greene, N. A. Sinnott-Armstrong, D. S. Himmelstein, P. J. Park, J. H. Moore, and B. T. Harris, "Multifactor dimensionality reduction for graphics processing units enables genome-wide testing of epistasis in sporadic ALS," *Bioinformatics*, vol. 26, pp. 694-695, 2010.
- [240] G. Hemani, A. Theodaridis, W. Wei, and C. Haley, "EpiGPU: exhaustive pairwise epistasis scans parallelized on consumer level graphics cards," *Bioinformatics*, vol. 27, pp. 1462-1465, 2011.
- [241] X. Hu, Q. Liu, Z. Zhang, Z. Li, S. Wang, L. He, and Y. Shi, "SHEsisEpi, a GPU-enhanced genome-wide SNP-SNP interaction scanning algorithm, efficiently reveals the risk genetic epistasis in bipolar disorder," *Cell research*, vol. 20, pp. 854-857, 2010.
- [242] Z. Wang, Y. Wang, K.-L. Tan, L. Wong, and D. Agrawal, "eCEO: an efficient Cloud Epistasis cOmputing model in genome-wide association study," *Bioinformatics*, vol. 27, pp. 1045-1051, 2011.
- [243] Z. Zhu, X. Tong, Z. Zhu, M. Liang, W. Cui, K. Su, M. D. Li, and J. Zhu, "Development of GMDR-GPU for gene-gene interaction analysis and its application to WTCCC GWAS data for type 2 diabetes," *PLoS one*, vol. 8, p. e61943, 2013.
- [244] J. González-Domínguez, B. Schmidt, J. C. Kässens, and L. Wienbrandt, "Hybrid CPU/GPU acceleration of detection of 2-SNP epistatic interactions in GWAS," in *Euro-Par 2014 Parallel Processing*, ed: Springer, 2014, pp. 680-691.
- [245] D. Sluga, T. Curk, B. Zupan, and U. Lotric, "Heterogeneous computing architecture for fast detection of SNP-SNP interactions," *BMC bioinformatics*, vol. 15, p. 216, 2014.
- [246] X. Guo, Y. Meng, N. Yu, and Y. Pan, "Cloud computing for detecting high-order genome-wide epistatic interaction via dynamic clustering," *BMC bioinformatics*, vol. 15, p. 1, 2014.
- [247] B. Goudey, M. Abedini, J. L. Hopper, M. Inouye, E. Makalic, D. F. Schmidt, J. Wagner, Z. Zhou, J. Zobel, and M. Reumann, "High performance computing enabling exhaustive analysis of higher order single nucleotide polymorphism interaction in Genome Wide Association Studies," *Health Information Science and Systems*, vol. 1, p. 3, 2015.
- [248] J. González-Domínguez and B. Schmidt, "GPU-accelerated exhaustive search for third-order epistatic interactions in case-control studies," *Journal of Computational Science*, vol. 8, pp. 93-100, 2015.
- [249] J. C. Kässens, L. Wienbrandt, J. González-Domínguez, B. Schmidt, and M. Schimmler, "High-speed exhaustive 3-locus interaction epistasis analysis on FPGAs," *Journal of Computational Science*, vol. 9, pp. 131-136, 2015.
- [250] V. K. Ramanan, L. Shen, J. H. Moore, and A. J. Saykin, "Pathway analysis of genomic data: concepts, methods, and prospects for future development," *TRENDS in Genetics*, vol. 28, pp. 323-332, 2012.
- [251] R. M. Cantor, K. Lange, and J. S. Sinsheimer, "Prioritizing GWAS results: a review of statistical methods and recommendations for their application," *The American Journal of Human Genetics*, vol. 86, pp. 6-22, 2010.
- [252] R.-H. Chung, "PUPPI: A pathway analysis method using protein-protein interaction network for case-control data," in *Computational Intelligence in Bioinformatics and Computational Biology (CIBCB), 2013 IEEE Symposium on*, 2013, pp. 238-241.
- [253] G. D. Bader, M. P. Cary, and C. Sander, "Pathguide: a pathway resource list," *Nucleic acids research*, vol. 34, pp. D504-D506, 2006.
- [254] C. F. Schaefer, K. Anthony, S. Krupa, J. Buchoff, M. Day, T. Hannay, and K. H. Buetow, "PID: the pathway interaction database," *Nucleic acids research*, vol. 37, pp. D674-D679, 2008.

- [255] M. Kanehisa and S. Goto, "KEGG: kyoto encyclopedia of genes and genomes," *Nucleic acids research*, vol. 28, pp. 27-30, 2000.
- [256] G. O. Consortium, "The Gene Ontology (GO) database and informatics resource," *Nucleic acids research*, vol. 32, pp. D258-D261, 2004.
- [257] G. Dennis, B. T. Sherman, D. A. Hosack, J. Yang, W. Gao, H. C. Lane, and R. A. Lempicki, "DAVID: database for annotation, visualization, and integrated discovery," *Genome biology*, vol. 4, p. R60, 2003.
- [258] D. Nishimura, "BioCarta," *Biotech Software & Internet Report: The Computer Software Journal for Scientist*, vol. 2, pp. 117-120, 2001.
- [259] P. D. Thomas, M. J. Campbell, A. Kejariwal, H. Mi, B. Karlak, R. Daverman, K. Diemer, A. Muruganujan, and A. Narechania, "PANTHER: a library of protein families and subfamilies indexed by function," *Genome Research*, vol. 13, pp. 2129-2141, 2003.
- [260] L. Bertram, M. B. McQueen, K. Mullin, D. Blacker, and R. E. Tanzi, "Systematic meta-analyses of Alzheimer disease genetic association studies: the AlzGene database," *Nature genetics*, vol. 39, pp. 17-23, 2007.
- [261] J. Zhu, J. Z. Sanborn, S. Benz, C. Szeto, F. Hsu, R. M. Kuhn, D. Karolchik, J. Archie, M. E. Lenburg, and L. J. Esserman, "The UCSC cancer genomics browser," *Nature methods*, vol. 6, pp. 239-240, 2009.
- [262] C. O'dushlaine, E. Kenny, E. A. Heron, R. Segurado, M. Gill, D. W. Morris, and A. Corvin, "The SNP ratio test: pathway analysis of genome-wide association datasets," *Bioinformatics*, vol. 25, pp. 2762-2763, 2009.
- [263] A. V. Segrè, L. Groop, V. K. Mootha, M. J. Daly, D. Altshuler, D. Consortium, and M. Investigators, "Common inherited variation in mitochondrial genes is not enriched for associations with type 2 diabetes or related glycemic traits," *PLoS genetics*, vol. 6, p. e1001058, 2010.
- [264] B. L. Yaspan, W. S. Bush, E. S. Torstenson, D. Ma, M. A. Pericak-Vance, M. D. Ritchie, J. S. Sutcliffe, and J. L. Haines, "Genetic analysis of biological pathway data through genomic randomization," *Human genetics*, vol. 129, pp. 563-571, 2011.
- [265] D. Zamar, B. Tripp, G. Ellis, and D. Daley, "Path: a tool to facilitate pathway-based genetic association analysis," *Bioinformatics*, vol. 25, pp. 2444-2446, 2009.
- [266] C. Herold, M. Mattheisen, A. Lacour, T. Vaitsiakhovich, M. Angisch, D. Drichel, and T. Becker, "Integrated genome-wide pathway association analysis with INTERSNP," *Human heredity*, vol. 73, pp. 63-72, 2012.
- [267] K. Wang, H. Zhang, S. Kugathasan, V. Annese, J. P. Bradfield, R. K. Russell, P. M. Sleiman, M. Imlielinski, J. Glessner, and C. Hou, "Diverse genome-wide association studies associate the IL12/IL23 pathway with Crohn Disease," *The American Journal of Human Genetics*, vol. 84, pp. 399-405, 2009.
- [268] E. J. Rossin, K. Lage, S. Raychaudhuri, R. J. Xavier, D. Tatar, Y. Benita, C. Cotsapas, M. J. Daly, and I. I. B. D. G. Constortium, "Proteins encoded in genomic regions associated with immune-mediated disease physically interact and suggest underlying biology," *PLoS genetics*, vol. 7, p. e1001273, 2011.
- [269] M. Holden, S. Deng, L. Wojnowski, and B. Kulle, "GSEA-SNP: applying gene set enrichment analysis to SNP data from genome-wide association studies," *Bioinformatics*, vol. 24, pp. 2784-2785, 2008.

- [270] H. Gui, M. Li, P. C. Sham, and S. S. Cherny, "Comparisons of seven algorithms for pathway analysis using the WTCCC Crohn's Disease dataset," *BMC research notes*, vol. 4, p. 386, 2011.
- [271] L.-C. Chuang, C.-F. Kao, W.-L. Shih, and P.-H. Kuo, "Pathway analysis using information from allele-specific gene methylation in genome-wide association studies for bipolar disorder," *PloS one*, vol. 8, p. e53092, 2013.
- [272] S. E. Baranzini, N. W. Galwey, J. Wang, P. Khankhanian, R. Lindberg, D. Pelletier, W. Wu, B. M. Uitdehaag, L. Kappos, and G. Consortium, "Pathway and network-based analysis of genome-wide association studies in multiple sclerosis," *Human molecular genetics*, vol. 18, pp. 2078-2090, 2009.
- [273] T. G. Lesnick, S. Papapetropoulos, D. C. Mash, J. Ffrench-Mullen, L. Shehadeh, M. de Andrade, J. R. Henley, W. A. Rocca, J. E. Ahlskog, and D. M. Maraganore, "A genomic pathway approach to a complex disease: axon guidance and Parkinson disease," *PLoS genetics*, vol. 3, p. e98, 2007.
- [274] Y. H. Lee, J.-H. Kim, and G. G. Song, "Pathway analysis of a genome-wide association study in schizophrenia," *Gene*, vol. 525, pp. 107-115, 2013.
- [275] F. Büchel, F. Mittag, C. Wrzodek, A. Zell, T. Gasser, and M. Sharma, "Integrative pathway-based approach for genome-wide association studies: identification of new pathways for rheumatoid arthritis and type 1 diabetes," *PloS one*, vol. 8, p. e78577, 2013.
- [276] F. Thabtah, "A review of associative classification mining," *The Knowledge Engineering Review*, vol. 22, pp. 37-65, 2007.
- [277] U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, "Advances in knowledge discovery and data mining," 1996.
- [278] S. Wedyan, "Review and comparison of associative classification data mining approaches," *International Journal of Computer, Information, Systems and Control Engineering*, vol. 8, pp. 34-45, 2014.
- [279] R. Agrawal, T. Imieliński, and A. Swami, "Mining association rules between sets of items in large databases," in *Acm sigmod record*, 1993, pp. 207-216.
- [280] P. Yu and D. J. Wild, "Fast rule-based bioactivity prediction using associative classification mining," *Journal of cheminformatics*, vol. 4, pp. 1-10, 2012.
- [281] B. L. W. H. Y. Ma, "Integrating classification and association rule mining," in *Proceedings of the 4th*, 1998.
- [282] W. Li, J. Han, and J. Pei, "CMAR: Accurate and efficient classification based on multiple class-association rules," in *Data Mining, 2001. ICDM 2001, Proceedings IEEE International Conference on*, 2001, pp. 369-376.
- [283] J. Han, "CPAR: Classification based on predictive association rules," in *Proceedings of the third SIAM international conference on data mining*, 2003, pp. 331-335.
- [284] G. Kundu, M. M. Islam, S. Munir, and M. F. Bari, "ACN: An associative classifier with negative rules," in *Computational Science and Engineering, 2008. CSE'08. 11th IEEE International Conference on*, 2008, pp. 369-375.
- [285] M.-L. Antonie and O. R. Zaïane, "An associative classifier based on positive and negative rules," in *Proceedings of the 9th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery*, 2004, pp. 64-69.
- [286] E. Baralis, S. Chiusano, and P. Garza, "A lazy approach to associative classification," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 20, pp. 156-171, 2008.

- [287] X. Wang, K. Yue, W. Niu, and Z. Shi, "An approach for adaptive associative classification," *Expert Systems with Applications*, vol. 38, pp. 11873-11883, 2011.
- [288] F. A. Thabtah, P. Cowling, and Y. Peng, "MMAC: A new multi-class, multi-label associative classification approach," in *Data Mining, 2004. ICDM'04. Fourth IEEE International Conference on*, 2004, pp. 217-224.
- [289] F. Thabtah, P. Cowling, and Y. Peng, "MCAR: multi-class classification based on association rule," in *Computer Systems and Applications, 2005. The 3rd ACS/IEEE International Conference on*, 2005, p. 33.
- [290] M. Nandhini and S. Sivanandam, "An improved predictive association rule based classifier using gain ratio and T-test for health care data diagnosis," *Sadhana*, vol. 40, pp. 1683-1699, 2015.
- [291] P. C. Phillips, "Epistasis—the essential role of gene interactions in the structure and evolution of genetic systems," *Nature Reviews Genetics*, vol. 9, pp. 855-867, 2008.
- [292] W. Li and J. Reich, "A complete enumeration and classification of two-locus disease models," *Human heredity*, vol. 50, pp. 334-349, 2000.
- [293] J. H. Moore, L. W. Hahn, M. D. Ritchie, T. A. Thornton, and B. C. White, "Application of genetic algorithms to the discovery of complex models for simulation studies in human genetics," in *Proceedings of the Genetic and Evolutionary Computation Conference/GECCO. Genetic and Evolutionary Computation Conference*, 2002, p. 1150.
- [294] W. N. Frankel and N. J. Schork, "Who's afraid of epistasis?," *Nature genetics*, vol. 14, pp. 371-373, 1996.
- [295] W. Weka, "3: data mining software in Java," *University of Waikato, Hamilton, New Zealand* ([www.cs.waikato.ac.nz/ml/weka](http://www.cs.waikato.ac.nz/ml/weka)), vol. 19, p. 52, 2011.
- [296] J. M. Akey, K. Zhang, M. Xiong, P. Doris, and L. Jin, "The effect that genotyping errors have on the robustness of common linkage-disequilibrium measures," *The American Journal of Human Genetics*, vol. 68, pp. 1447-1456, 2001.
- [297] J. R. González, L. Armengol, X. Solé, E. Guinó, J. M. Mercader, X. Estivill, and V. Moreno, "SNPassoc: an R package to perform whole genome association studies," *Bioinformatics*, vol. 23, pp. 654-655, 2007.
- [298] L. A. Juan R González, Elisabet Guinó, Xavier Solé, and Víctor Moreno. (2014-04-23) SNPs-based whole genome association studies. Available: <https://cran.r-project.org/web/packages/SNPassoc/SNPassoc.pdf>
- [299] F.-T. Chiang, K.-L. Hsu, C.-D. Tseng, W.-H. Hsiao, H.-M. Lo, T.-H. Chern, and Y.-Z. Tseng, "Molecular variant M235T of the angiotensinogen gene is associated with essential hypertension in Taiwanese," *Journal of hypertension*, vol. 15, pp. 607-611, 1997.
- [300] S.-J. Wu, F.-T. Chiang, W. J. Chen, P.-H. Liu, K.-L. Hsu, J.-J. Hwang, L.-P. Lai, J.-L. Lin, C.-D. Tseng, and Y.-Z. Tseng, "Three single-nucleotide polymorphisms of the angiotensinogen gene and susceptibility to hypertension: single locus genotype vs. haplotype analysis," *Physiological genomics*, vol. 17, pp. 79-86, 2004.
- [301] A. A. Motsinger and M. D. Ritchie, "Multifactor dimensionality reduction: an analysis strategy for modelling and detecting gene-gene interactions in human genetics and pharmacogenomics studies," *Human genomics*, vol. 2, p. 318, 2006.
- [302] J. H. Moore, J. C. Gilbert, C.-T. Tsai, F.-T. Chiang, T. Holden, N. Barney, and B. C. White, "A flexible computational framework for detecting, characterizing, and interpreting statistical

patterns of epistasis in genetic studies of human disease susceptibility," *Journal of theoretical biology*, vol. 241, pp. 252-261, 2006.

- [303] M. Clemons and P. Goss, "Estrogen and the risk of breast cancer," *N engl J med*, vol. 344, pp. 276-285, 2001.
- [304] J. A. Anderson, *An introduction to neural networks*: MIT press, 1995.
- [305] F. Lescai and C. Franceschi, "The impact of phenocopy on the genetic analysis of complex traits," *PloS one*, vol. 5, p. e11876, 2010.
- [306] J. L. Haines and M. A. Pericak-Vance, *Genetic analysis of complex disease*: John Wiley & Sons, 2006.
- [307] N. J. Schork, D. Fallin, B. Thiel, X. Xu, U. Broeckel, H. J. Jacob, and D. Cohen, "14 The future of genetic case-control studies," *Advances in genetics*, vol. 42, pp. 191-212, 2001.
- [308] W. D. Shannon, M. A. Province, and D. Rao, "Tree-based recursive partitioning methods for subdividing sibpairs into relatively more homogeneous subgroups," *Genetic epidemiology*, vol. 20, pp. 293-306, 2001.
- [309] R. A. King, J. I. Rotter, and A. G. Motulsky, *The genetic basis of common diseases*: Oxford University Press, 2002.
- [310] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436-444, 2015.
- [311] A. Candel, V. Parmar, E. LeDell, and A. Arora, "Deep Learning with H2O," 2015.
- [312] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097-1105.
- [313] J. J. Tompson, A. Jain, Y. LeCun, and C. Bregler, "Joint training of a convolutional network and a graphical model for human pose estimation," in *Advances in neural information processing systems*, 2014, pp. 1799-1807.
- [314] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, and T. N. Sainath, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *Signal Processing Magazine, IEEE*, vol. 29, pp. 82-97, 2012.
- [315] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, 2014, pp. 3104-3112.
- [316] M. Helmstaedter, K. L. Briggman, S. C. Turaga, V. Jain, H. S. Seung, and W. Denk, "Connectomic reconstruction of the inner plexiform layer in the mouse retina," *Nature*, vol. 500, pp. 168-174, 2013.
- [317] A. Bordes, S. Chopra, and J. Weston, "Question answering with subgraph embeddings," *arXiv preprint arXiv:1406.3676*, 2014.
- [318] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch," *The Journal of Machine Learning Research*, vol. 12, pp. 2493-2537, 2011.
- [319] S. Min, B. Lee, and S. Yoon, "Deep Learning in Bioinformatics," *arXiv preprint arXiv:1603.06430*, 2016.
- [320] D. S. Yeung, Cloete, I., Shi, D., Ng, W.W.Y., *Sensitivity Analysis for Neural Networks*: Springer-Verlag Berlin Heidelberg, 2010.

- [321] Y. Bengio, I. J. Goodfellow, and A. Courville, "Deep learning," *An MIT Press book in preparation. Draft chapters available at <http://www.iro.umontreal.ca/~bengioy/dlbook>*, 2015.
- [322] E. Beqari, "A Very Basic Introduction to Feed-Forward Neural Networks," 2018.
- [323] M. A. Nielsen, "Neural networks and deep learning (2015)," *Also available at: <http://neuralnetworksanddeeplearning.com>*, 2016.
- [324] S. Renals. (2014). *Multi-Layer Neural Networks*. Available: <https://www.inf.ed.ac.uk/teaching/courses/asr/2013-14/asr08a-nnDetails.pdf>
- [325] B. Recht, C. Re, S. Wright, and F. Niu, "Hogwild: A lock-free approach to parallelizing stochastic gradient descent," in *Advances in Neural Information Processing Systems*, 2011, pp. 693-701.
- [326] S. Aiello, T. Kraljevic, and P. Maj, "Package 'h2o,'" 2015.
- [327] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, pp. 1929-1958, 2014.
- [328] T. D. Gedeon, "Data mining of inputs: analysing magnitude and functional measures," *International Journal of Neural Systems*, vol. 8, pp. 209-218, 1997.
- [329] C. Arno, "The Definitive Performance Tuning Guide for H2O Deep Learning," <http://blog.h2o.ai/2015/02/deep-learning-performance/>, Ed., ed, February 26, 2015.
- [330] C. Angermueller, T. Pärnamaa, L. Parts, and O. Stegle, "Deep learning for computational biology," *Molecular systems biology*, vol. 12, 2016.
- [331] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, "On the importance of initialization and momentum in deep learning," in *International conference on machine learning*, 2013, pp. 1139-1147.
- [332] M. D. Zeiler, "ADADELTA: an adaptive learning rate method," *arXiv preprint arXiv:1212.5701*, 2012.
- [333] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *Journal of Machine Learning Research*, vol. 13, pp. 281-305, 2012.
- [334] Beckmw. (2013). *Sensitivity analysis for neural networks*. Available: <https://beckmw.wordpress.com/2013/10/07/sensitivity-analysis-for-neural-networks/>
- [335] S. Lek, A. Belaud, P. Baran, I. Dimopoulos, and M. Delacoste, "Role of some environmental variables in trout abundance models using neural networks," *Aquatic Living Resources*, vol. 9, pp. 23-29, 1996.
- [336] I. Jolliffe, *Principal component analysis*: Wiley Online Library, 2002.
- [337] J.-B. Chen, Y.-H. Yang, W.-C. Lee, C.-W. Liou, T.-K. Lin, Y.-H. Chung, L.-Y. Chuang, C.-H. Yang, and H.-W. Chang, "Sequence-based polymorphisms in the mitochondrial D-loop and potential SNP predictors for chronic dialysis," *PloS one*, vol. 7, p. e41125, 2012.
- [338] S. Glander. (2017). *Building deep neural nets with h2o and rsparkling that predict arrhythmia of the heart* Available: [https://shiring.github.io/machine\\_learning/2017/02/27/h2o](https://shiring.github.io/machine_learning/2017/02/27/h2o)
- [339] M. Beck, "NeuralNetTools: Visualization and Analysis Tools for Neural Networks," *R package version*, vol. 1, 2015.
- [340] D. Maji, A. Santara, S. Ghosh, D. Sheet, and P. Mitra, "Deep neural network and random forest hybrid architecture for learning to detect retinal vessels in fundus images," in

*Engineering in Medicine and Biology Society (EMBC), 2015 37th Annual International Conference of the IEEE, 2015*, pp. 3029-3032.

- [341] P. Kotschieder, M. Fiterau, A. Criminisi, and S. R. Bulo, "Deep neural decision forests," in *Computer Vision (ICCV), 2015 IEEE International Conference on*, 2015, pp. 1467-1475.
- [342] C. Strobl, A.-L. Boulesteix, A. Zeileis, and T. Hothorn, "Bias in random forest variable importance measures," in *Workshop on Statistical Modelling of Complex Systems*, 2006.
- [343] D. Falconer and T. Mackay, "Small populations: II Less simplified conditions," *Introduction to Quantitative Genetics*. Addison Wesley Longman Limited: Edinburgn Gate, Harlow Essex, UK, pp. 76-78, 1996.
- [344] R. Culverhouse, B. K. Suarez, J. Lin, and T. Reich, "A perspective on epistasis: limits of models displaying no main effect," *The American Journal of Human Genetics*, vol. 70, pp. 461-471, 2002.
- [345] R. J. Urbanowicz, J. Kiralis, J. M. Fisher, and J. H. Moore, "Predicting the difficulty of pure, strict, epistatic models: metrics for simulated model selection," *BioData mining*, vol. 5, p. 15, 2012.
- [346] M. H. Wang, R. Sun, J. Guo, H. Weng, J. Lee, I. Hu, P. C. Sham, and B. C.-Y. Zee, "A fast and powerful W-test for pairwise epistasis testing," *Nucleic acids research*, vol. 44, pp. e115-e115, 2016.
- [347] P. G. Ferrario and I. R. König, "Transferring entropy to the realm of GxG interactions," *Briefings in bioinformatics*, p. bbw086, 2016.
- [348] C. Busch and R. Hegele, "Genetic determinants of type 2 diabetes mellitus," *Clinical genetics*, vol. 60, pp. 243-254, 2001.
- [349] A. L. Price, N. A. Zaitlen, D. Reich, and N. Patterson, "New approaches to population stratification in genome-wide association studies," *Nature Reviews Genetics*, vol. 11, p. 459, 2010.
- [350] S. Bhattacharjee, Z. Wang, J. Ciampa, P. Kraft, S. Chanock, K. Yu, and N. Chatterjee, "Using principal components of genetic variation for robust and powerful detection of gene-gene interactions in case-control and case-only studies," *The American Journal of Human Genetics*, vol. 86, pp. 331-342, 2010.
- [351] A. Niu, S. Zhang, and Q. Sha, "A Novel Method to Detect Gene–Gene Interactions in Structured Populations: MDR-SP," *Annals of human genetics*, vol. 75, pp. 742-754, 2011.
- [352] A. Romero, P. L. Carrier, A. Erraqabi, T. Sylvain, A. Auvolat, E. Dejolie, M.-A. Legault, M.-P. Dubé, J. G. Hussin, and Y. Bengio, "Diet networks: Thin parameters for fat genomic," *arXiv preprint arXiv:1611.09340*, 2016.
- [353] M. D. Ritchie, B. C. White, J. S. Parker, L. W. Hahn, and J. H. Moore, "Optimizationof neural network architecture using genetic programming improvesdetection and modeling of gene-gene interactions in studies of humandiseases," *BMC bioinformatics*, vol. 4, p. 28, 2003.

# Glossary

**Allele** is a variant form of a gene located at a specific position on a Chromosome.

**Area under the curve (AUC)** is a measure that determines the performance of the models across all the possible classification thresholds.

**Bonferroni correction** is a method performed to reduce type I errors by correcting individual p values in multiple statistical testing.

**Bootstrap samples** are smaller random samples that are obtained from the original data with replacement.

**Chi-square test** is a statistical distribution used to determine the significant difference between expected and observed frequencies when a null hypothesis is true.

**Cross-Validation** is a resampling technique that divides original data into smaller subsets to validate the models. One subset is used for testing and remaining subsets are used for training the machine learning models. Multiple rounds of testing and training are performed with different partitions of the data to reduce the variability. Finally, the validation results over the multiple rounds are averaged.

**Data mining** is the process of finding hidden patterns and anomalies from large data to obtain potentially useful information.

**Deoxyribonucleic acid (DNA)** is a double helix macromolecule united with hydrogen bonds, which comprise of instructions for development of an organism.

**Epistasis** is defined as the interactions between set of genes and their associations with a phenotype. Initial studies defined epistasis as a SNP or a gene at a locus suppresses the expression of another SNP or gene at different locus.

**Estrogen** is a hormone secreted for the growth, development, sexual maturity and reproductive ability of a female.

**F1** generation is the first set of offspring produced by a set of parents.

**F2** generation is the offspring of a cross between two F1 individuals.

**Genetic heterogeneity** is a phenomenon in which same or similar phenotype may be caused by mutation of genes at different loci or different genetic mechanism.

**Genome-wide association study (GWAS)** is the study of genetic variations in different individuals with the intention of revealing the associations with a trait or a disease.

**Genotype** is the genetic composition of an individual organism. A gene for a particular character or trait exists in two allelic forms (for example: A and a). The possible

genotype combinations are AA (homozygous dominant), Aa (heterozygous), and aa (homozygous recessive).

**Heritability** is the observed difference on a trait among individuals of a population due to genetic variation.

**High-dimensional data** contain information on large number of variables. In genetic data, the number of variables can exceed the number of samples making extremely difficult to analyse.

**Hyperplane** is a subspace of one dimension less than its ambient space.

**Linkage disequilibrium (LD)** is an association between two nearby loci. Two loci are in LD if the probability of recombination is less than 0.5 and statistically related linking between two loci.

**Locus** is a specific location of a gene on a chromosome. Plural of locus is loci.

**Logarithmic loss (Logloss)** or cross entropy loss measures the performance of the models by minimizing the uncertainty between actual and predicted class labels.

**Machine learning** is the ability of an algorithm to learn from experience. It automatically produces models with rules and patterns from the data.

**Marginal effect** measures the impact of change in dependent variable when a independent variable is changed.

**Minor allele frequency (MAF)** is the frequency at which the least common allele occurs in a given population.

**Missing heritability** is the fact that single genetic variants cannot explain the heritability of trait or complex diseases.

**Null hypothesis ( $H_0$ )** assumes no associations between groups of the population. Test statistics (p-value) is used to determine whether to reject or accept the null hypothesis.

**Out-of-bag (OOB) error** is an unbiased estimation of the test set errors. It is executed internally during run to construct accurate trees.

**Over-fitting** occurs when a statistical function is closely fitted to a set of data that reduces the performance of the model.

**Penetrance** is the proportion of a population with a given genotype that displays the phenotype

**Phenocopy** is a variation in a phenotype due to environmental factors, such that the phenotype matches with the phenotype determined by genetic factors. These changes in phenotype are not heritable and are not due to mutations in DNA sequence.

**Phenotype** is the physical appearance of an organism with respect to a trait allele

**Polygenic inheritance** or multifactorial inheritance refers to a trait or phenocopy that is influenced by multiple genes.

**Population stratification** is the systematic ancestry differences in allele frequencies that present between sub-population of a population.

**Power** of the method is the number of times the known SNP interactions are identified correctly.

**P-value** is the probability of a statistical model when the null hypothesis is true. It determines the statistical significance of the results. Large p-values indicate the weak evidence, where else small p-values indicate the strong evidence to reject the null hypothesis.

**Random sampling** is a method of selecting a subset of random samples from a statistical population in which each number of the subset has an equal probability of being chosen.

**Sensitivity** is a statistical measure that identifies the proportion of actual / true positives to total number of cases in the population.

**Single nucleotide polymorphism (SNP)** is a common genetic variation occurs due to a change in a single nucleotide (A, T, C and G) at a specific location of the DNA sequence.

**Specificity** is a statistical measure that identifies the proportion of actual / true negatives to total number of controls in the population.

**Threshold** value is a proportion that determines disease risk status to a specific multi-locus genotype combination.

**Type I error** is the probability of rejecting a true null hypothesis (false positives).

**Type II error** is the probability of failing to reject a false null hypothesis (false negatives).

## **Appendix**

### **Statement of Contribution by Others**

To Whom It May Concern, I, Suneetha Uppu, contributed to the theoretical development of the method, implementation, evaluation and preparing the manuscript entitled:

**S. Uppu**, A. Krishna, and "A deep hybrid model to detect multi-locus interacting SNPs in the presence of noise", *International Journal of Medical Informatics*, vol. 119, pp. 134-151, 2018.

The co-author contributed in editing the manuscript, interpreting the results and revising it critically for important intellectual content.

---

Suneetha Uppu

I, as a Co-Author, endorse that this level of contribution by the candidate indicated above is appropriate.

---

Aneesh Krishna

## Statement of Contribution by Others

To Whom It May Concern, I, Suneetha Uppu, contributed in reviewing the current literature and preparing the manuscript entitled:

**S. Uppu**, A. Krishna, and R. Gopalan, "A review on methods for detecting SNP interactions in high-dimensional genomic data", *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 15 (2), pp. 599 - 612, 2018.

The co-authors contributed in editing the manuscript and revising it critically for important intellectual content.

---

Suneetha Uppu

I, as a Co-Author, endorse that this level of contribution by the candidate indicated above is appropriate.

---

Aneesh Krishna

---

Raj P.Gopalan

## Statement of Contribution by Others

To Whom It May Concern, I, Suneetha Uppu, contributed to the theoretical development of the method, implementation, evaluation and preparing the manuscript entitled:

**S. Uppu**, A. Krishna, and R. Gopalan, "Combining associative classification with multifactor dimensionality reduction for predicting higher-order SNP interactions in case-control studies", *International Journal of Computational Intelligence in Bioinformatics and Systems Biology*, 22nd October 2017.

The co-authors contributed in formulating the theoretical development of the method, interpreting the results, editing the manuscript and revising it critically for important intellectual content.

---

Suneetha Uppu

I, as a Co-Author, endorse that this level of contribution by the candidate indicated above is appropriate.

---

Aneesh Krishna

---

Raj P.Gopalan

## Statement of Contribution by Others

To Whom It May Concern, I, Suneetha Uppu, contributed to the theoretical development of the method, implementation, evaluation and preparing the manuscript entitled:

**S. Uppu**, A. Krishna, and R. P. Gopalan, "A Deep Learning Approach to Detect SNP Interactions", *Journal of Software (JSW)*, vol. 11, pp. 965-975, 2016.

The co-author contributed in editing the manuscript, interpreting the results and revising it critically for important intellectual content.

---

Suneetha Uppu

I, as a Co-Author, endorse that this level of contribution by the candidate indicated above is appropriate.

---

Aneesh Krishna

## Statement of Contribution by Others

To Whom It May Concern, I, Suneetha Uppu, contributed to the theoretical development of the method, implementation, evaluation and preparing the manuscript entitled:

**S. Uppu** and A. Krishna, "Evaluation of associative classification-based multifactor dimensionality reduction in the presence of noise", *Network Modeling Analysis in Health Informatics and Bioinformatics*, vol. 5, pp. 1-9, 2016.

The co-author contributed in interpreting the results, editing the manuscript and revising it critically for important intellectual content.

---

Suneetha Uppu

I, as a Co-Author, endorse that this level of contribution by the candidate indicated above is appropriate.

---

Aneesh Krishna

## Statement of Contribution by Others

To Whom It May Concern, I, Suneetha Uppu, contributed to the theoretical development of the method, implementation, evaluation and preparing the manuscript entitled:

**S. Uppu**, A. Krishna, and R. P. Gopalan, "Rule-based analysis for detecting epistasis using associative classification mining", *Network Modeling Analysis in Health Informatics and Bioinformatics*, vol. 4, pp. 1-19, 2015.

The co-authors contributed in formulating the theoretical development of the method, interpreting the results, editing the manuscript and revising it critically for important intellectual content.

---

Suneetha Uppu

I, as a Co-Author, endorse that this level of contribution by the candidate indicated above is appropriate.

---

Aneesh Krishna

---

Raj P.Gopalan

## Statement of Contribution by Others

To Whom It May Concern, I, Suneetha Uppu, contributed to the theoretical development of the method, implementation, evaluation and preparing the manuscript entitled:

**S. Uppu**, A. Krishna, and R. P. Gopalan, "Detecting SNP Interactions in Balanced and Imbalanced Datasets using Associative Classification", *Australian Journal of Intelligent Information Processing Systems*, vol. 14, 2014.

The co-authors contributed in formulating the theoretical development of the method, interpreting the results, editing the manuscript and revising it critically for important intellectual content.

---

Suneetha Uppu

I, as a Co-Author, endorse that this level of contribution by the candidate indicated above is appropriate.

---

Aneesh Krishna

---

Raj P.Gopalan

## Statement of Contribution by Others

To Whom It May Concern, I, Suneetha Uppu, contributed to the theoretical development of the method, implementation, evaluation and preparing the paper entitled:

**S. Uppu** and A. Krishna, "An intensive search for higher-order gene-gene interactions by improving deep learning model", in *18th IEEE International Conference on Bioinformatics and Bioengineering (BIBE)*, 2018, pp.104-109.

The co-author contributed in editing the manuscript, interpreting the results and revising it critically for important intellectual content.

---

Suneetha Uppu

I, as a Co-Author, endorse that this level of contribution by the candidate indicated above is appropriate.

---

Aneesh Krishna

## Statement of Contribution by Others

To Whom It May Concern, I, Suneetha Uppu, contributed to the theoretical development of the method, implementation, evaluation and preparing the paper entitled:

**S. Uppu** and A. Krishna, "Tuning Hyper-parameters for Gene Interaction Models in Genome-Wide Association Studies", in *International Conference on Neural Information Processing*, 2017, pp. 791-801.

The co-author contributed in editing the manuscript, interpreting the results and revising it critically for important intellectual content.

---

Suneetha Uppu

I, as a Co-Author, endorse that this level of contribution by the candidate indicated above is appropriate.

---

Aneesh Krishna

## Statement of Contribution by Others

To Whom It May Concern, I, Suneetha Uppu, contributed to the theoretical development of the method, implementation, evaluation and preparing the paper entitled:

**S. Uppu** and A. Krishna, "Improving strategy for discovering interacting genetic variants in association studies," in *International Conference on Neural Information Processing*, 2016, pp. 461-469.

The co-author contributed in editing the manuscript, interpreting the results and revising it critically for important intellectual content.

---

Suneetha Uppu

I, as a Co-Author, endorse that this level of contribution by the candidate indicated above is appropriate.

---

Aneesh Krishna

## Statement of Contribution by Others

To Whom It May Concern, I, Suneetha Uppu, contributed to the theoretical development of the method, implementation, evaluation and preparing the paper entitled:

**S. Uppu**, A. Krishna, and R. P. Gopalan, "Towards Deep Learning in genome-Wide Association Interaction studies", in *Pacific Asia Conference on Information Systems (PACIS)*, 2016, p. 20.

The co-authors contributed in editing the manuscript, interpreting the results and revising it critically for important intellectual content.

---

Suneetha Uppu

I, as a Co-Author, endorse that this level of contribution by the candidate indicated above is appropriate.

---

Aneesh Krishna

---

Raj P.Gopalan

## Statement of Contribution by Others

To Whom It May Concern, I, Suneetha Uppu, contributed to the theoretical development of the method, implementation, evaluation and preparing the paper entitled:

**S. Uppu**, A. Krishna, and R. P. Gopalan, "A Multifactor Dimensionality Reduction Based Associative Classification for Detecting SNP Interactions", in *Neural Information Processing*, 2015, pp. 328-336.

The co-authors contributed in formulating the theoretical development of the method, editing the manuscript, interpreting the results and revising it critically for important intellectual content.

---

Suneetha Uppu

I, as a Co-Author, endorse that this level of contribution by the candidate indicated above is appropriate.

---

Aneesh Krishna

---

Raj P.Gopalan

## Statement of Contribution by Others

To Whom It May Concern, I, Suneetha Uppu, contributed to the theoretical development of the method, implementation, evaluation and preparing the paper entitled:

**S. Uppu**, A. Krishna, and R. P. Gopalan, "An Associative Classification Based Approach for Detecting SNP-SNP Interactions in High-dimensional Genome," in *IEEE International Conference on Bioinformatics and Bioengineering (BIBE)*, 2014, pp. 329-333.

The co-authors contributed in formulating the theoretical development of the method, editing the manuscript, interpreting the results and revising it critically for important intellectual content.

---

Suneetha Uppu

I, as a Co-Author, endorse that this level of contribution by the candidate indicated above is appropriate.

---

Aneesh Krishna

---

Raj P.Gopalan

## Statement of Contribution by Others

To Whom It May Concern, I, Suneetha Uppu, contributed to the theoretical development of the method, implementation, evaluation and preparing the paper entitled:

**S. Uppu** and A. Krishna, "Convolutional model for predicting gene-gene interactions," in *25th International Conference on Neural Information Processing (ICONIP)*, pp.127-137, 2018.

The co-author contributed in editing the manuscript, interpreting the results and revising it critically for important intellectual content.

---

Suneetha Uppu

I, as a Co-Author, endorse that this level of contribution by the candidate indicated above is appropriate.

---

Aneesh Krishna