Edith Cowan University
Research Online

Theses: Doctorates and Masters | Theses

2014

Evolutionary approaches for feature selection in biological data

Vinh Q. Dang
Edith Cowan University

Recommended Citation
Dang, V. Q. (2014). *Evolutionary approaches for feature selection in biological data*. Retrieved from https://ro.ecu.edu.au/theses/1276

This Thesis is posted at Research Online.
https://ro.ecu.edu.au/theses/1276

Edith Cowan University
Research Online

Theses: Doctorates and Masters

Theses

2014

Evolutionary approaches for feature selection in biological data

Vinh Q. Dang
Edith Cowan University

Recommended Citation
Dang, V. Q. (2014). *Evolutionary approaches for feature selection in biological data*. Retrieved from http://ro.ecu.edu.au/theses/1276

This Thesis is posted at Research Online.
http://ro.ecu.edu.au/theses/1276

USE OF THESIS

The Use of Thesis statement is not included in this version of the thesis.

Edith Cowan University

Copyright Warning

You may print or download ONE copy of this document for the purpose of your own research or study.

The University does not authorize you to copy, communicate or otherwise make available electronically to any other person any copyright material contained on this site.

You are reminded of the following:

- Copyright owners are entitled to take legal action against persons who infringe their copyright.

- A reproduction of material that is protected by copyright may be a copyright infringement. Where the reproduction of such material is done without attribution of authorship, with false attribution of authorship or the authorship is treated in a derogatory manner, this may be a breach of the author’s moral rights contained in Part IX of the Copyright Act 1968 (Cth).

- Courts have the power to impose a wide range of civil and criminal sanctions for infringement of copyright, infringement of moral rights and other offences under the Copyright Act 1968 (Cth). Higher penalties may apply, and higher damages may be awarded, for offences and infringements involving the conversion of material into digital or electronic form.

Evolutionary Approaches for Feature Selection in Biological Data

A dissertation submitted in partial fulfilment of the requirements for the degree of
Doctor of Philosophy of Computer Science

By: Vinh Quoc Dang
Bachelor of Computer Science (Honours)

Faculty of Health, Engineering and Science
Edith Cowan University, Perth, Western Australia

Principal Supervisor: Associate Professor Chiou Peng Lam
Associate supervisor: Dr. Simon Laws & Professor Ralph Martins

Date of submission
January 2014

Abstract

Data mining techniques have been used widely in many areas such as business, science, engineering and medicine. The techniques allow a vast amount of data to be explored in order to extract useful information from the data. One of the foci in the health area is finding interesting biomarkers from biomedical data. Mass throughput data generated from microarrays and mass spectrometry from biological samples are high dimensional and is small in sample size. Examples include DNA microarray datasets with up to 500,000 genes and mass spectrometry data with 300,000 m/z values. While the availability of such datasets can aid in the development of techniques/drugs to improve diagnosis and treatment of diseases, a major challenge involves its analysis to extract useful and meaningful information. The aims of this project are: 1) to investigate and develop feature selection algorithms that incorporate various evolutionary strategies, 2) using the developed algorithms to find the “most relevant” biomarkers contained in biological datasets and 3) and evaluate the goodness of extracted feature subsets for relevance (examined in terms of existing biomedical domain knowledge and from classification accuracy obtained using different classifiers). The project aims to generate good predictive models for classifying diseased samples from control.

Copyright and access declaration

I certify that this thesis does not, to the best of my knowledge and belief:

i. incorporate without acknowledgment any material previously submitted for a degree or diploma in any institution of higher education;

ii. contain any material previously published or written by another person except where due reference is made in the text; or

iii. contain any defamatory material.

Dated....28/5/2014..........

Acknowledgements

PhD Thesis submission is a special milestone achievement in the life of higher degree research students. This is not only the time to indicate the journey of the thesis is about to finish but also the time to express my thanks and appreciations towards people who greatly influenced my research.

First of all, special thanks to my principal supervisor, Associate Professor Chiou Peng Lam for her dedication in guiding and supporting me throughout the course of research. I am so grateful and proud to be guided under her supervision. I have learnt so much, not only research skills to complete the thesis but also teamwork skills in solving problems and finding optimal solutions. I really appreciate the time and effort she gave to my thesis, as well as her sharing of knowledge in the area of data mining and bioinformatics. Without her tremendous help and encouragement, the thesis would not have been completed as it has been.

I would like to thank my associate supervisors, Professor Ralph Martins and Dr. Simon Laws for their support and sharing their expertise in the area of Alzheimer's disease specifically and bioinformatics in general. Special thanks also goes to Dr. Chang Su Lee for his support and advice during his time as part of my supervision team. I also would like to thank other research students for their friendliness, support and encouragement towards me. It is a pleasure to work with them and to share experiences in the research and in the student life during the time of the course. I am so grateful for the support of my family. Especially I thank my wife and children for their patience, understanding, support and care for me over the years of my study. They also gave me the strength to go through the tough time during the study and the joy to complete the thesis.

List of Publications

Dang, V. Q., Lam, C.-P., & Lee, C. S. (2011). *Incorporating genetic algorithm into rough feature selection for high dimensional biomedical data*. Paper presented at the 3rd International Symposium on IT in Medicine and Education *(ITME),* Guangzhou, China. (Includes work described in Chapter 4)

Dang, V. Q., Lam, C.-P, & Lee, C. S. (2013). *NSC-GA: Search for optimal shrinkage thresholds for nearest shrunken centroid.* Paper presented at the 10th annual IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB), Singapore. (Includes work described in Chapter 5)

Dang, V. Q. & Lam, C.-P. (2013). *NSC-NSGA2: Optimal Search for Finding Multiple Thresholds for Nearest Shrunken Centroid*. Paper presented at the IEEE International Conference on Bioinformatics and Biomedicine (BIBM). Shanghai, China. (Includes work described in Chapter 8) [19.6% acceptance rate]

Table of Contents

Abstract ... iii

Copyright and access declaration ... v

Acknowledgements ... vii

List of Publications ... ix

Table of Figures ... xvii

Table of Tables ... xxiii

Table of Abbreviations ... xxix

1. Introduction ... 1

1.1. Statement of the Problem ... 2

1.2. The purpose of the study ... 7

1.3. The contributions of this study ... 7

1.4. Significance ... 12

1.5. Structure of the thesis ... 13

1.6. Summary ... 15

2. Literature review ... 16

2.1. Data mining and feature selection techniques ... 16

2.1.1. Data mining ... 16

2.1.1.1. Classification ... 17

2.1.1.2. Feature selection ... 19

2.2. Feature selection approaches have been developed in the area of bioinformatics ... 22

2.2.1. Feature selection using EAs ... 22

2.2.1.1. Genetic algorithm ... 22

2.2.1.2. Multi-objective evolutionary algorithms ... 24

2.2.1.3. Memetic algorithms (MAs) ... 25

2.2.2. Feature selection using RST ... 25

2.2.3. Feature selection using NSC ... 26

2.2.4. Feature selection using hybrid approach ... 29

2.3. Techniques related to the implementations of proposed approaches in the study for FS and classification ... 31

2.3.1. K-Means ... 31

2.3.2. Rough Set Theory ... 32

2.3.3. Nearest Shrunken Centroid algorithm 34

2.3.4. Evolutionary Algorithm 38

2.3.4.1. Genetic Algorithm 38

2.3.4.2. Multi-objective evolutionary algorithms 43

2.3.5. Memetic Algorithms (MAs) 44

2.3.6. Similarity distance measures 49

2.3.6.1. Euclidean distance 49

2.3.6.2. Mahalanobis distance 50

2.3.6.3. Pearson distance 51

2.3.6.4. Mass distance 52

2.4. Discussion and Summary 53

3. Datasets, Evaluation strategy, Convergence and Termination criteria 55

3.1. Datasets 55

3.1.1. Ray *et al*. Alzheimer's Disease (AD) datasets 56

3.1.2. Alon *et al*. Colon cancer data 58

3.1.3. Leukemia cancer data 59

3.1.4. Lung cancer data 59

3.1.5. Lymphoma cancer data 60

3.1.6. Prostate cancer data 61

3.1.7. Ovarian cancer data 61

3.2. Evaluation Strategy 62

3.3. Termination criteria 66

3.4. Genetic Algorithm and state of convergence 68

3.5. Summary 69

4. Rough set theory and GA approach (RST-GA) 71

4.1. The proposed approach, RST-GA 72

4.1.1. Phase 1: Feature reduction 73

4.1.2. Phase 2: Generate a distinction table 80

4.1.3. Phase 3: Feature selection via GA search optimization 82

4.1.4. Parameter tuning for the GA 87

4.2. Experiment results 88

4.2.1. Alon *et al*. Colon cancer data 88

4.2.2. Leukemia cancer data 98

4.3. Discussion 104

4.4. Summary 105

5. Incorporating NSC and GA, NSC-GA 107

5.1. Introduction 107

5.2. The proposed approach, NSC-GA 110

5.2.1. Issues related to the proposed approach, NSC-GA 112

5.2.1.1. Encoding chromosomes 112

5.2.1.2. Estimate initial range of values for shrinkage thresholds 112

5.2.1.3. Fitness evaluation using NSC as a fitness evaluator 115

5.2.2. Steps in the proposed approach, NSC-GA 115

5.2.2.1. Step 1: Th_{max} calculation 116

5.2.2.2. Step 2: GA search optimization 116

5.3. Parameter settings for GA 122

5.3.1. Parameter settings 123

5.3.2. Results of parameter tuning 125

5.3.2.1. Ray *et al.* AD data 125

5.3.2.2. Alon *et al.* Colon cancer data 128

5.3.2.3. Leukemia cancer data 129

5.3.2.4. Lung cancer data 130

5.3.2.5. Lymphoma cancer data 131

5.3.2.6. Ovarian cancer data 132

5.3.2.7. Prostate cancer data 133

5.4. Experiment results 136

5.4.1. Ray *et al.* Alzheimer's Disease (AD) data 137

5.4.2. Alon *et al.* Colon cancer data 139

5.4.3. Leukemia cancer data 141

5.4.4. Ovarian cancer data 144

5.4.5. Lymphoma cancer data 145

5.4.6. Lung cancer data 147

5.4.7. Prostate cancer data 149

5.5. Discussion 153

5.6. Summary 156

6. Incorporating the NSC algorithm into MA 157

6.1. The proposed approach, NSC-MA 158

6.1.1. NSC-MA proposed approach 159

6.1.2. Steps of the proposed approach, NSC-MA 161

6.1.3. Parameter settings 164

6.2. Experiment results 164

6.2.1. Ray *et al.* Alzheimer's Disease data 165

6.2.2. Alon *et al.* Colon cancer data 167

6.2.3. Leukemia cancer data 169

6.2.4. Ovarian cancer data 170

6.2.5. Lymphoma cancer data 172

6.2.6. Lung cancer data 173

6.2.7. Prostate cancer data 175

6.3. Discussion 178

6.4. Summary 178

7. Incorporating different similarity distance measures into the NSC algorithm in the NSC-GA approach 180

7.1. NSC*-GA proposed approach 180

7.1.1. NSC-GA with Mahalanobis (NSC$_M$-GA), Pearson (NSC$_P$-GA) and Mass distance measure (NSC$_{MD}$-GA) 181

7.2. Experiment results 184

7.2.1. NSC$_M$-GA 185

7.2.1.1. Ray *et al.* AD data 185

7.2.1.2. Alon *et al.* Colon cancer data 186

7.2.1.3. Leukemia cancer data 187

7.2.1.4. Lymphoma cancer data 188

7.2.1.5. Lung cancer data 190

7.2.1.6. Ovarian cancer data 191

7.2.1.7. Prostate cancer data 192

7.2.2. NSC$_P$-GA approach 194

7.2.2.1. Ray *et al.* AD data 194

7.2.2.2. Alon *et al.* Colon cancer data 195

7.2.2.3. Leukemia cancer data 196

7.2.2.4. Lymphoma cancer data 197

7.2.2.5. Lung cancer data 198

7.2.2.6. Ovarian cancer data 200

7.2.2.7. Prostate cancer data 201

7.2.3. NSC$_{MD}$-GA approach 202

7.2.3.1. Ray *et al.* AD data 202

7.2.3.2. Alon *et al.* Colon cancer data 204

7.2.3.3. Leukemia cancer data 205

7.2.3.4. Lymphoma cancer data 206

7.2.3.5. Lung cancer data 207

7.2.3.6. Ovarian cancer data 208
7.2.3.7. Prostate cancer data 209
7.2.4. Summary: selected feature subsets and corresponding classification results 210
7.3. Summary 219
8. Incorporating Nearest shrunken centroid and multi-objective evolutionary algorithm for searching multiple shrinkage threshold solutions 221
8.1. Introduction 221
8.2. The proposed approach, NSC-NSGA2 222
8.2.1. Issues related to the proposed approach, NSC-NSGA2........ 224
8.2.1.1. Encoding chromosomes........ 224
8.2.1.2. Fitness evaluation using NSC as a fitness evaluator........ 224
8.2.2. Steps of the proposed approach, NSC-NSGA2 225
8.2.2.1. Step 1: $\mathrm{Th}_{\max}$ calculation........ 225
8.2.2.2. Step 2: Multi-objective evolutionary algorithm search optimization........ 225
8.2.3. Parameter settings for NSGA2........ 234
8.3. Experiment results 235
8.3.1. Ray *et al.* AD data........ 236
8.3.2. Alon *et al.* Colon cancer data........ 239
8.3.3. Leukemia cancer data........ 241
8.3.4. Ovarian cancer data........ 243
Using the same experimental procedure as before, the proposed algorithm was evaluated using the Ovarian cancer dataset. A typical convergence plot of Pareto optimal front with 5 solutions is shown in Figure 8-14. 243
8.3.5. Lymphoma cancer data 245
8.3.6. Lung cancer data 247
Using the same experimental procedure as before, the proposed was evaluated using the Lung cancer dataset. A typical convergence plot of Pareto optimal front with 3 solutions is shown in Figure 8-16. 247
8.3.7. Prostate cancer data........ 249
8.4. NSC-NSGA2 with 3 objective functions........ 254
8.5. The proposed approach, NSC-NSGA2 with Mahalanobis distance measure.. 256
8.6. Summary 258
9. Conclusion and Future work........ 260
9.1. Conclusion 260

9.2. Future work 264

Table of Figures

Figure 2-1 A general process of supervised learning for classification 17

Figure 2-2 Steps in a FS process 20

Figure 2-3 Basic k-means algorithm 32

Figure 2-4 Rough set with lower and upper approximation of a given class, C, adapted from Han, Kamber and Pei (2006, p. 352) and Hu *et al*. (2008, p. 3582)..... 33

Figure 2-5 Steps of the NSC algorithm 37

Figure 2-6 Basic steps of GA 38

Figure 2-7 Tournament selection procedure 40

Figure 2-8 Example of a one point binary crossover 41

Figure 2-9 Bit flip mutation procedure 42

Figure 2-10 Uniform mutation procedure 42

Figure 2-11 An example of LS using pair-wise strategy 46

Figure 2-12 Pair-wise LS with i*mprovement first strategy* used in Elbeltagi *et al*. (2005) 47

Figure 2-13 Procedure of greedy search strategy using *adding and subtracting* LS (Elbeltagi *et al*., 2005) 48

Figure 2-14 Mahalanobis distance measure 50

Figure 3-1 General mining structure: the breakdown of a dataset into an unseen test dataset (U) (brown colour) and a training dataset (T) (green colour), in which (T) is further split into 10 folds for 10 fold CV. 64

Figure 3-2 Termination procedure for GA 67

Figure 3-3 An example of a convergence status plot 69

Figure 4-1 Framework of the proposed approach, RST-GA.. 72

Figure 4-2 Example of part of normalized data using min-max normalization 75

Figure 4-3 Example of partial table of Th_l, Th_m andTh_u threshold values generated as initial starting points for k-means using the quartile statistics method. 77

Figure 4-4 Example of cluster centroid positions before and after k-means 78

Figure 4-5 Example of a "*01**" table 79

Figure 4-6 An example of d-*Distinction* table 81

Figure 4-7 Algorithm for initialisation of population using *d-Distinction* table............ 83

Figure 4-8 Algorithm for fitness calculation using f_1, f_2 and f objective functions........ 85

Figure 4-9 Algorithm for generating a new population 86

Figure 4-10 A typical convergence plot for maximum fitness and ($f_{max} - f_{avg}$) associated with the Colon dataset..89

Figure 4-11 A whisker plot for maximum fitness for 15 runs 90

Figure 4-12 A typical state of convergence plot for maximum fitness and ($f_{max} - f_{avg}$) values associated with the Leukemia cancer dataset 98

Figure 5-1 Shrinkage of threshold of 2 class centroid toward overall centroid in NSC108

Figure 5-2 Framework of the proposed approach, NSC-GA 111

Figure 5-3 Example of convergence plot for AD training dataset with the application of Th$_{max}$ calculation using NSC-GA...113

Figure 5-4 Example of convergence plot for AD training dataset without the application of Th$_{max}$ calculation and Th$_{upper}$ > Th$_{max}$ 113

Figure 5-5 Example of convergence plot for AD training dataset without the application of Th$_{max}$ calculation and the value of Th$_{upper}$ < Th$_{max}$ 114

Figure 5-6 Algorithm for calculating Th$_{max}$...116

Figure 5-7 Initial population algorithm using RNG...117

Figure 5-8 An example of an initial population..118

Figure 5-9 An example of creating a new population..121

Figure 5-10 Typical convergence of fitness plots for AD data associated with each of the 4 different parameter settings from DeJong, Genfenstette, Goldberg and Alander...125

Figure 5-11 An example plot of the state of convergence: ($f_{max} - f_{avg}$) versus generations (using Goldberg's set of parameter settings)...........................126

Figure 5-12 An example plot of the state of convergence: ($f_{max} - f_{avg}$) versus generations for 4 sets of parameter settings 127

Figure 5-13 Typical convergence of fitness plots for Colon cancer data associated with each of the 4 different parameter settings from DeJong, Grefenstette, Goldberg and Alander. 128

Figure 5-14 Typical convergence of fitness plots for Leukemia cancer data associated with parameter settings from DeJong, Grefenstette, Goldberg and Alander 129

Figure 5-15 Typical convergence of fitness plots for Lung cancer data with 4 different parameter settings from DeJong, Grefenstette, Goldberg and Alander 130

Figure 5-16 Typical convergence of fitness plots for Lymphoma cancer data for each of the 4 sets of parameter settings: DeJong, Grefenstette, Goldberg and Alander. 131

Figure 5-17 Typical convergence of fitness plots for Ovarian cancer data using each of the 4 different parameter settings from DeJong, Grefenstette, Goldberg and Alander 132

Figure 5-18 Typical convergence of fitness plots for Prostate cancer data with 4 different parameter settings from DeJong, Grefenstette, Goldberg and Alander 133

Figure 5-19 A typical convergence of fitness plot for the training set of AD 138

Figure 5-20 A typical convergence plot for the training set of Colon cancer data 140

Figure 5-21 A typical convergence plot for the training set of Leukemia cancer data. 142

Figure 5-22 A typical convergence plot for the training set of Ovarian cancer data 144

Figure 5-23 A typical convergence plot for the training set of Lymphoma cancer data 146

Figure 5-24 A typical convergence plot for the training set of Lung cancer data 148

Figure 5-25 A typical convergence plot for the training set of Prostate cancer data 150

Figure 6-1 Framework of the proposed approach, NSC-MA, using MA with adding and subtracting *Improvement First Strategy* LS 160

Figure 6-2 Procedure of *adding and subtracting LS with Improvement First Strategy* 162

Figure 6-3 Algorithm for generating a new population using MA incorporated *adding and subtracting LS with Improvement First Strategy* 163

Figure 6-4 A comparison convergence of fitness plot for AD training dataset using NSC-MA and NSC-GA associated with one typical run 165

Figure 6-5 Plots for convergence of fitness from a typical run for Colon training dataset using NSC-MA and NSC-GA 167

Figure 6-6 Plots for convergence of fitness from a typical run for Leukemia training dataset using NSC-MA and NSC-GA 169

Figure 6-7 Plots for convergence of fitness from a typical run for Ovarian training dataset using NSC-MA and NSC-GA 170

Figure 6-8 Plots for convergence of fitness from a typical run for Lymphoma training dataset using NSC-MA and NSC-GA 172

Figure 6-9 Plots for convergence of fitness from a typical run for Lung training dataset using NSC-MA and NSC-GA 174

Figure 6-10 A comparison convergence of fitness plot for Prostate training dataset using NSC-MA and NSC-GA associated with one typical run 175

Figure 7-1 Framework of NSC$_*$-GA 181

Figure 7-2 A typical plot for convergence of fitness for the training data of AD using NSC$_M$-GA 185

Figure 7-3 A typical plot for convergence of fitness for the training data of Colon using NSC$_M$-GA 186

Figure 7-4 A typical plot for convergence of fitness for the training data of Leukemia using NSC$_M$-GA 187

Figure 7-5 A typical plot for convergence of fitness for the training data of Lymphoma using NSC$_M$-GA 189

Figure 7-6 A typical plot for convergence of fitness for the training data of Lung using NSC$_M$-GA 190

Figure 7-7 A typical plot for convergence of fitness for the training data of Ovarian using NSC$_M$-GA 191

Figure 7-8 A typical plot for convergence of fitness for the training data of Prostate using NSC$_M$-GA 193

Figure 7-9 A typical plot for convergence of fitness for the training data of AD using NSC$_P$-GA 194

Figure 7-10 A typical plot for convergence of fitness for the training data of Colon using NSC$_P$-GA 195

Figure 7-11 A typical convergence of fitness plot for the training data of Leukemia data using NSC$_P$-GA 196

Figure 7-12 A typical plot for convergence of fitness for the training data of Lymphoma using NSC$_P$-GA 197

Figure 7-13 A typical plot for convergence of fitness for the training data of Lung using NSC$_P$-GA 199

Figure 7-14 A typical plot for convergence of fitness for the training data of Ovarian using NSC$_P$-GA 200

Figure 7-15 A typical plot for convergence of fitness for the training data of Prostate using NSC$_P$-GA 201

Figure 7-16 A typical plot for convergence of fitness for the training data of AD using NSC$_{MD}$-GA 203

Figure 7-17 A typical plot for convergence of fitness for the training data of Colon using $\mathrm{NSC}_{\mathrm{MD}}$-GA 204
Figure 7-18 A typical plot for convergence of fitness for the training data of Leukemia using $\mathrm{NSC}_{\mathrm{MD}}$-GA 205
Figure 7-19 A typical plot for convergence of fitness for the training data of Lymphoma using $\mathrm{NSC}_{\mathrm{MD}}$-GA 206
Figure 7-20 A typical plot for convergence of fitness for the training data of Lung using $\mathrm{NSC}_{\mathrm{MD}}$-GA 207
Figure 7-21 A typical plot for convergence of fitness for the training data of Ovarian using $\mathrm{NSC}_{\mathrm{MD}}$-GA 208
Figure 7-22 A typical plot for convergence of fitness for the training data of Prostate using $\mathrm{NSC}_{\mathrm{MD}}$-GA 209
Figure 8-1 Framework of the proposed approach, NSC-NSGA2 223
Figure 8-2 Pareto front solutions 226
Figure 8-3 Major steps of NSGA2 adapted from Deb *et al*. (2002) 227
Figure 8-4 Initial population algorithm using RNG 227
Figure 8-5 An example of an initial population with 10 chromosomes with shrinkage threshold values and the number of features in their corresponding sets ... 228
Figure 8-6 *Crowded tournament* selection algorithm used in Deb *et al*. (2002) 230
Figure 8-7 *Crowding distance* algorithm used in Deb *et al*. (2002) 230
Figure 8-8 Non-dominated sorting procedure used in Deb *et al*. (2002) 232
Figure 8-9 Steps for generating the new population from the combined population ... 233
Figure 8-10 An example of Pareto front of 9 shrinkage threshold solutions 234
Figure 8-11 A typical Pareto front plot of objective function *f1* against f_2 for AD dataset 236
Figure 8-12 A typical Pareto front plot of objective function f_1 against f_2 for the Colon cancer dataset 239
Figure 8-13 A typical plot of a Pareto front of objective function f_1 against f_2 for the Leukemia cancer dataset 241
Figure 8-14 A typical Pareto front plot of f_1 against f_2 for Ovarian cancer dataset 243
Figure 8-15 A typical Pareto front plot of f_1 against f_2 for Lymphoma cancer dataset . 245
Figure 8-16 A typical Pareto front plot of f_1 against f_2 for Lung cancer dataset 247
Figure 8-17 A typical Pareto front plot of f_1 against f_2 for Prostate cancer dataset 249

Figure 8-18 A typical Pareto front plot of f_1 against f_2 for Leukemia cancer dataset using the NSC_M-NSGA2 approach..256

Table of Tables

Table 3-1 Summary of seven public datasets used in the study....................................55

Table 3-2 Description of subsets associated with the Ray *et al.*'s dataset (2007) 57

Table 3-3 Description of subsets of Colon data 58

Table 3-4 Description of subsets of Leukemia data..59

Table 3-5 Description of subsets of Lung data 60

Table 3-6 Description of subsets of Lymphoma data 60

Table 3-7 Description of subsets of Prostate data...61

Table 3-8 Description of subsets of Ovarian data...62

Table 4-1 A set of parameters used to run GA..87

Table 4-2 Results associated with RST-GA (proposed approach) and from Banerjee, *et al.* (2007) using k-NN classifiers, with k=1, 3, 5, and 7 on the Colon unseen test set...91

Table 4-3 List of genes associated with the Colon Cancer dataset for each of the 15 optimal sets of features obtained from 15 independent runs of RST-GA.....93

Table 4-4 Results of classification for the 6 selected genes (highlighted in blue in Table 4-2) with 22 WEKA classifiers on the Colon unseen test set 97

Table 4-5 Results for RST-GA (proposed approach) and Banerjee, *et al.*(2007) using for k-NN classifier with k=1, 3, 5, and 7 on the Leukemia unseen test set. The column heading "Net" stands for the overall classification (%) for all classes on the Leukemia unseen test dataset and "Net" is calculated using Equation (4.3). 100

Table 4-6 List of 15 sets of genes selected for the Leukemia cancer dataset using RST-GA. 101

Table 4-7 Results of classification for the 5 and 14 selected genes (highlighted in blue and green, respectively, in...103

Table 5-1 Computational time spent for one trial involving the 4 parameter settings..124

Table 5-2 Four sets of parameter settings used in the tuning process..........................124

Table 5-3 Summary of results of running the algorithm using each of the 4 sets of parameter settings for 3 independent runs 134

Table 5-4 Parameter settings used in Chapter 5, 6 and 7 136

Table 5-5 Classification results for the AD data using NSC-GA approach and from Ray *et al*. (2007) 137

Table 5-6 List of 11 proteins selected using NSC-GA 139

Table 5-7 Classification results, for the Colon data using NSC-GA approach, and from Klassen and Kim (2009) 140

Table 5-8 Twenty eight genes selected for Colon cancer data using NSC-GA 141

Table 5-9 Classification results for the Leukemia data using NSC-GA approach and from Tibshirani *et al*. (2002), Klassen and Kim (2009), S. Wang and Zhu (2007) and J. Fan and Fan (2008) 141

Table 5-10 Nine genes selected by the proposed NSC-GA for Leukemia cancer data 143

Table 5-11 Classification results for the Ovarian data using NSC-GA approach and from Foss (2010) 144

Table 5-12 Classification results for the Lymphoma data using NSC-GA approach and from Klassen and Kim (2009) 145

Table 5-13 The sets of 7 and 12 genes selected by NSC-GA for Lymphoma cancer data 147

Table 5-14 Classification results for the Lung data from NSC-GA approach and from Klassen and Kim (2009), Tai and Pan (2007) and J. Fan and Fan (2008) .. 148

Table 5-15 The sets of 8, 9, 10 and 11 genes selected by NSC-GA, for Lung cancer data 149

Table 5-16 Classification results for the Prostate data from NSC-GA approach and from Klassen and Kim (2009), Tai and Pan (2007) and J. Fan and Fan (2008) .. 151

Table 5-17 Classification results for AD, Colon, Leukemia, Ovarian, Lymphoma, Lung and Prostate cancer data using NSC-GA 152

Table 5-18 Results of classification for the 6 selected genes with 22 WEKA classifiers on the Prostate unseen test set 155

Table 6-1 Parameter settings used in the proposed approach, NSC-MA 164

Table 6-2 Classification results and time to converge for NSC-GA and NSC-MA using AD data 166

Table 6-3 Classification results and time to converge for NSC-GA and NSC-MA using the Colon cancer data 168

Table 6-4 Classification results and time to converge for NSC-GA and NSC-MA using the Leukemia cancer data 170

Table 6-5 Classification results and time to converge for NSC-GA and NSC-MA using the Ovarian cancer data ... 171

Table 6-6 Classification results and time to converge for NSC-GA and NSC-MA using the Lymphoma cancer data ... 173

Table 6-7 Classification results and time to converge for NSC-GA and NSC-MA using the Lung cancer data ... 174

Table 6-8 Classification results and time to converge for NSC-GA and NSC-MA using the Prostate cancer data ... 176

Table 6-9 Summary of results obtained from the NSC-GA and NSC-MA approach ... 177

Table 7-1 Comparison of classification results between NSC-GA and NSC_M-GA for AD data ... 186

Table 7-2 Comparison of classification results between NSC-GA and NSC_M-GA for Colon cancer data ... 187

Table 7-3 Comparison of classification results between NSC-GA and NSC_M-GA for Leukemia data ... 188

Table 7-4 Comparison of classification results between NSC-GA and NSC_M-GA for Lymphoma data ... 189

Table 7-5 Comparison of classification results between NSC-GA and NSC_M-GA for Lung data ... 190

Table 7-6 Comparison of classification results between NSC-GA and NSC_M-GA for Ovarian data ... 192

Table 7-7 Comparison of classification results between NSC-GA and NSC_M-GA for Prostate data ... 193

Table 7-8 Comparison of classification results between NSC-GA and NSC_P-GA for AD data ... 195

Table 7-9 Comparison of classification results between NSC-GA and NSC_P-GA for Colon data ... 196

Table 7-10 Comparison of classification results between NSC-GA and NSC_P-GA for Leukemia data ... 197

Table 7-11 Comparison of classification results between NSC-GA and NSC_P-GA for Lymphoma data ... 198

Table 7-12 Comparison of classification results between NSC-GA and NSC_P-GA for Lung data ... 199

Table 7-13 Comparison of classification results between NSC-GA and NSC$_P$-GA for Ovarian data 201

Table 7-14 Comparison of classification results between NSC-GA and NSC$_P$-GA for Prostate data 202

Table 7-15 Comparison of classification results between NSC-GA and NSC$_{MD}$-GA for AD data 203

Table 7-16 Comparison of classification results between NSC-GA and NSC$_{MD}$-GA for Colon data 205

Table 7-17 Comparison of classification results between NSC-GA and NSC$_{MD}$-GA for Leukemia data 206

Table 7-18 Comparison of classification results between NSC-GA and NSC$_{MD}$-GA for Lymphoma data..207

Table 7-19 Comparison of classification results between NSC-GA and NSC$_{MD}$-GA for Lung data..208

Table 7-20 Comparison of classification results between NSC-GA and NSC$_{MD}$-GA for Ovarian data 209

Table 7-21 Comparison of classification results between NSC-GA and NSC$_{MD}$-GA for Prostate data 210

Table 7-22 Summary of the sets of selected features obtained from NSC$_M$-GA, NSC$_P$-GA, NSC$_{MD}$-GA and NSC-GA for AD data 211

Table 7-23 Summary of the sets of selected features obtained from NSC$_M$-GA, NSC$_P$-GA, NSC$_{MD}$-GA and NSC-GA for Colon data 212

Table 7-24 Summary of the sets of selected features obtained from NSC$_M$-GA, NSC$_P$-GA, NSC$_{MD}$-GA and NSC-GA for Leukemia data....................................213

Table 7-25 Summary of the sets of selected features obtained from NSC$_M$-GA, NSC$_P$-GA, NSC$_{MD}$-GA and NSC-GA for Lymphoma data 214

Table 7-26 List of 80 genes in the selected set obtained from NSC$_P$-GA for Lymphoma data 215

Table 7-27 Summary of the sets of selected features obtained from NSC$_M$-GA, NSC$_P$-GA, NSC$_{MD}$-GA and NSC-GA for Lung data 215

Table 7-28 Summary of the sets of selected features obtained from NSC$_M$-GA, NSC$_P$-GA, NSC$_{MD}$-GA and NSC-GA for Ovarian data.......................................216

Table 7-29 Summary of the sets of selected features obtained from NSC$_M$-GA, NSC$_P$-GA, NSC$_{MD}$-GA and NSC-GA for Prostate data.......................................217

Table 7-30 Summary of classification results for the respective unseen test datasets using the corresponding feature sets obtained using NSC-GA, NSC_{M}-GA, NSC_{P}-GA and NSC_{MD}-GA for each of the seven datasets 218

Table 8-1 Parameter set used for NSC-NSGA2..235

Table 8-2 Sets of selected proteins using the proposed approach, NSC-NSGA2, for AD data 237

Table 8-3 Classification results for the AD data using the sets of selected features from NSC-NSGA2 approach 238

Table 8-4 Sets of genes selected using the proposed approach, NSC-NSGA2 for Colon cancer data...240

Table 8-5 Classification results for the Colon cancer data using the sets of selected features from NSC-NSGA2 approach...241

Table 8-6 Sets of genes selected using the proposed approach, NSC-NSGA2 for Leukemia cancer data...242

Table 8-7 Classification results for the Leukemia cancer data using the sets of selected features from NSC-NSGA2 approach...243

Table 8-8 Subsets of genes selected using the proposed approach, NSC-NSGA2 for Ovarian cancer data..244

Table 8-9 Classification results for the Ovarian cancer data using the sets of selected features from NSC-NSGA2 approach...244

Table 8-10 Subsets of genes selected using NSC-NSGA2 for Lymphoma cancer data 246

Table 8-11 Classification results for the Lymphoma cancer data using the sets of selected features from NSC-NSGA2 approach...247

Table 8-12 Subsets of genes selected using the proposed approach, NSC-NSGA2 for Lung cancer data 248

Table 8-13 Classification results for the Lung cancer data using the sets of selected features from NSC-NSGA2 approach...249

Table 8-14 Subsets of features selected using the proposed approach, NSC-NSGA2 for Prostate cancer data...250

Table 8-15 Classification results for the Prostate cancer data using the sets of selected features from NSC-NSGA2 approach...251

Table 8-16 Summary of results for NSC-GA and NSC-NSGA2.................................252

Table 8-17 Result from applying 2 and 3 objective functions for the proposed approach, NSC-NSGA2, on AD and Leukemia dataset 255

Table 8-18 Sets of genes obtained by NSC_M-NSGA2 for the Leukemia cancer data .. 257

Table 8-19 Results from NSC-NSGA2 and NSC_M-NSGA2 for Leukemia cancer dataset 257

Table 9-1 Common features selected across difference approaches............................ 263

Table 9-2 Advantages and limitation of the developed approaches 264

Table of Abbreviations

Abbreviation	**Meaning**
AD	Alzheimer’s Disease
CV	Cross validation
DM	Data mining
EA	Evolutionary algorithm
FS	Feature selection
GA	Genetic algorithm
k-NN	k - nearest neighbour
LS	Local search
MA	Memetic Algorithm
MCI	Mild cognitive impairment
MOEA	Multi-objective evolutionary algorithm
MS	Mass spectrometry
NAD	Non-Alzheimer’s Disease
NDC	Non-demented control
NSC	Nearest shrunken centroid
NSGA2	Non-dominated sorting genetic algorithm
OD	Other dementia
OND	Other neurological disease
PAM	Prediction Analysis for Microarrays
RNG	Random number generator
RST	Rough set theory
WEKA	Waikato Environment for Knowledge Analysis

1. Introduction

Recently, mass throughput technologies such as microarrays and mass spectrometry (MS) have been developed and widely used in the biomedical domain. A large number of biological datasets involving different types of diseases such as cancer and Alzheimer's disease (AD) have been generated using these technologies (Ma & Huang, 2008; Stoeckel & Fung, 2007). Microarrays allow thousands of genes to be measured simultaneously in a single experiment. MS technology produces enormous amounts of high-dimensional datasets about cellular functions. Examples of DNA microarray datasets include gene arrays with up to 500,000 genes and MS datasets with 300,000 m/z (a unit of measure) values (Aliferis, Statnikov, & Samrdinos, 2006).

Typically, biomedical research involving the above mentioned techniques is linked to prevention, diagnosis and drug development for treatment of diseases; with a focus in diseases such as cancer and Alzheimer's disease. According to Cancer Research UK and American Cancer Society, globally cancer is a leading cause of disease and cause of death. In 2008, 12.7 million new cancer cases and 7.6 million people died of cancer. The worldwide trend is predicted to be a significant increase of 22 million new cases each year by 2030 (American Cancer Society, 2011; Cancer Research UK, 2012), that is about 286 million people will be diagnosed to have cancer by 2030.

According to the World Alzheimer 2012 Report, globally, about 36 million people have Alzheimer's Disease or dementia and this number will increase to 66 million and 115 million by 2030 and 2050 respectively, that is about one new case every four seconds (Alzheimer's Association, 2012; Alzheimer's Australia, 2012). There are no early diagnostic tests that are definitive for this disease, with a definitive diagnosis only possible following a post-mortem examination of the brain for evidence of the disease's characteristic neuropathology (MayoClinic, 2013). However, the pathogenic processes of Alzheimer's disease are likely to begin years before clinical symptoms are observed. Therefore, the need for biological markers (biomarkers) defined as "*a substance, physiological characteristic or gene, that indicates, or may indicate, the presence of disease, a physiological abnormality or a psychological condition*" (Biological marker, n.d.), whose measurable levels are altered prior to clinical symptoms is of paramount importance. The need to detect Alzheimer's disease via an "*equivalent pregnancy test* " has been repeatedly stated in the literature (Trojanowskl, 2004). The ideal diagnostic test is one that is inexpensive, has a high specificity and can be carried out as easily and accurately as a "*pregnancy test*"; enabling diagnosis as early as possible (Hooper, Lovestone, & Sainz-Fuertes, 2008)

While the availability of such datasets can aid in the development of techniques and drugs to improve diagnosis and treatment of diseases, the nature and the enormous volume of such mass throughput data challenge the power of data mining (DM) in terms of their analysis to extract useful and meaningful information. A fundamental problem in identifying biomarkers from high dimensional data involves a systematic search for relevant features; to reduce the dimensionality of the dataset to a small, yet highly reliable and discriminative subset that is representative, improving the classification accuracy and reducing the computational cost (Hanczar *et al.*, 2003; Somorjai, Dolenko, & Baumgartner, 2003).

1.1. Statement of the Problem

Analysis of high dimensional data, in general, have problems that arise from '*the curse of dimensionality*' (R. Clarke *et al.*, 2008), which relates to a very large number of attributes (features) and '*the curse of dataset sparsity*' (Somorjai *et al.*, 2003), which relates to the small number of samples (e.g. in the case of a prostate cancer dataset with 12600 features and only 102 samples) in the dataset. These problems result in overfitting, inaccurate classification and high computational cost in searching through the feature subspace (Kim, Kim, Lim, & Kim, 2010). Owing to the complexity of the data it is very important that the number of features be reduced in order to improve classification accuracy and to perform the analysis with less computational cost (Liu, Li & Wong, 2002). Additionally, owing to the *curse of dimensionality*, traditional statistical approaches and machine learning techniques are not effective in analysing these types of datasets (Yu & Liu, 2004).

One approach to find biomarkers is to use feature selection (FS) techniques to select the most relevant feature subsets. "*Feature selection can be defined as a problem of finding a set of minimum number of relevant features that describe the dataset.*" (Kim *et al.*, 2009, p.2). It is the process of going through the vast amount of data, including a large number of features in the dataset, to select relevant feature subsets (possibly an optimal subset), which improve the classification accuracy in terms of sensitivity (a probability that the prediction is positive when the disease is present, i.e., true positive prediction) and specificity (a probability that the prediction is negative when the disease is not present, i.e., true negative prediction) (Dash & Liu, 1997, 2003). In addition the identification of an optimal subset of features, capable of providing absolute discriminatory information, can lead to the development of inexpensive diagnostic assays with a few features (genes) and which subsequently can be widely deployed in clinical settings.

Another consideration in addressing the problem of finding relevant biomarkers is related to characteristics typically associated with biological datasets that make the DM task especially challenging. These include the following:

- **Noisy data**: This can be attributed to differences in experimental setups; technologies and impression with their associated devices and software; and variances in biological observations.
- **Datasets typically are of small sample size but high dimensional**: Unlike traditional domains associated with DM applications, biological datasets typically have only a small number of samples (at best in the hundreds), while the number of features, is typically in tens of thousands. This characteristic leads to the phenomena, curse of dimensionality and over-fitting in classification tasks. Algorithms developed to carry out DM in traditional domains are not suitable to be used to analyse these datasets. In addition, this characteristic will also create a scenario where there is a high likelihood of finding *false positives* in classification tasks owing to chance, and robust methods to validate the classification models are vital.
- **Complexities of interactions amongst features in a biological dataset.** Features in the biological datasets are not independent; their correlation structure is not fully understood in many cases. Many data analysis approaches only involved evaluating each feature separately and do not consider possible correlations amongst features. However, from a biomedical perspective, groups

of features are known to work together as pathway components in a biological process.

- **Biological and diagnostic relevance**

  Another point to note is that data obtained via mass throughput technologies such as microarray serves 2 functions:

  - biological relevance - by providing measurements related to mechanisms underlying the disease and
  - diagnostic relevance - as relevant features in the construction of accurate diagnostic classifiers for prediction.

  It is vital to understand the interplay between diagnostic and biological relevance -- that the former is neither a necessary nor a sufficient condition for the latter. First, high correlation between disease status and specific features do not necessarily imply that they have a causal relationship with the specific disease. Likewise, in constructing accurate diagnostic classifiers it is highly unlikely that all biologically relevant features may be utilized. The outcome is that selected features on the basis of their diagnostic relevance need be validated for biological relevance by examination of the literature for relevance to the specific disease and by subsequent experimental analysis. Second, biological evidence suggest that typically multiple sets, each with a finite number of features, are responsible for a specific disease (i.e. multiple causes -- features can be combined in many different ways, all leading to a specific disease). The outcome here is that identifying multiple sets of biomarkers is important for discovering correlations among features and to support evaluation of different combinations in the diagnostic phase.

- **Validation of results from data analysis and absolute ground truth:** Absolute ground truths are not available in this field for validation of results associated with the data analysis. In other disciplines, experts can be readily available to provide ground truths but in the areas of proteomics and genomics, biomedical knowledge in terms of differential physiological behaviour pertinent to specific biological states is currently inadequate. The ultimate judge for validation would involve biological validation (e.g. clinical trials), for which one will have to focus on some specific subsets of minimal size. Results from data analysis are also likely to be useful if they are position in context and can be subsequently followed up with more focused studies by biomedical researchers.

Many FS algorithms have already been developed and used in various areas such as business, science, engineering, and in recent times, increasingly applied in the area of bioinformatics. Cho *et al*. (2003) used a genetic algorithm (GA) together with a neural network classifier to select relevant features from Alzheimer's disease datasets. A Support Vector Machine (SVM) classifier was used in Mukerjee *et al*.'s study (1998) to select features from a Leukemia microarray cancer dataset.

Existing work involving evolutionary approaches include Li, Liu and Bai (2008) incorporated a GA into filter and wrapper methods to search for feature subsets from a Prostate MS dataset; Deb and Reddy(2003) incorporated the method of weighted voting into a multi-objective evolutionary algorithm (MOEA) called non-dominated sorting algorithm (NSGA2) to search for multiple sets of optimal features for Leukemia, Colon, Lymphoma, GCM and NCI60 cancer data.

Rough set theory (RST) (Pawlak, 1982) was developed on a mathematical basis and has been used in DM to analyse vague, uncertain or incomplete data in datasets and to remove redundant features effectively (Pawlak, 1997). RST has also already been used to select features for biomedical data in numerous studies (Punitha & Santhanam, 2008). However approaches incorporate RST with an evolutionary algorithm (EA) such as MOEA (NSGA2) or GA to search for optimal set of features for high dimensional biological data are limited. For example, Banerjee *et al*. (2007) proposed an evolutionary Rough Set based FS technique for analysing gene expression data.

The Nearest Shrunken Centroid (NSC) algorithm (Tibshirani, Hastie, Narasimhan, & Chu, 2002) with its most well-known software implementation being known as Prediction Analysis for Microarrays (PAM), has been widely used as a FS and classification method for high dimensional biomedical data in numerous studies (Ray et al., 2007). NSC selects features by shrinking a class centroid for each feature toward its overall centroid for all classes using a shrinkage threshold value.

Shrinkage threshold values associated with the application of NSC have usually been selected via 2 approaches, Cross Validation (CV) (Tibshirani et al., 2002; S. Wang & Zhu, 2007; K. Yeung & R. Bumgarner, 2003) and empirical approach (Klassen & Kim, 2009; Levner, 2005; Ray *et al.*, 2007). With the CV approach, the dataset is divided randomly equal into *k* parts, each part consists of approximate proportion of a number of samples and classes. One part takes turn to be the test set while the other *k-1* parts are used as the training set. The procedure is repeated *n* times to obtain the prediction error rate for each time. The overall prediction error rate is then calculated by averaging the errors from all iterations. The selected optimal threshold value is based on the CV prediction errors associated with the different threshold values. With the empirical approaches, the optimal shrinkage threshold was selected based on the lowest classification error over a range of shrinkage thresholds. However, threshold values selected using CV and empirical approaches are not precisely tuned for the specific dataset to obtain optimal classification results. This is due to the fact that these approaches are limited in terms of exploring the entire search space of threshold values in relation to the dataset, resulting in threshold values that may not be the optimal. Optimal shrinkage threshold values used in the NSC algorithm would make a vital difference in selecting optimal feature sets and subsequently, improving the classification accuracy.

To address the challenges in the analysis of mass throughput data such as microarray data, FS is seen as a vital first step to identify relevant features for classification. Although many FS techniques have been developed and used for analysis of such high dimensional biological data, most of the existing work typically involved deterministic approaches, attempting to find a unique set of biomarkers. The development of techniques capable of extracting multiple potential sets of biomarkers for subsequent analysis and the incorporation of evolutionary algorithms, especially MOEA in these FS techniques is limited. Following this direction, this research study aimed to develop evolutionary based FS techniques for analysis of high dimensional biological data generated from molecular biology techniques such as microarrays, metabolite profiling and mass spectrometry and to evaluate these techniques, both in terms of their performances and the validity of the extracted information.

1.2. The purpose of the study

The aims of this project are:

- to investigate and develop FS algorithms that incorporate various evolutionary strategies, specifically investigating the use of evolutionary strategies in conjunction with RST and NSC;
- to evaluate the developed algorithms in terms of finding the "most relevant" biomarkers contained in biological datasets and
- to evaluate the *goodness* of extracted feature subsets for relevance (examined in terms of existing biomedical domain knowledge and classification accuracy form the perspectives of sensitivity and specificity associated with different classifiers). The project aims to generate sets of features for construction of good predictive models for classifying diseased samples from control.

1.3. The contributions of this study

The area of bioinformatics is "data rich", as the breakthroughs in the development of mass throughput technologies resulted in huge volumes of data being produced. However, this area increasingly suffers from a situation where biomedical researchers lack the time and the appropriate tools to complete a sound and comprehensive analysis of these huge volumes of data in order to make biological sense and to use the data optimally. The study contributes in the area of bioinformatics, in the development of FS techniques that aid in the analysis of datasets acquired using mass throughput technologies. Specifically, the study examines the development of FS techniques that incorporates evolutionary algorithms, especially MOEA. Unlike existing techniques, the developed approaches support FS by simultaneously considering tradeoffs between a number of criteria (e.g. high classification accuracy and a small number of features). Additionally, the developed techniques is cost and time effective, allowing researchers to use computer time to analyse realms of data as the output from the techniques are multiple sets of potential features (biomarkers) that can be further investigated to explore both diagnostic and biological relevance. The following section details the contributions of the study.

- **A FS technique incorporating the use of GA, K-means and RST for analysis of high-dimensional biomedical data**

  Use of RST as a FS technique in bioinformatics has been limited and this study investigated an approach of combining RST with a GA. The first step in this approach, RST-GA, employed the k-means algorithm to generate class centroids from the training data. The class centroids are used as initial seed values for RST to partition the data that subsequently led to the reduction of a large number of features. GA was then utilised as a search method to find sets of optimal features. Unlike deterministic approaches that produce the same set of optimal features, this approach produced a different set of features from each run of RST-GA. Identification of multiple sets of biomarkers with high diagnostic relevance is important as it allows biomedical researchers to examine these sets using existing biological knowledge to determine sets to validate for biological relevance in subsequent clinical studies.

- **Use of evolutionary algorithms for enhancement of the NSC algorithm**

  The NSC algorithm has been widely used as a FS and classification method for high dimensional biomedical data in many studies (Bair & Tibshirani, 2004; Klassen & Kim, 2009; Lee, Lee, Park, & Song, 2005; Ravetti & Moscato, 2008; Ray et al., 2007; K. Y. Yeung & R. E. Bumgarner, 2003). A shrinkage threshold value must also be provided to the NSC and this is normally selected via a manual "*trial and error*" process which can be very time consuming. The resulting shrinkage threshold value from this manual process may be limited by the granularity of the initial pre-determined values. In this study, evolutionary based approaches, NSC-GA, NSC-MA and NSC-NSGA2 involving GA, memetic algorithm (MA) and MOEA (NSGA2) respectively, were developed to find shrinkage threshold values automatically. These approaches eliminate the need to find the shrinkage threshold value manually and produced more precise shrinkage threshold values. For NSC-GA described in Chapter 5 and NSC-MA in Chapter 6, the shrinkage threshold value is determined on the basis of a single objective function which is an aggregation of 2 separate objective functions (i.e. evaluation criteria). For NSC-NSGA2 described in Chapter 8, multiple optimal shrinkage threshold values are obtained while simultaneously considering different tradeoffs amongst multiple objective functions.

Unlike approaches (e.g. S. Wang and Zhu (2007)) that attempted to improve the performance of NSC by modifying it, the original NSC algorithm (Tibshirani *et al.*, 2002) is used here, thus potentially the proposed techniques can also be incorporated into any modified NSC.

A point to note with regards to the nature of shrinkage thresholds is that rather than being an exact value, a narrow range of values maps to the same set of features. This implies that owing to the stochastic nature of evolutionary approaches, these approaches produced different results (i.e. different shrinkage threshold values resulting in different sets of features) from each run of the technique. However if these shrinkage threshold values are only slightly different, they mapped to the same set of features, thus in some cases, producing identical sets of features from a number of different runs. While having less variability, these approaches still produce multiple sets of biomarkers from the analysis of a dataset.

Lastly, another advantage associated with the proposed approaches being able to produce more precise shrinkage threshold values is obtaining better classification accuracy when the corresponding optimal set of features is employed in NSC to classify unseen test datasets.

- **Investigate the impact of using different distance measures in the NSC algorithm**

  The study investigated the impact of using different distance measures: Mahalanobis, Pearson and Mass Distance (MD), in the NSC algorithm employed in NSC-GA for analysis of high dimensional biological data. In the NSC (Tibshirani *et al.*, 2002), Euclidean distance is employed as an evaluation measure in determining the score used to classify sample points. The Euclidean distance is not an effective nor a robust measure for classification when compared to other similarity measures such as Mahalanobis, Pearson and MD (Datta & Datta, 2003; Ding & Peng, 2005; Yona, Dirks, Rahman, & Lin, 2006).

This investigation contributes to a better understanding of using the different distance measures in NSC and the impact these have on the task of finding the most relevant features that can lead to higher classification accuracy for biological data.

- **Identification of a number of subsets of relevant features for Alzheimer's disease, Colon, Leukemia, Lung, Lymphoma, Ovarian and Prostate cancer data.**

  Approaches developed in this study were evaluated using seven datasets. Indirectly, each of these datasets was analysed for finding optimal sets of biomarkers, by considering tradeoffs between high classification accuracy and minimum number of features. Most existing studies (Banerjee et al., 2007; Fan & Fan, 2008; Foss, 2010; Tai & Pan, 2007; S. Wang & Zhu, 2007) evaluate their developed techniques using various datasets and only reported their findings in terms of the size of the optimal feature sets and associated classification accuracy. However, besides examining the classification performance of a set of features, its relevance to its corresponding domain is crucial. Unlike previous work, this study lists the extracted features from the analysis of each datasets and where possible, examines the relevance of these features by searching the literature. These subsets of relevant features can be used by biomedical researchers for further clinical investigation to validate their biological relevance to the specific disease.

- **Impact of using specific classifiers on sensitivity and specificity**

  Sensitivity and specificity associated with classification are two measures that are of great interest to the biomedical community in their efforts to find biomarkers and to assess them as to how well they can predict relevant outcomes. Sensitivity represents the probability of correctly diagnosing a condition (i.e. the proportion of truly affected (i.e. diseased)) in a sample population that is identified by the test as being diseased). On the other hand, specificity represents the proportion of truly non-diseased that the test identified as such. This study demonstrated that different classifiers constructed using the same set of features produced different sensitivity and specificity in the classification of test data. In other words, there can be classifier-bias – for example, classifier C, constructed using a set of features demonstrates high sensitivity and low specificity; when in actual fact, there may exists a number of other classifiers constructed using the same set of features and demonstrating both high sensitivity and high specificity.

Thus in a DM analysis for finding suitable sets of biological markers, a number of classifiers should be used instead of just using one. This will avoid cases of missing out on sets of features with high discriminatory capabilities that should be further investigated in early diagnostic test developments but were rejected on the basis of their sensitivity/specificity obtained via a specific classifier.

- **A set of techniques for comprehensive analysis of biological datasets**
  As mentioned previously, data from mass throughput technologies typically consists of a small number of samples where each is composed of thousands of features. Additionally these features have correlation relationships that are still not fully understood. From a biomedical perspective, groups of features are also known to work together as components in a biological pathway. Given these complexities in the data, manual evaluation to find sets of features would be intractable. Many existing data analysis approaches in bioinformatics only involved evaluating each feature separately (univariate analysis) and do not consider possible correlations amongst features nor the joint behavior of a combination of features. The set of techniques developed in this study attempts to address this limitation where the basis of the selection involved the evaluation of different combinations of features by simultaneously considering two or more selection criteria.

Owing to the stochastic nature of the evolutionary based approaches developed in this study, multiple optimal sets, consisting of a varying number of features, and of high diagnostic relevance are obtained. While the multiple sets of features obtained via RST-GA (described in Chapter 4) showed a varying degree of overlap (in other words, different numbers of common features), NSC-based approaches, namely NSC-GA, NSC-MA, NSC-NSGA2 produced feature sets where a smaller set is always a subset of a larger set from analysis of a specific dataset.

An interesting consequence of this characteristic is seen when all these different sets of features obtained via a NSC-based approach (e.g. NSC-GA) is used to construct classifiers for classifying unseen test data. The domain expert can make informed decision based on the tradeoffs between classification accuracy and size of a feature set. For example, in the event where a set with 6 features produced the same classification accuracy as a set with 7 features, the domain expert can examine the 7th feature and use domain knowledge to decide on it potential relevance and make a decision about its inclusion in subsequent analysis. Equally in another scenario, if sets with 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 21 and 23 features respectively resulted in classifiers producing the same classification accuracy on the unseen test dataset, it would then appear that a major contributing factor relates to 1 feature and the domain expert may validate this in a subsequent clinical trial. This sort of information from the analysis is important as reducing the number of features to a smaller promising set, for further investigations, would reduce costs associated with future experiments and development of diagnostic toolkits.

In summary, this thesis contributes towards a better understanding of incorporating evolutionary approaches in the development of techniques for analysing biological data from mass throughput technologies. The techniques developed here can be used for a comprehensive analysis of a dataset, extracting information that biomedical researchers can use to make informed decisions with regards to evaluation of sets of biomarkers for biological relevance. The thesis also contributed to an increased understanding of the impact of employing different similarity measure in NSC and demonstrated the need to be aware of the possibility of classifier-biased when examining the sensitivity and specificity associated with a specific set of features.

1.4. Significance

- The developed techniques improves NSC and allows researchers using NSC to be able to obtain shrinkage thresholds automatically, thus reducing time and effort required.

- The developed techniques can be used for a comprehensive analysis of high dimensional biological dataset, extracting information that biomedical researchers can use to make informed decisions for subsequent investigations of sets of biomarkers. Instead of traditional univariate analysis, the developed techniques allowed biomedical researchers to examine the joint classification behaviour of different sets of features in the development of diagnostic toolkits.

1.5. Structure of the thesis

The thesis consists of nine chapters. The primary theme in this thesis is the investigation of evolutionary approaches for feature selection in biological data and this thread of investigation starts with the pilot study described in Chapter 4, progressing to investigation of GA for automatically obtaining the shrinkage thresholds for NSC in Chapter 5, followed by the investigation involving memetic algorithms in Chapter 6 and culminating in the multi-objective approaches described in Chapter 8. Chapter 7 described the investigations to examine the impact of different the secondary theme being the impact of different similarity measures in NSC. Some preliminary concepts and descriptions of the datasets for evaluation of the developed approaches are described in Chapter 3.

The following section gives an overview of each of the remaining chapters in the study.

Chapter 2 describes a literature review consisting of 2 major sections 1) a review of DM techniques and algorithms that have been previously developed by other researchers for FS and classification in the domain of bioinformatics; and 2) technical descriptions of algorithms that have been employed in the proposed approaches in this study. These include RST, NSC, GA, NSGA2, MA and different similarity distance measures (Euclidean, Mahalanobis, Pearson and MD) algorithms are discussed.

Chapter 3 describes 7 biological datasets, Alzheimer's disease, Colon, Leukemia, Lung, Lymphoma, Ovarian and Prostate cancer data, used in the study. Data configurations for training and test sets, 10 fold cross validation (CV) strategy, as well as general information of how the datasets were used to evaluate the developed algorithms are also detailed in this chapter.

Chapter 4 describes the proposed approach RST-GA, which incorporated RST and GA. K-means clustering method is used to find the centroid for partitioning data in the reduction of features using RST approach with a non-deterministic algorithm, GA. The results of evaluating the proposed are described using Colon and Leukemia cancer data.

Chapter 5 describes the proposed approach of incorporating NSC into GA (NSC-GA) to automatically search for optimal shrinkage threshold values for NSC. In this chapter, the details of the proposed approach, NSC-GA that utilised the training dataset for obtaining optimal shrinkage threshold values for NSC automatically are described. Also in this chapter, the details of the proposed approach that employed NSC as an evaluator to evaluate the goodness of the feature subsets and GA as a search strategy to find optimal shrinkage threshold values for NSC are described.

Chapter 6 describes the proposed approach of incorporating NSC into MA (NSC-MA) to improve the search for finding optimal shrinkage threshold values for NSC automatically. The details of a local search implemented in MA are also described in this chapter.

Chapter 7 describes the proposed approach of incorporating different similarity distance measures (Mahalanobis, Pearson and MD), into the NSC-GA framework to improve the search for finding smaller sets of relevant features that lead to higher classification accuracy.

Chapter 8 describes the proposed approach of incorporating the NSGA2 algorithm as a MOEA into NSC to find multiple solution sets of optimal shrinkage threshold values automatically for NSC. In this chapter, the details of the proposed approach, NSC-NSGA2 that employed the NSC algorithm as an evaluator are also described.

Chapter 9 summarises the main findings from the thesis and outlined the proposed approaches that have been developed in the study. Future work is also discussed at the end of this chapter.

1.6. Summary

This chapter has provided the background and highlights some important aspects that lead to the use of FS algorithms and DM as necessary tools to select relevant features and analyse biological data. The purpose of the study and contributions from the study were also discussed. In the next chapter, previous studies associated with FS in bioinformatics are described. Techniques applicable to this study are reviewed.

2. Literature review

This chapter consists of 3 sections: 1) Review of DM and FS techniques; 2) review of some existing work which applied FS techniques to select relevant feature subsets and classify data in the area of bioinformatics; 3) review of techniques which were incorporated in the implementation of proposed approaches in this study.

2.1. Data mining and feature selection techniques

As mentioned earlier in Chapter 1, microarrays and mass spectrometry techniques generate massive amounts of high dimensional data. It is good in terms of data enrichment and availability, but at the same time challenging in terms of selecting the most relevant features for classifying the data accurately. DM and FS are approaches that have been widely used for analysing high dimensional data. FS techniques are used to select optimal (of minimal size) sets of relevant features from a high dimensional dataset efficiently. The following sections describe the general concepts of DM and FS, and specially the three categories of FS methods: filter, wrapper and embedded.

2.1.1. Data mining

The task of automatically finding interesting patterns from large data repositories is known as DM, and it can be categorized into predictive and descriptive tasks (Tan, Steinbach, & Kumar, 2006). Classification techniques are associated with predictive tasks where the aim is to predict the target variable using values of other attributes of the dataset. This is known as a supervised learning classification technique because the classification algorithm has to be trained in the training phase to produce the predictive model which is then evaluated for its performance in the testing phase. For descriptive tasks, techniques like clustering, also known as an unsupervised learning classification technique, are used to classify data that do not have class labels. Thus, the classification model does not need to be trained prior to perform the predictive task. The unsupervised classification technique classifies data based on their similarity measures into a group (class), e.g., similar distance measures, similar gene profiles. General concepts of classification techniques are described in the following section.

2.1.1.1. Classification

As mentioned earlier, classification is a type of supervised learning. The classifier needs to be trained with a training data, and evaluated with test data before being used for the classification on an unknown data (Tan *et al.*, 2006). The process of supervised learning may be illustrated by the following figure:

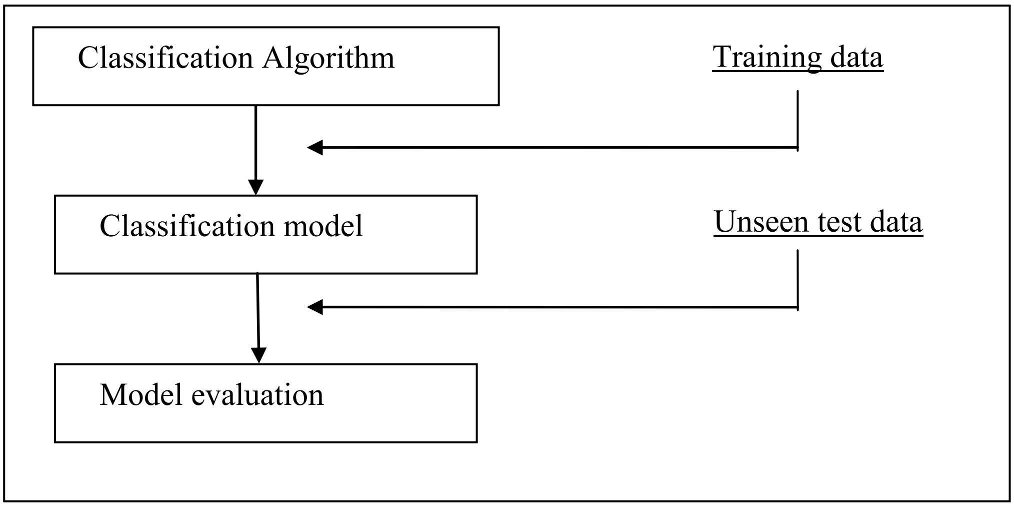


Figure 2-1 A general process of supervised learning for classification

As seen in Figure 2-1, in the process of supervised learning, a dataset is split into a training set and a test set. The training dataset is used to train the classifier, generating a classification model, whilst the unseen test dataset (not seen in the training phase) is used to evaluate the classification model for its accuracy on prediction.

One measure of the performance of the training model is the accuracy of its prediction. A smaller error rate in prediction indicates better and a more reliable a model. A confusion matrix table is used to show the number of correct and incorrect predictions for each class. Classification accuracy and error rates are calculated using Equation (2.1) and (2.2) (Tan *et al.*, 2006), as follows.

$$\text{Accuracy rate} = \frac{\text{number of correct predictions}}{\text{Total number of predictions}} \tag{2.1}$$

$$\text{Error rate} = \frac{\text{number of wrong predictions}}{\text{Total number of predictions}} \tag{2.2}$$

According to Han, Kamber and Pei (2006), Tan *et al*. (2006) and Witten and Frank (2005), the performance of a classifier is evaluated by using one of the following: holdout, random subsampling, cross-validation and bootstrap techniques.

- Holdout technique with stratification: the dataset is stratified into training data and test data (e.g. ½ for training and ½ for testing or 2/3 for training and 1/3 for testing). Stratification is the process where samples in the dataset are divided proportionally into the training and test datasets with balance in classes. The training data is used to build a classification model and the test data is used to evaluate the accuracy of the model.
- Random subsampling: the principle is similar to the holdout method but the training and test process are repeated a number of times to obtain classification accuracy accordingly. Each time the data are splitted randomly into the training set and the test set (Dieterle, 2003). The average classification accuracy over a number of iterations is the overall result for the classifier's accuracy.
- Cross-validation (CV): is the method to divide the dataset into a number (k) of subsets, e.g. 10 fold CV where k=10. Each subset consists of a proportional number of samples with balanced number in classes (stratification). K-1 subsets are used as training data to train the classification model and the remaining subset is used as test data to evaluate the model. This procedure is repeated k times, e.g., for 10 fold CV the procedure repeated 10 times, therefore every subset, in turn, is used as a test set. The average accuracy over k times is the overall classification accuracy of the model. K-fold CV methods are normally used in conjunction with the stratified holdout method as a standard method for evaluating classification results, e.g., *stratified 10 fold CV* (Witten & Frank, 2005).
- Bootstrap: similar to random subsampling, but samples that have been selected for the training data still remained in the original dataset, so that they have a chance to be chosen again. Bootstrap 0.632 is a popular approach used to evaluate the classifier. With the Bootstrap 0.632 approach, the training set consisting of 63.2% of the samples and the test set consisting of 36.8% of the samples (Dieterle, 2003). The calculation of bootstrap .632 is shown in Equation (2.3) (Tan *et al*. (2006), as follows.

$$Accuracy = \frac{1}{n}\sum_{i=1}^{n}(0.632\varepsilon_i + 0.368Acc_m) \quad (2.3)$$

2.1.1.2. Feature selection

Generic Steps in FS

According to Dash and Liu (1997) and Hall (1999) FS techniques consist of 3 major steps:

Step 1: Apply a search strategy to obtain subsets of features. The search strategies include: evolutionary algorithms, e.g., GAs or MOEAs, greedy, best first search with forward search selection (FSS) and backward search selection (BSS).

Step 2: Employ evaluation criteria such as distance measures, information measures, dependence measures or consistency measures. These measures are considered as a filtering mechanism, because they are independent processes to evaluate the candidate sunsets. Another measure is classifier error rate if a classifier is involved in the process of evaluating the candidate subsets.

Step 3: Determine a stopping criterion to stop the iteration process of selecting subsets. Stopping criteria might be based on a pre-defined maximum number of generations to run the algorithm or the convergence of the algorithm or a solution is found (Lancashire, Rees, & Ball, 2008). The following figure illustrates the above steps:

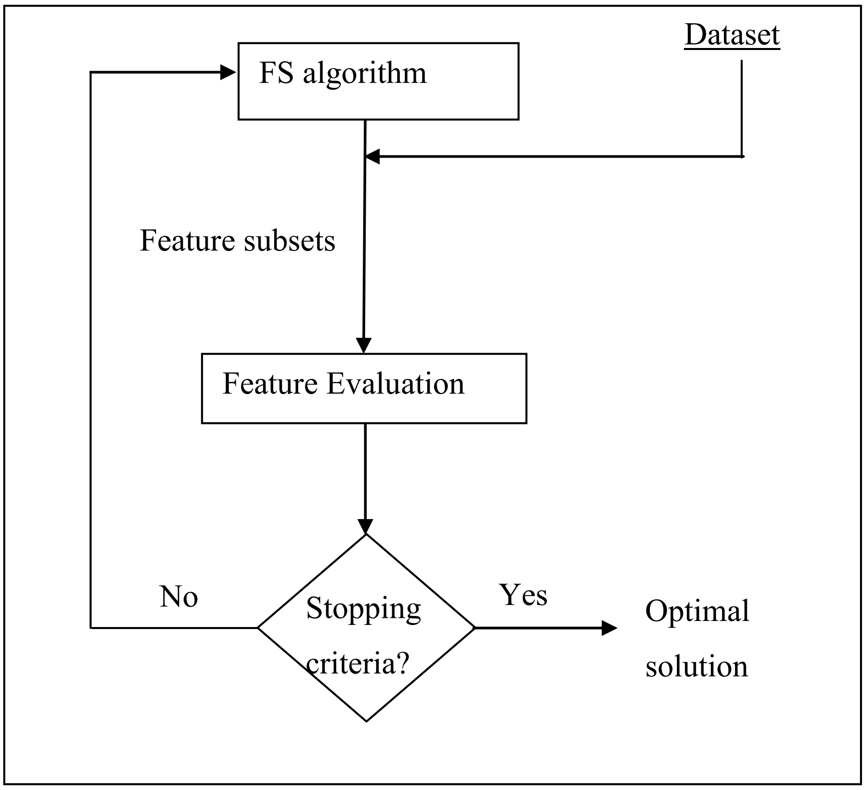


Figure 2-2 Steps in a FS process

Many FS techniques have been developed and applied to a variety of fields such as DM, bioinformatics and health related areas (Portinale & Saitta, 2002; Saeys, Inza, & Larranaga, 2007). In general, these techniques fall into three categories: filter methods, wrapper methods or embedded methods, which are described in the following sections.

Filter methods

Filter methods are performed prior to the use of a learning algorithm. Filter methods use separate independent techniques such as T-test, Kolmogorov Smirnov (KS) test and P-test to rank individual features (Levner, 2005). Filter methods select relevant features by calculating scores for each feature. Features with low scores are eliminated from the list. Thus only a number of high scoring features are retained and considered as relevant features. At the end of the filtering process, only one feature subset is generated and used to construct the classifier.

Wrapper methods

Wrapper methods are different from the filter methods described above. Instead of finding a relevant feature subset by a separate independent process, the wrapper method has its own machine learning algorithm (classifier) employed as part of the FS process. Unlike the filter method, numerous feature subsets are generated in the wrapper method and each of them evaluated using the machine learning algorithm. The process iterates a number of times until the best feature subset is found (Guyon, 2007; Guyon & Elisseeff, 2006; Inza, Larranaga, Blanco, & Cerrolaza, 2004; Saeys *et al.*, 2007). The number of iterations depends on the total number of features in the dataset (i.e., more features, more subsets generated and thus more iterations are needed to obtain an optimal feature subset).

According to Kohavi and John (1997), and Hall (1999), the learning algorithm (wrapper) is considered as a "*black box*" due to components of the black box, including FS, feature evaluation and the learning algorithm (classifier) itself, are not known from the outside. The way the method works is that the "*black box*" generates feature subsets using the training dataset and evaluates them using the classifier error or accuracy rate. The process stops when the termination condition(s) is met and the best feature subset is selected, and subsequently it is used for constructing the classifier.

Embedded methods

Embedded methods are methods that have a FS algorithm built into their classifiers, so that the search for relevant attributes can be done within the classifier itself using the dataset. As a result, a set of features is selected, and then a predictive model is generated and evaluated by the classifier.

Due to the importance of relevant (optimal) features in classification of high dimensional data, developing an advanced FS technique that can select the most relevant features from this type of data is one of the foci of DM community. A large number of search strategies have been developed by many DM researchers for finding optimal feature subsets that can be used in the investigations for biomarkers. The following sections describe some FS techniques that have been proposed in previous studies.

2.2. Feature selection approaches have been developed in the area of bioinformatics

The following sections describe some existing work related to FS approaches in the domain of bioinformatics. This section consists of 4 sub sections that described FS approaches involving evolutionary algorithms, rough set theory (RST), nearest shrunken centroid (NSC) and hybrid FS approaches, respectively.

2.2.1. Feature selection using EAs

2.2.1.1. Genetic algorithm

The GA has been employed for FS as a standalone approach or as a hybrid approach which incorporates other algorithms such as SVM, k-Nearest Neighbor (k-NN) for finding feature subsets for high dimensional biological data. The following section describes some of these approaches.

The GA was used in the study of Yang and Honavar (1998) to select relevant features for the Wisconsin diagnostic breast cancer data. The overall fitness of sets of features is evaluated based on the aggregation of 2 objective functions: classification accuracy obtained from neural networks and the cost of performing the classification for each candidate feature subset (solution). The study has demonstrated that GA selected a set of features which is half the size of the entire feature space and still retained the same accuracy of 92.1% as to the case of using all the features. In the study of Handels, Rob, Kreusch, Wolf, & Poppl (1998), GA was also employed to select features for a tumour skin cancer dataset. Similarly, the fitness of candidate solutions are also evaluated based the aggregation of 2 objective functions, one for a number of features selected in the set and another one for its associated classification accuracy. That is, one objective function is employed for maximizing the minimal set of set features and another objective function is employed for maximizing the classification accuracy of the selected feature set. This study also showed GA selected a small subset of 5 features with the resulting classification accuracy of 97.7% and GA outperformed other search methods such as the greedy and the ranking algorithms.

Another approach using the GA in the process of selecting relevant features was carried out Jourdan, Dhaenens, & Talbi (2001). In this approach, the procedure of FS was carried out using GA and k-means in 2 steps: 1) The GA was utilized for searching optimal features with the aim to select a small subset of features from datasets with a large number of features, 2) selected features from step 1 are used as initial input features for a k-means clustering algorithm to cluster the data. As a result, the execution time of the algorithm is much faster than using k-means without GA; from 7500 minutes down to 1 minute, and data were clustered effectively (Jourdan *et al*., 2001).

In the study of Sun, Babbs and Delp (2005), the GA was compared to Adaptive Sequential Forward floating search (ASFFS) method for FS. Both methods were evaluated using a small dataset of images of Breast cancer that consisted of 296 normal regions and 164 cancerous regions. As a result, ASFFS outperform GA in terms of ROC (Receiver Operating Characteristic) analysis (A_z). ASFFS achieved $A_z = 0.964$ and the GA achieved $A_z = 0.917$. The study concluded that, the GA application was more suitable for a large dataset, while ASFFS performed better for a small or medium dataset (Sun *et al*., 2005).

The GA was also applied to select a relevant set of features for a prostate protein MS dataset in the study of Li *et al* (2008). Multivariate filter and wrapper methods were used as objective functions in the GA to determine the fittest individual. With the multivariate filter method, an evaluation criterion is built based on the scatter matrix and Bhattachayya distance. With the wrapper method, an evaluation function is built based on classification error rate and the posterior probability. This study achieved 92.7% classification accuracy for the multivariate filter method and 97.75% for the wrapper method. These results showed that the GA based multivariate filter and wrapper methods as its objective functions improved the classification accuracy when compared to other FS methods such as PCA and sequential selection methods (Y. Li et al., 2008).

2.2.1.2. Multi-objective evolutionary algorithms

In the study of Deb and Reddy (2003), a MOEA, called non-dominated genetic algorithm (NSGA2), was implemented with binary encoding representation to find multiple optimal feature sets for microarray cancer datasets: Colon, Leukemia and Lymphoma. Three objective functions, f_1, f_2, f_3, were implemented in their approach. f_1 is for the size of gene subsets, f_2 is for the number of mismatches (errors) in the training dataset and f_3 is for the number of mismatches (errors) in the test dataset. The proposed approach, NSGA2, obtained 352 different three-gene sets that gave 100% classification accuracy. In addition, NSGA2 was employed in an approach where a local search strategy was incorporated into a MOEA in the study of Mitra and Banka (2006) for performing *biclustering* on yeast and human B-cells datasets.

Rough sets and fuzzy set-based approaches for FS have also been combined with MOEAs to select features and classify high dimensional datasets in the domain of bioinformatics (Banerjee *et al.*, 2007; S. Mitra & Hayashi, 2006). Banerjee *et al.* (2007) proposed an evolutionary rough set based FS technique for analysing gene expression data. The new FS approach was based on the RST with the application of MOEA to search for optimal subsets. NSGA2 was employed as a MOEA to optimize 2 objective functions simultaneously and generated a set of multiple optimal solutions. RST was employed to generate a *distinction* table of smaller sets of relevant features and used as initial inputs for NSGA2 to search for multiple optimal solution sets. This approach was evaluated using Colon, Lymphoma and Leukemia microarray cancer datasets. As a result, the number of relevant selected genes was smaller compared when to other selection methods, such as neural networks, t-test based FS and SVM, and also the accuracy of the classification was still retained at a very high level. This achievement is due to the fact that, RST was used to generate reducts in the form of small subsets of relevant features initially and then NSGA2 optimized the reducts to find the best subset (minimal reducts) of relevant features with highest classification accuracy (Banerjee *et al.*, 2007).

2.2.1.3. Memetic algorithms (MAs)

Zhu, Ong and Dash (2007) used a MA to search for relevant features for the Colon, Central Nervous System, Leukemia, Breast, Lung and Ovarian microarray cancer datasets. They proposed the Wrapper-Filter Feature Selection Algorithm (WFFSA) and Markov Blanket-Embedded Genetic Algorithm (MBEGA) which involves MA. Both approaches were based on the traditional GA and a local search (LS) algorithm such as ranking filter method for WFFSA and Markov Blanket for MBEGA. In these approaches, binary representation was used for encoding chromosomes and the SVM classifier was employed to evaluate the fitness of individuals in the population. The MA based approach outperformed GA in terms of a faster convergence and smaller feature subsets with higher classification accuracies.

Recently, Kannan and Ramaraj (2010) employed MA with a correlation based filter ranking method as the LS algorithm and the Naïve Bayes classifier as a fitness evaluator to evaluate the fitness of feature subsets. In their approach, binary representation was used for encoding chromosomes, Subset Size-Oriented Common Feature method was used for crossover and random bit flip method was used for mutation. The proposed approach outperformed the other search algorithm such as GA and ReliefF-based GA in terms of obtaining smaller feature subsets and higher classification accuracy.

2.2.2. Feature selection using RST

An approach incorporating a greedy search algorithm into RST for selecting relevant features was proposed by Zhong, Dong, and Ohsuga (2001). In their approach, RST was first used to generate reducts (sets of minimal features), which were evaluated using the *Generalization Distribution Table* and the R*ough Set* theory (GDT-RS) rules discovery system (Dong, Zhong, & Ohsuga, 1999; Zhong, Dong, & Ohsuga, 1998). GDT is used to evaluate the goodness of a rule and the RS theory is used to find the best rule. A set of *indispensable* features is called "CORE" and cannot be eliminated from the feature list, and can be also used to classify data. (Zhong *et al.*, 2001). A greedy search strategy was employed to search for optimal reducts from the reducts generated from the RST step. Firstly, features (reducts) obtained from the RST step were used as initial feature inputs for the greedy search algorithm, and then the greedy algorithm finds relevant features from the feature list using GDT-RS rules for feature evaluation. Features selected are then added to the reduct until the set of optimal features are obtained. As a result, the proposed approach selected the optimal set with 4 features for the Breast cancer data, and 17 and 19 features for gastric cancer data (Zhong *et al.*, 2001).

Midelfart *et al.* (2002) applied RST to select relevant feature sets and classify microarray gene expression data. High dimensional microarray data might contain irrelevant features that affect RST in terms of generating a large number of reducts of irrelevant features and therefore less accuracy in class prediction. In order to address this problem, Midelfart *et al.* (2002) used t-test statistics to measure features; first by calculating the centroid of each class for each attribute and then to measure the difference between them for any significance. Only the features with highest t-test statistics were selected as significant features and subsequently used as feature inputs for RST. The approach was applied to the gastric cancer data and the number of features obtained in the selected subsets are from this approach range from 17-161 features (Midelfart *et al.*, 2002).

A new RST approach, called *roughfication*, was proposed to handle real values for microarray data in the study of Ślezak and Wróblewski (2007). In the traditional RST approach, real data values must be discretised prior to applying RST to generate reducts and to classify data. The *Roughfication* approach creates a new information system (IS) which based on the original IS. The new system used symbolic values (instead of real values in the original system) during rule generation processes. The symbols are used to form decision rules and subsequently used to predict the class for new samples. This approach was evaluated using the Breast cancer dataset and results obtained were compatible with other classification approaches (Ślezak & Wróblewski, 2007).

2.2.3. Feature selection using NSC

One of the popular FS and classification algorithms in bioinformatics is the NSC algorithm (Tibshirani *et al.*, 2002) (due to its algorithm is simple and effective). It is also known as Prediction Analysis for Microarrays (PAM) which is a software implementation of the NSC. The NSC has been used in numerous studies (Arai & Barakbah, 2007; Klassen & Kim, 2009; Levner, 2005; Ray *et al.*, 2007; Rocha de Paula, Gómez Ravetti, Berretta, & Moscato, 2011; Tibshirani, Hastie, Narasimhan, & Chu, 2003).

Tibshirani *et al*. (2003) used the NSC to analyse the Small Blue Round Cell Tumours (SRBCT) and Leukemia datasets and obtained the set of 43 genes and 21 genes, respectively. The set of 43 genes constructed a classifier that achieved the classification accuracy of 100% and the set of 21 genes resulted in a higher classification accuracy when compared to analysis involving the same datasets in Golub *et al*. (1999) using 50 genes (Tibshirani *et al*., 2003).

Arai and Barakbah (2007) compared the NSC method with other classification methods such as Fisher's Linear Discriminant Analysis (FLDA), Logistic regression (LOGISTIC), k-NN, SVM, Penalized Discriminant Analysis (PDA) using the SRBCT, Lung NSCI60 and Yeast datasets they showed that the NSC algorithm outperformed these other methods in terms of classification accuracy.

In the study of Klassen and Kim (2009), the NSC algorithm was used to select features for 7 different microarray cancer datasets namely, the SRBCT, Acute Leukemia, Prostate, Lymphoma, Colon, Lung and MLL Leukemia datasets. From the analysis involving the application of NSC, 43 features were selected for SRBCT, 21 features for Acute Leukemia, 6 features for Prostate, 25 features for Lymphoma, 16 features for Colon, 5 features for Lung, and 52 features for the MML Leukemia datasets with 100%, 94.11%, 90.91%, 86.6%, 75%, 93.7% and 95.4% test classification accuracy, respectively.

Levner (2005) used the NSC algorithm to classify Ovarian (OC-H4, OC-WCX2a, OC-WCX2b) and Prostate (PC-H4, PC-IMAC-Cu) MS cancer datasets. The study experimented with the use of 20 different shrinkage threshold values ranging from 0.5 to 10 in increments of 0.5 to find the optimal shrinkage threshold. From their analysis, the average classification accuracy for the five datasets were 62.1% for OC-H4, 94.4% for OC-WCX2a, 97.2% for OC-WCS2b, 73.6% for PC-H4, and 76.4% for PC-IMAC-Cu. The study also experimented with 200 different shrinkage threshold values ranging from 0.5 to 10 in increments of 0.05 and obtained the same classification results.

Ray *et al*. (2007) used PAM to analyse their Alzheimer's disease dataset. From this analysis, a set of 18 proteins were selected from 120 proteins. The set of 18 proteins were used in the classification of test samples (Alzheimer's disease, Non-demented control (NDC), mild cognitive impairment (MCI)). The result from the analysis was an overall 89 % classification accuracy. The performance of PAM was better than other algorithms such as GA-ANN by Cho *et al*. (2003) (Ray *et al*., 2007). Following the discovery of the 18 protein biomarker from Ray *et al.'*s study (2007), Ravetti and Moscato (2008) and de Paula, Ravetti, Berretta, and Moscato (2011) also used the NSC algorithm to perform classification on the same Ray *et al.'s* Alzheimer's disease dataset (2007).

Many approaches have also been proposed for modifying the NSC algorithm with the aim of improving its performance. For example, Yeung and Bumgarner (2003) developed the uncorrelated shrunken centroid (USC) and the error-weighted, uncorrelated shrunken centroid (EWUSC) algorithms which are based on the NSC algorithm. The proposed algorithms removed redundant, correlated genes which reduced the number of features needed for classification. These algorithms were applied to different types of cancer datasets such as Colon, Leukemia and Ovarian. The results showed improvements in the classification accuracy and also in a smaller number of relevant features. S. Wang and Zhu (2007) proposed 2 methods, Adaptive L_∞-norm Penalized NSC (ALP-NSC) and Adaptive Hierarchically Penalized NSC (AHP-NSC) with 2 different penalty functions. ALP-NSC method penalizes the maximum absolute relative difference ($|d_{ik}|$) between the class centroid and overall centroid for the i^{th} gene, if the maximum absolute $|d_{ik}|$ is shrunken to 0 then all d_{ik} are automatically shrunken to 0. ALP-NSC also penalizes each gene differently by using a pre-defined weighting scheme (w_j); w_j is small (i.e. less penalty applied) for genes that distinguishes different classes, and w_j is large (i.e. more penalty applied) for genes that are similar and do not distinguish different classes. AHP-NSC penalizes the relative difference (d_{ik}) hierarchically, i.e., within i^{th} gene, different levels of d_{ik} are applied. The proposed methods were used to analyse the Leukemia. Their study showed ALP-NSC and AHP-NSC outperformed NSC in terms of selecting smaller sets of features with similar classification accuracy.

Although there is extensive work involving the NSC, both from using it to analysis and from modifications for improvements, a major drawback is the determination of the shrinkage threshold value. This value is still being manually selected using CV or empirical methods. In addition, this value impacts on FS and classification in NSC. This drawback limits the NSC algorithm to perform its best, owing to the fact that if incorrect or sub-optimal shrinkage threshold values are provided to NSC, then the algorithm does not perform fully at its best in selecting optimal feature subsets and subsequently can lead to a lower classification accuracy. Thus, it is essential to develop methods that can automatically find the shrinkage threshold values for NSC. That is, the process of selecting the shrinkage threshold value is carried out automatically using the respective training data. Subsequently, the optimal shrinkage threshold value obtained from the automated process is used in the NSC algorithm to perform FS and classification. This would overcome the existing drawback of the NSC algorithm. The following section describes some of the hybrid approaches that incorporate a classifier and an EA for selecting relevant features.

2.2.4. Feature selection using hybrid approach

A hybrid approach that has been used to optimize the search for feature subsets is to incorporate an EA (e.g. GA) with another algorithm (e.g. SVM) (Pujari, 2001). The following section describes some studies that used hybrid EA approaches in FS and classification.

In Peng, Xu, Ling, Peng and Du's study (2003) study, GA was used in conjunction with SVM to select features from 2 datasets, namely the NCI60 and GCM cancer datasets. Unlike other search strategies that search for the best feature one at a time, GA searches for subsets of features in high dimensional data, hence the algorithm is able to select a small feature subset with a high accuracy of classification. The results of applying the approach to 4 cancer datasets (Colon, leukemia, NCI60 and GCM) has shown that, the algorithm is able to find a smaller subset of relevant genes that produces a higher classification accuracy than previous methods such as rank-based gene selection and all paired binary SVM (AP-SVM) (Peng *et al*., 2003)

Cho *et al*. (2003) proposed an approach that incorporated GA and ANN for selecting relevant features to classify an Alzheimer's disease dataset of 32 samples with 118 features. An initial feature subset was generated, each feature in this subset was evaluated using ANN to determine their fitness. GA performed FS based on the fitness of the individuals. Only dominant features from each generation were selected. These selected features were used as a relevant feature subset to input for the neural network, which increased the network efficiency. Experimental results showed that 35 features were selected from 118 features, and the classification accuracy was 81.9% on the test data. GA was able to select relevant features to classify Alzheimer's disease data from non- Alzheimer's disease data, which was very useful for early detection of the disease (Cho *et al*., 2003)

Jirapech-Umpai and Aitken (2004) proposed a hybrid EA approach for multiclass classification. The approach combines GA and k-NN with the use of 6 ranking methods (Information gain, Twoing rule, Gini index, Sum minority, Max minority and Sum of variances) as fitness selection method to determine best features for GA. In the study, binary representation was used for chromosomes in the GA population, k-NN was employed as a measure function between samples using Euclidean distance. The proposed algorithm of GA and k-NN was evaluated using 2 microarray datasets: Leukemia and NCI60. The approach selected sets of features with 92% - 98% and 76.23% classification accuracy on the 2 datasets, respectively (Jirapech-Umpai & Aitken, 2004).

In the study of Li Li et al. (2005), the combination of GA and SVM (GA-SVM) has also been implemented to select an optimal subset of genes. The proposed GA-SVM used the power of GA for searching relevant features, and the SVM classifier to evaluate the *goodness* of feature subsets. The approach was applied to a diffuse large B-cell lymphoma (DLBCL) microarray dataset. From the analysis, 99% classification accuracy was obtained, which outperformed other FS methods such as the combination of GA and k-NN (GA-kNN), and filter methods (t-test, non-parametric scoring) (Li Li et al., 2005)

Lu, Tian, Neary, Liu, and Wang (2008) proposed a hybrid FS approach, incorporating GA to improve FS on 2 microarray datasets: Colon and Prostate cancer dataset. The hybrid algorithm uses the features selected from other selection methods: 2 from filter based methods (entropy-based and T-statistics) and 1 from a wrapper method (SVM-RFE). The features selected from these three methods are combined together to form a feature population. GA uses this feature population as an initial population to start with and to produce an optimal (or near optimal) subset with a smaller size, but more accurate in prediction. The result from the study shows that, hybrid FS with GA is more effective, efficient and accurate in selecting small subsets than the other FS methods mentioned above. The study also found that top-ranked features do not necessarily give more accuracy than the lower-ranked features because interaction, correlation and redundancy between features are to be considered when classifying the data (Lu *et al.*, 2008)

2.3. Techniques related to the implementations of proposed approaches in the study for FS and classification

2.3.1. K-Means

The *k*-means clustering algorithm was proposed by MacQueen (1967) . It is one of the most commonly used clustering algorithms for grouping data into different clusters for large datasets (Huang, 1998). The following figure illustrates a basic k-means algorithm.

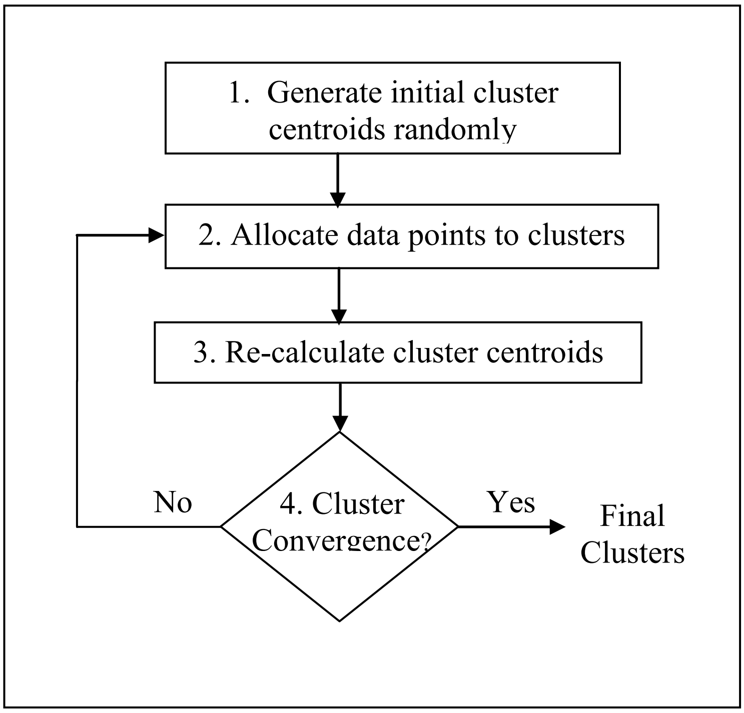

1. Generate initial cluster centroids randomly
2. Allocate data points to clusters
3. Re-calculate cluster centroids
No
4. Cluster Convergence?
Yes
Final Clusters

Figure 2-3 Basic k-means algorithm

As seen in Figure 2-3, step 1 is to generate initial centroids randomly for the k clusters, i.e., one centroid for one cluster; in step 2, each data point is placed into the cluster that has a closest centroid to the data point; step 3 is to re-calculate the new centroid for each cluster using the new data points and step 4 is to check for cluster convergence. Step 2 and 3 are repeated until the cluster centroids do not change, i.e., convergence takes place.

2.3.2. Rough Set Theory

It is common for datasets to contain decision variables (classes) which cannot be used to differentiate the samples. For example, two or more samples have the same attribute values but belong to different classes and therefore the samples cannot be assigned correctly to the class they belong based on values for these types of variables. This causes problems in classification when the classifier tries to classify data to a certain class. A rough set (RS) approach was proposed by Pawlak (1982). This approach was developed on a mathematical basis and could be used to classify indiscernible data. The RS approach has also been used effectively in FS (Hu, Yu, Liu, & Wu, 2008; Pujari, 2001; Swiniarski, 2001). According to Han, Kamber and Pei (2006), the RS is based on *equivalence classes* containing samples that are identical in terms of attributes describing the data. The RS classifies a class by using a lower approximation and an upper approximation for the class. The lower approximation for the class consists of all the samples that can be described as definitely belonging to the class, "*positive cases*", whilst the upper approximation for the class consists of all the samples that are described as possibly belonging to the class, "*possible cases*" (Pujari, 2001, p. 57).

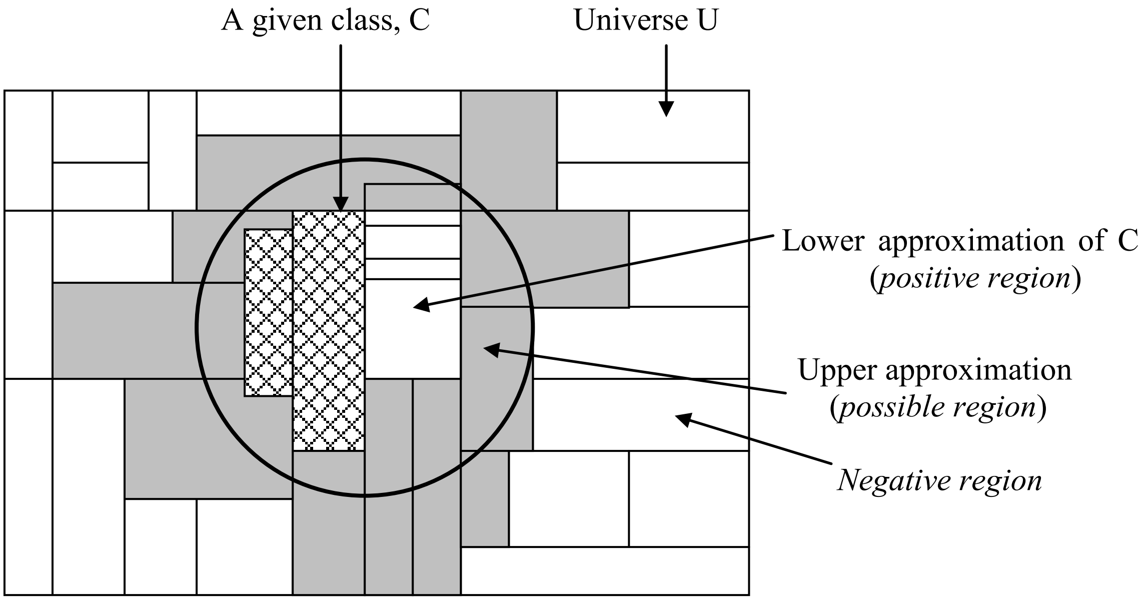

A given class, C
Universe U
Lower approximation of C (*positive region*)
Upper approximation (*possible region*)
Negative region

Figure 2-4 Rough set with lower and upper approximation of a given class, C, adapted from Han, Kamber and Pei (2006, p. 352) and Hu *et al*. (2008, p. 3582).

The circle in Figure 2-4 represents a given class (C) that consists of the outlined cross hatched rectangular region (*positive region*) as a lower approximation, shaded rectangular region (*boundary region*) as an upper approximation. Each rectangle of the positive and boundary region represents an equivalence class. The samples of the positive region are identified as belonging to C; whilst the samples of the boundary region partly covered by C (i.e., samples with similar feature values which belong to more than one class) are possibly belonging to C, but that status cannot be verified with certainty. All the samples outside the boundary in the white rectangular region (*negative region)* are definitely not belonging to C.

The lower approximation of class C: $\underline{A}C$= $\{x\colon [x]_A \subseteq C\}$ where $[x]_A$ is an equivalence class. Thus all the samples in the equivalence classes are in C. The upper approximation of class C: $\overline{A}C$= $\{x\colon [x]_A \cap C \neq \emptyset\}$. Thus not all the samples in the equivalence classes are in C. The result of the intersection between equivalence classes and C is a non-empty set. The boundary region is the region of the difference between lower and upper region and is calculated as Boundary AC = $\overline{A}C$- $\underline{A}C$. The boundary region indicates the roughness of C. The smaller boundary region has the better confidence in classification. The negative region is the region outside the upper approximation region, NC = U-$\overline{A}C$.

The accuracy of the rough set is calculated by dividing the lower approximation by the upper approximation (lower/upper).

Rough set can also be used to as a pre-processing approach to eliminate a number of redundant attributes for a high dimensional data based on the equivalence classes, lower and upper approximation (Jaaman, Shamsuddin, Yusob, & Ismail, 2009), prior to applying a FS or/and classification technique to select optimal feature subsets to classify data more effectively.

2.3.3. Nearest Shrunken Centroid algorithm

As mentioned earlier, the NSC algorithm has been used widely in bioinformatics as a FS and classification technique to select the most relevant features and to classify high dimensional biomedical data, e.g., Leukemia data. The following section describes the NSC algorithms in details.

The NSC algorithm shrinks the class centroid for each feature (gene) toward the overall centroid for all classes by an amount of shrinkage threshold, Δ. The class centroid $\bar{x}_{ik}$ for class *k* for gene *i* is calculated using Equation (2.4).

$$\bar{x}_{ik} = \sum_{j \in C_k} x_{ij} \,/\, n_k \qquad (2.4)$$

where x_{ij} is a gene expression value for gene *i* =1...*p* and sample *j* = 1...*m*, C_k is an index of n_k samples in class *k*.

The overall class centroid $\bar{x}_i$ for gene *i* is calculated using Equation (2.5).

$$\bar{x}_i = \sum_{j=1}^{n} x_{ij} \,/ n \qquad (2.5)$$

The relative difference, d_{ik} is the difference in class centroid, $\bar{X}_{ik}$ and the overall class centroid, $\bar{x}_i$, standardized by the within class standard deviation of gene *i*, s_i. The formula for calculating relative difference d_{ik}, is defined by Equation (2.6).

$$d_{ik} = \frac{\bar{x}_{ik}\text{-}\bar{x}_i}{m_k(s_i+s_o)} \qquad (2.6)$$

where $m_k = \sqrt{\frac{1}{n_k}+\frac{1}{n}}$

s_0 = median value of s_i over all genes

The relative difference, d_{ik} is evaluated to 0 if it is equal to 0 or smaller than the threshold, Δ, else reduce d_{ik} by the threshold, Δ. The updated d_{ik} is called a shrunken relative difference, d'_{ik}. The calculation for d'_{ik} is shown in Equation (2.7).

$$d'_{ik} = sign\ (d_{ik})(|d_{ik}| - \Delta)\ \text{ if } |d_{ik}| > \Delta\text{. Otherwise 0} \qquad (2.7)$$

Class centroid for gene i is updated by using the new value of d'_{ik} as shown in Equation (2.8).

$$\bar{x}'_{ik} = \bar{x}_i + m_k\,(s_i + s_o)\ d'_{ik} \qquad (2.8)$$

If a gene is shrunk to zero for all classes, then it is considered not different from the overall centroid (i.e. irrelevant genes from a classification point of view) and is eliminated from the gene list (Klassen & Kim, 2009) Genes with at least one positive shrunken relative difference (over all classes K) are retained as relevant attributes (K. Yeung & R. Bumgarner, 2003). Selected attributes are then evaluated by calculating the discriminant score for class k for a new sample $X^* = \{x^*_1, x^*_2, \ldots, x^*_p\}$, as shown in Equation (2.9).

$$\delta_k(x^*) = \sum_{i=1}^{p} \frac{(x^*_i - \overline{x'}_{ik})^2}{(s_i + s_o)^2} - 2\log\pi_k \qquad (2.9)$$

The first part of Equation (2.9) is the standardized squared distance of x^* to the k^{th} shrunken centroid, and the second term of Equation (2.9) is a correction based on the class prior probability π_k, where $\pi_k = n_k/n$.

Based on the discriminant score for each class, sample x^* is classified to the class k that has a minimal discriminant score defined by Equation (2.10).

$$C(x^*) = \ell \quad \text{if} \quad \delta_{\ell}(x^*) = \min_k \ \delta_k(x^*) \qquad (2.10)$$

where $C(x^*)$ is an assigned class of sample x, $\delta_k(x^*)$ is a class discriminant score, $\delta_\ell(x^*)$ is a minimal class discriminant score.

The general steps of NSC algorithm are shown in the following figure.

Step 1. Calculate class centroid for attribute (gene) i of class k

$$\bar{x}_{ik} = \sum_{j\in C_k} x_{ij}/n_k$$

Step 2. Calculate overall centroid for all classes ($\bar{x}_i$)

$$\bar{x}_i = \sum_{j=1}^{n} x_{ij}/n$$

Step 3. Calculate the relative difference ($\mathrm{d_{ik}}$)

- Calculate class standard deviation of attribute ($\mathrm{s_i}$)

$$\mathrm{s_i}^2 = \frac{1}{n-K} \sum_k \quad \sum_{j\in Ck} \left(\bar{x}_{ij} - \bar{x}_{ik}\right)^2$$

- Calculate $\mathrm{s_o}$, median value of $\mathrm{s_i}$ over all attributes
- Calculate $m_k = \sqrt{\frac{1}{n_k} + \frac{1}{n}}$
- Calculate relative difference $d_{ik} = \frac{\bar{x}_{ik} - \bar{x}_i}{m_k(s_i + s_o)}$

Step 4. calculate the shrunken relative difference (d'_{ik})

```
if $|d_{ik}|$ > threshold ($\Delta$)
    while $|d_{ik}|$>threshold ($\Delta$)
        $|d_{ik}| = |d_{ik}|-\Delta$
    $d'_{ik}$ = sign($d_{ik}$) ($|d_{ik}|$)
else
    $d'_{ik}$=0
```

Step 5. Update class centroids for attribute i

$$\bar{x}'_{ik} = \bar{x}_i + m_k\,(s_i + s_o)\; d'_{ik}$$

Step 6. Repeat step 1 to 5 until all attributes are processed

Step 7. Select relevant attributes with at least one positive shrunken relative difference (d'_{ik}) over all classes

Step 8. Evaluate the set of relevant attributes selected

- calculate discriminant score for class k for a new sample (x^*)

$$\delta_k(x^*) = \sum_{i=1}^{p} \frac{(x^*_i - \bar{x}'_{ik})^2}{(s_i + s_o)^2} - 2 \log \pi_k$$

where x^* is a sample with attribute values $x^*_1,\ x^*_2,..,\ x^*_p$

$$\pi_k = = n_k / n$$

- Assign sample x^* to the class k that has a minimal discriminant scores:

$$C(x^*) = \ell \quad \text{if} \quad \delta_\ell(x^*) = \min_k\ \delta_k(x^*)$$

Figure 2-5 Steps of the NSC algorithm

2.3.4. Evolutionary Algorithm

Evolutionary algorithm (EA) is a search method, based on the principle of survival of the fittest which was borrowed from the evolution of biological nature. Basically, a number of generations are iterated through EA; each generation consists of a numerous individuals. The later generations contain fitter individuals which maybe a subset of previous generations. Only individuals which survive as the fittest are retained from generation to generation, and the fittest individual subset is selected at the end of the process. GA and MOEA are the 2 typical types of EA which are described in the following section.

2.3.4.1. Genetic Algorithm

GA was proposed and developed by Holland (1975) and is based on Darwin's theory of survival of the fittest. GA consists of components such as population representation, objective function, evaluation of population, selection, crossover, and mutation operators. Figure 2-6 shows the steps in a GA.

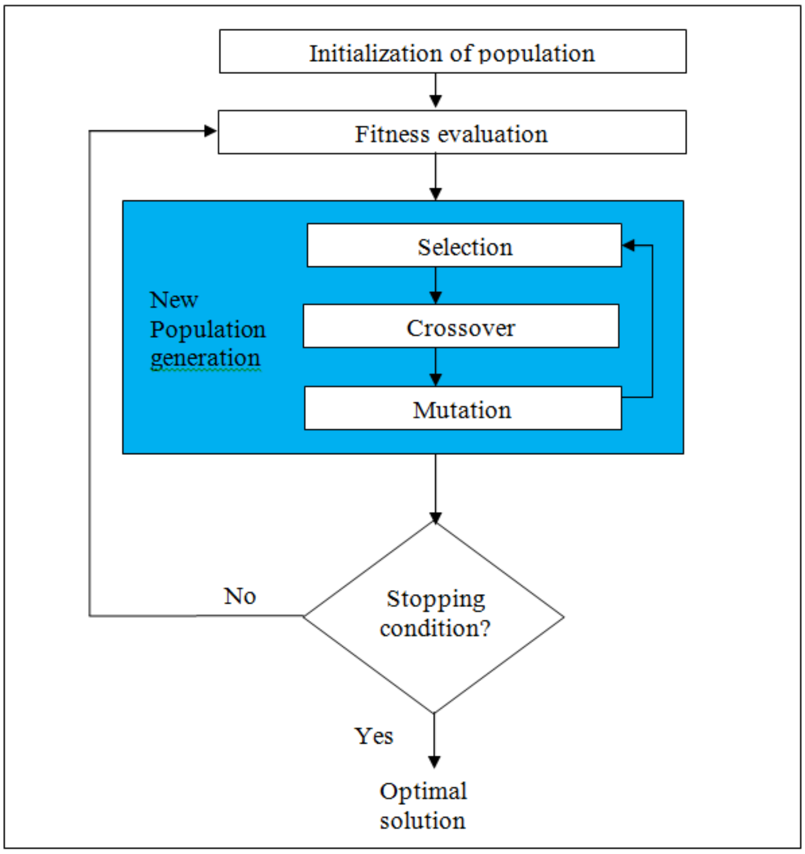


Figure 2-6 Basic steps of GA

- Initial population

Individuals in the population are randomly generated.

- Fitness evaluation

The algorithm uses an objective function (s) to evaluate the fitness of individuals in the population.

- Selection

Once the process of ranking the fitness of individuals is done then the selection of individuals is carried out in order to find which individuals will be combined to produce offspring. Many selection techniques have been used in the area of GA and these include: Ranking selection, Roulette Wheel selection and Tournament selection. According to Miller and Goldberg (1995), an ideal selection technique is the technique that would be simple in implementation, efficient in performance and adaptable in different domains. Tournament selection has been widely used in GA because of its usefulness and robustness, and it satisfied all the criteria mentioned above (Miller & Goldberg, 1995).

Tournament selection is also known as a *random tournament selection* that selects *k* number of individuals randomly from the population pool to form a tournament group of size *k* and the fittest individual from the group is then selected for crossover (Goldberg & Deb, 1991; Hoefsloot, 2013; Miller & Goldberg, 1995). A *binary tournament selection* is a special case of *random tournament selection* in which, the size of the tournament group is 2. That is, two individuals are selected randomly from the population to form a tournament group of size 2 and the best individual of the group (i.e. the best of the two) is selected (Deb, Pratap, Agarwal, & Meyarivan, 2002; Suzuki, Takahashi, & Shibahara, 1995). The following figure describes the general process of selecting individuals using tournament selection.

```
Input:
        Chromosome population ($p$)
        Fitness population ($F_p$)
Output:
        Selected parent chromosomes ($C_{parents}$)
Steps:
  1. Set $k$ = size of tournament
  2. Set Tournament ($T_k$) = {∅}
  3. Set $n$ = max number of parents to be selected
  4. For counter from 1 to $n$
        a. For counter from 1 to $k$
                • Select chromosomes randomly from $p$
                • Store selected chromosomes into $T_k$
        b. Compare fitness of individuals in $T_k$ using $F_p$
        c. Select a chromosome with the best fitness ($C_{best}$)
        d. Store $C_{best}$ into $C_{parents}$
```

Figure 2-7 Tournament selection procedure

- Crossover (recombination)

The crossover process combines two or more selected parents from previous steps to produce offspring. This method depends on the type of chromosome representation, e.g., binary crossover (crossover with single, double, multi points, uniform and arithmetic) is used for binary representation. Generally, the steps in crossover involve: 1) two parent candidates are selected for crossover, 2) a crossover parameter is used to determine whether the crossover operation will take place, 3) a random number is drawn, i.e., for one point crossover, in the case of multi points crossover then more than one random number need to be drawn, to determine the position (s) where the crossover take place on the parents, 4) the parents are crossed over at the randomly selected position(s) to produce the new individuals (offspring). The following figure illustrates one point binary crossover between two parents which is related to the study.

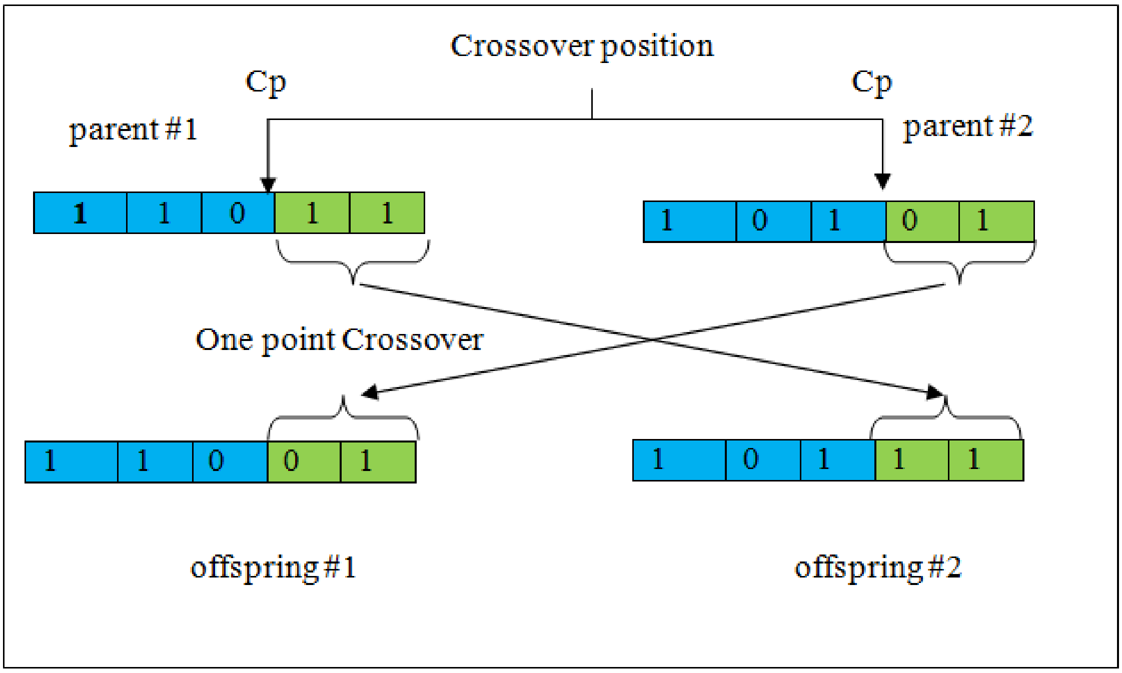


Figure 2-8 Example of a one point binary crossover

As seen in Figure 2-8, two parents are split into ‘head’ (in blue) and ‘tail’ (in green) at the cross position (Cp), the ‘tails’ of 2 parents are inter-changed to produce 2 offspring.

- Mutation

After the crossover process, offspring are mutated to produce new individuals with different features which are not present in their parents. According to Eiben & Smith (2007), mutation operators can be applied for binary, integer, real and permutation encoding representations. The following section describes the bit flip mutation procedure for binary representation and uniform mutation procedure for real-value representation, both of which are used in this study.

Bit flip mutation for binary representations:

A bit flip mutation is the type of mutation where each bit in the chromosome is allowed to change its value independently with a small mutation probability (P_m). That is, if the random number generated for the bit is less than P_m and if the bit is 1 then it changes (flips) to 0 or if the bit is 0 then it changes to 1 (Eiben & Smith, 2007).

```
Input:
        Chromosome (chrom)
        Mutation probability ($P_m$)
Output:
        Modified chromosome ($chrom_{Mod}$)
Steps:
   1. Set len = length of chrom
   2. Generate a random number ($R_n$) in the range [0, 1] using a
      random number generator (RNG)
   3. If $R_n < P_m$
        For counter =1 to len
            Generate a random number ($R_n$) in the range [0, 1] using RNG
            If $R_n \leq P_m$
                Do bit flip on chrom [counter]
        $chrom_{Mod}$ = chrom
   4. Else
        No mutation
```

Figure 2-9 Bit flip mutation procedure

The algorithm for Uniform mutation for real-encoding representations is shown in Figure 2-10.

```
Input:
        Chromosome (chrom)
        Mutation probability ($P_m$)
Output:
        Modified chromosome ($chrom_{Mod}$)
Steps:
     1. Set len = length of chrom
     2. Generate a random number ($R_n$) in the range [0, 1] using RNG
     3. If $R_n < P_m$
        Find the lower bound value of chromL$_b$
        Find the upper boundary value of chromU$_b$
        For counter =1 to len
            Generate a random number ($R_n$) in the range [0, 1] using RNG
            If $R_n \leq P_m$
                Calculate chrom[counter] =chromL$_b$ + ($R_n$ * (chromU$_b$ - chromL$_b$))
        $chrom_{Mod}$ = chrom
    4. Else
        No mutation
```

Figure 2-10 Uniform mutation procedure

- New population generation:

Best offspring from selection, crossover and mutation process is placed into the new generation. The process of selection, crossover and mutation are repeated until the new generation of the population is completed.

- Termination:

The process of fitness evaluation, crossover, mutation and new population generation are repeated until a stopping condition is met. For example, such as a solution is found after a pre-defined number of iterations.

2.3.4.2. Multi-objective evolutionary algorithms

In the real world, tasks are normally associated with multiple conflicting objectives such as the conflict between performance, cost, fuel efficiency, reliability, etc., For example, a car that performs well but consumes less fuel and is of a reasonable price. There is no single best solution that satisfies multiple conflicting objectives simultaneously, rather a set of solutions with trade-offs between conflicting objectives. Multi objective algorithms use more than one objective functions to optimize a problem. MOEA solves the problem effectively by dealing with multiple conflicting characteristics represented by the objective functions, and generates a set of optimal solutions, e.g., Pareto front of optimal solutions, which are the set of all non-dominated solutions (Ayala & Coelho, 2008).

MOEA is classified on the basis of its selection approach. There are three different types of MOEA (Coello & Lamont, 2004):

- Aggregating function approach which combines all the objective functions into a single objective function. The weighting (w) of each objective function, which indicates the importance of one objective function over the others, is used in this approach, e.g., $F= w_1f_l +w_2f_2 +\dots w_nf_n$. The limitation of this approach is that it does not give a set of different possible best solutions to satisfy all objectives, rather than one general solution for all objectives.
- Population based approaches which use the population to improve the diversity of the search but not incorporating the concept of Pareto front in the selection process.

Vector Evaluated Genetic Algorithm (VEGA) (Schaffer, 1985) is a typical example for this type of approach. At each generation, sub-populations are created based on objective functions, i.e., each objective function is used in turn in the selection process to generate a subpopulation of size of the total population size (M) over a number of k objectives. A new population of size M is then created from these subpopulations. Genetic algorithm evolves the new population with the use of selection scheme, crossover and mutation operator. The drawback of this approach is that if an individual has a good overall fitness for all objectives but is not the best individual for any individual objective, then it is discarded.

- Pareto based approaches which incorporate the concept of a Pareto front into MOEA. The objective function used to search for a Pareto Front which is a set of optimal solutions is defined by Ayala and Coelho (2008), as follows.

$$\text{Optimized } F(x) = (f_1(x), f_2(x), \ldots f_n(x))$$

where n= 1, 2,.., k; decision variables x = (x_1, x_2, …, x_n) ∊ X; X = feasible solution set; f_n are objective functions. The concept of a Pareto front is discussed in Chapter 8.

NSGA2 incorporates the concept of Pareto front into MOEA (Deb *et al.*, 2002). This study uses NSGA2 in the approach of incorporating NSGA2 into NSC for finding multiple optimal shrinkage thresholds. NSGA2 will be described in Chapter 8.

2.3.5. Memetic Algorithms (MAs)

MAs are similar to evolutionary algorithms such as GA. A common definition of MA is "*A memetic is an Evolutionary Algorithm that includes one or more local search phases within its evolutionary cycle*" (Krasnogor & Smith, 2005, p. 2). Gene values in GA are known as *memes* (Dawkins, 2006) in MA. The term, *meme*, is referred to a *unit of culture evolution* or *transmission* where the local improvement for chromosomes takes place using local search (LS) algorithms such as hill climbing (Elbeltagi, Hegazy, & Grierson, 2005; Wu, 2001a). Thus MA is a hybrid of EAs which combines an EA and a local search (LS) to improve the fitness of chromosomes (Krasnogor & Smith, 2005; Wu, 2001a).

MA has additional steps of LS for improving the fitness of chromosomes in the population by finding local optimum neighbours in chromosomes prior to the normal process of crossover and mutation operations. Each new population is evolved locally using LS and then globally via GA. This cycle repeats until the stopping criteria such as global convergence takes place or the pre-defined number of generations has been executed.

The combination of GA and LS makes MA more efficient and effective in terms of processing time for converging to optimal solutions, finding smaller sets of features, and improving classification accuracy when compared to other traditional EAs such as GA (Elbeltagi *et al*., 2005; Zhu *et al*., 2007). Different LS strategies such as pair-wise LS (Merz & Freisleben, 1999), improvement first strategy LS and greedy LS (Zhu *et al*., 2007) can be incorporated into GA in different ways. A LS strategy can be applied to

- only elite chromosomes or
- the entire population, or
- either after the crossover and/or mutation operation

The following section describes some strategies of implementing MA with different LS methods.

Elbeltagi *et al*. (2005) described a LS using pair-wise swapping proposed by Metz and Freislenben (1999). A swapping strategy to interchange 2 memes (genes) was applied to chromosomes in order to find the best local neighbour in the chromosome. Figure 2-20 illustrates the use of this pair-wise swapping strategy.

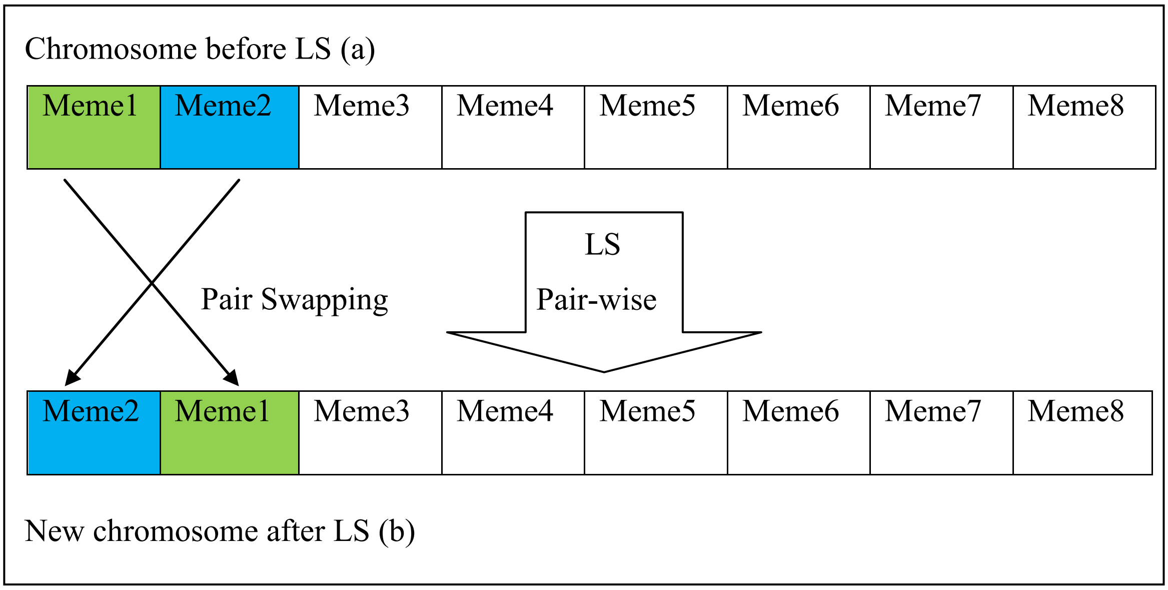


Figure 2-11 An example of LS using pair-wise strategy

As seen in Figure 2-11, the pair of 'Meme1' and 'Meme2' of chromosome (a) is swapped to form a new chromosome (b). The process of swapping between the pair continues for the remaining pairs, e.g., pair of Meme1 and Meme3, Meme1 and Meme4, ..., Meme1 and Meme8, pair of Meme2 and Meme3, Meme2 and Meme4, ..., Meme2 and Meme8, and so on. The number of pairs (N) to be swapped is calculated using Equation (2.11).

$$N = \frac{1}{2} (n (n\text{-}1)) \qquad (2.11)$$

where *n* is the length of chromosome.

For example, let chromosome A has the length, *n* = 1000, then N = ½ (1000(1000-1)) = 499500. That is, LS needs to process 499500 pair-wise operations. This could involve a large computational time when using this LS strategy. According to Elbeltagi *et al.* (2005), in order to reduce the cost of computational time, the swapping between pairs stops as soon as the fitness of the chromosome is improved (Merz & Freisleben, 1999). This is known as an *improvement first strategy* (Zhu *et al.*, 2007), i.e., no need to continue performing the swap for the remaining pairs once the first improvement of the chromosome has been found. The procedure of pair-wise LS with improvement first strategy is described below.

```
Input
  Chromosome (chrom)
  Fitness of chrom
Output
  Improved chromosome ($chrom_{Imp}$)
Steps
  1. Calculate the number of pairs of memes, $N=1/2 (n (n-1))$
  2. For counter =1 to N
       Swap the positions of the pair to create a new chromosome ($chrom_{new}$)
       Evaluate the fitness of $chrom_{new}$
       If the fitness of $chrom_{new}$ > fitness of chrom
            $chrom_{Imp} = chrom_{new}$
            Stop swapping pairs and exit
       Process the next chromosome
```

Figure 2-12 Pair-wise LS with i*mprovement first strategy* used in Elbeltagi *et al*. (2005)

The *adding subtracting* LS strategy involves searching for a better chromosome in terms of fitness by adding or subtracting a small random value to a meme (gene) value in the chromosome to create a new chromosome. The fitness of the new chromosome is then evaluated, if an improvement is obtained then the new chromosome is retained otherwise discarded. The process continues for the rest of the memes in the chromosome (Elbeltagi *et al*., 2005). This is called a *greedy search strategy* (Zhu *et al*., 2007) or a *hill climbing search strategy* where the search progresses from the current chromosome to the one that has a better fitness (Kohavi & John, 1997; H. Wang, Wang, & Yang, 2009). According to Elbeltagi *et al*. (2005), MA using the *adding and subtracting* LS with a greedy strategy outperformed GA in terms of a better classification accuracy and processing time. The procedure of *adding and subtracting* LS with a greedy strategy (Elbeltagi *et al*., 2005) is described in Figure 2-13.

```
Input
  Chromosome ($chrom$)
  Fitness of chrom ($chrom_{fit}$)
  Population size ($S$)
  Chromosome length ($len$)
Output
  Improved chromosome ($chrom_{Imp}$)
Steps
   1. Generate a random value, $R_n$
   2. For $counter1$ =1 to $S$
         For $counter2$ =1 to $len$
            Add $R_n$ to $chrom[counter2]$ to create a new chromosome ($chrom_{new}$)
            Evaluate the fitness of $chrom_{new}$
            If fitness of $chrom_{new} > chrom_{fit}$
                 $chrom_{Imp} = chrom_{new}$
                 update $chrom = chrom_{new}$
            Else
                 subtract $chrom[counter]$from $R_n$ to create $chrom_{new}$
                 evaluate the fitness of $chrom_{new}$
                 if fitness of $chrom_{new} > chrom_{fit}$
                    $chrom_{Imp} = chrom_{new}$
                    update $chrom = chrom_{new}$
```

Figure 2-13 Procedure of greedy search strategy using *adding and subtracting* LS (Elbeltagi *et al*., 2005)

According to Zhu *et al.* (2007), the *improvement first strategy LS* outperformed the *greedy strategy LS*. Their study also found that when applying the *improvement first strategy LS* on a few of elite chromosomes, results obtained were better than those obtained from applying LS on all chromosomes.

Elbeltagi *et al.* (2005) proposed another MA approach where the LS is applied to offspring after the crossover or mutation process. In the Guided Local Search (GLS) Based Memetic Algorithm (Krasnogor & Smith, 2005), the LS is applied to offspring after the crossover and mutation operations.

2.3.6. Similarity distance measures

From the literature, it can be seen that different similarity measures have been used to cluster gene expression data into groups of similar genes and in classification. A similarity measures is used in the process for grouping genes into clusters, whereby genes in the same cluster are as similar as possible, and are very different from genes in another cluster. For example, Pearson correlation measure considers the correlation between two genes for measuring the similarity between genes. Similar genes have a positive correlation and are related (Leale et al., 2013). Genes that similar in expressions (i.e. close in similarity distance measure) are grouped into a cluster (class) (Deshpande, VanderSluis, & Myers, 2013). Genes in the same cluster are likely to be involved in the same cellular processes and biological functions (Paul & Maji, 2014). The set of selected features from a biological perspective implies that the level of expressions associated with the selected biomarkers differ significantly between disease and non-disease. There's little existing evidence as to which measure is most effective but previous studies have shown that the use of different similarity measures have an impact on the clustering/classification results. The following sections describe some commonly used similarity distance measures. These include Euclidean, Mahalanobis, Pearson correlation and Mass distance.

2.3.6.1. Euclidean distance

Euclidean distance satisfies the triangle inequality and is the most commonly used distance measure. It is the method of measuring the distance between 2 points based on Pythagoras' theorem ($A^2 = B^2 + C^2$). For example, points with 2 dimensions A $\{x_1, x_2\}$ and B $\{y_1, y_2\}$, the squared distance between A and B is the total of squared differences of coordinates between A and B. Hence the distance is the square root of $(x_1 - y_1)^2 + (x_2 - y_2)^2$, as shown in Equation (2.12).

$$EucliD_{AB} = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2} \tag{2.12}$$

Points in multi-dimensional space, e.g., A$\{x_1, x_2, .., x_n\}$ and B$\{y_1, y_2, .., y_n\}$, the Euclidean distance measure is calculated as follows:

$$EucliD_{AB} = \sqrt{\sum_{j=1}^{n}(x_j - y_j)^2} \tag{2.13}$$

Euclidean distance have also been used in classification to classify a sample with multiple features, $X\{x_1, x_2, .., x_n\}$ to a class. In this case, a sample is assigned to a class based on the distance between the sample and its class centroid. The Equation (2.13) is re-written as follows.

$$EucliD = \sqrt{\sum_{j=1}^{n}(x_j - \mu_{jk})^2} \tag{2.14}$$

Where μ_{jk} is the mean of class k of jth feature

x_j is the sample of jth feature

n is a number of features.

2.3.6.2. Mahalanobis distance

Mahalanobis distance is a popular method that has been used widely as a distance measure in clustering and classification (Wölfel & Ekenel, 2005). Mahalanobis distance (Mahalanobis, 1936) is the method of measuring the distance between the centroids of 2 classes or the distance between a variable and a class centroid. Unlike Euclidean distance, in which the different class densities are considered to be equal and only the distance from a data point to a class centroid is a criterion for classification, in Mahalanobis distance, the different class densities are taken into account when classifying data (McLachlan, 1999). Figure 2-14 illustrates Mahalanobis distance measure.

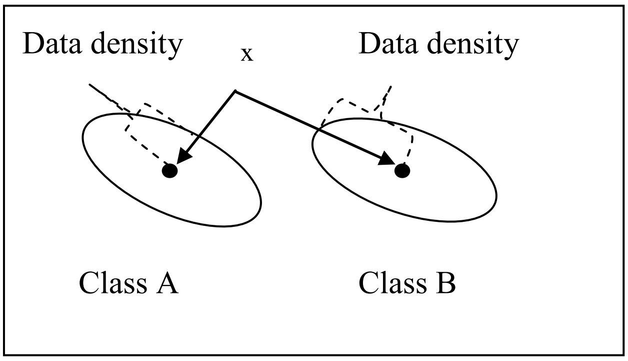


Figure 2-14 Mahalanobis distance measure

Variance of each variable and the co-variance between variables are taken into consideration in the Mahalanobis distance calculation. It handles problems associated with poorly and highly correlated features in a dataset.

Mahalanobis distance measure is calculated as follows.

$$MahaD^2 = (\boldsymbol{x} - \boldsymbol{\mu})^T \Sigma^{-1} (\boldsymbol{x} - \boldsymbol{\mu}) \tag{2.15}$$

where $\boldsymbol{\mu}$**:** class centroid

superfix T: matrix transpose

Σ^{-1}: inverse covariance matrix

2.3.6.3. Pearson distance

Pearson correlation (Pearson, 1895) is a method for measuring the correlation between 2 variables. The correlation is measured in the range of -1 to +1. +1 means the correlation is a perfect positive linear relationship, 0 implies an uncorrelated relationship and -1 is a perfect negative linear relationship. Pearson correlation is calculated as follows.

$$(r) = \frac{\Sigma (x-\bar{X})\,(y-\bar{Y})}{\sqrt{(x-\bar{X})^2}\,\sqrt{(y-\bar{Y})^2}} \tag{2.16}$$

$$\textit{Pearson Distance (}P_D\textit{)} = 1 - r \tag{2.17}$$

where x is variable value

$\bar{X}$ is class centroid

According to Equation (2.17), when r approaches 1, P_D approaches 0, i.e., the distance is 0, thus attributes have a linear relationship; when r approaches to 0, P_D approaches to 1 (P_D = 1-0), i.e., the distance is 1, thus attributes have an uncorrelated relationship; when r approaches to -1, P_D approaches to 2 (P_D =1-(-1)=2), i.e., the distance is 2, thus attributes have negative linear relationship.

The problem of using the Pearson distance as defined by Equation (2.17) is that the relationship between the distance and correlation coefficient is not mapping appropriately for measuring the correlation distance between variables. To address this, D. Wang, Wang, Lu, Song and Cui's study's (2010) used the absolute Pearson's correlation coefficients, | r | to measure the similarity for microRNA, P_D is calculated using | r | instead of r. Thus Pearson correlation distance is now calculated as follows.

$$P_D = 1 - |r| \tag{2.18}$$

According to Equation (2.18), when r approaches to 1 or -1 then | r | = 1, P_D approaches to 0 (P_D=1- |-1|), that is the distance is 0 and attributes have a positive or negative linear relationship; when r approaches to 0, P_D approaches to 1 (P_D = 1 - 0), that is the distance is 1 and attributes have an uncorrelated relationship.

2.3.6.4. Mass distance

Euclidean, Mahalanobis and Pearson distance do not consider the background distribution of attributes while calculating the distance (Yona *et al.*, 2006).

MD measure is a method that has been used for evaluating gene expression similarity and takes into account the background distribution of attribute values in the calculation of the distance (Yona *et al.*, 2006). Unlike the other measures such as Euclidean, Mahalanobis and Pearson, MD calculates the distance between two variables by measuring the relative difference between the variables and by measuring their probability mass (volume). Two variables are more similar (closer) when they have smaller volume (Yona *et al.*, 2006).

The equations used to calculate MD for 2 variables (a, b) are taken from Yona, *et al.* (2006).

Calculation of probability mass for 2 variables a and b for sample *i*:

$$MASS(_{a_i b_i}) = \int_{\min(a_i,b_i)}^{\max(a_i,b_i)} Prob_i(x)\,dx \tag{2.19}$$

Where $Prob_i(x) = N_{(\mu_i,\sigma_i)}(x)$=$\left(\frac{1}{\sigma} * \sqrt{2\pi}\right) * e^{-\frac{(x-\mu)^2}{2\sigma^2}}$ is the normal distribution for sample i.

$$Prob_i(x) = \left(\frac{1}{\sigma} * \sqrt{2\pi}\right) * e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{2.20}$$

Where μ is a class centroid

σ is a standard deviation of the class

$$dx \ is \ \Delta\text{x} = \frac{max-min}{n} \tag{2.21}$$

max and min are the maximum and minimum value of the variables

Hence $MASS(_{a_ib_i})$ can be re-written using Equation (2.20) and (2.21) as follows.

$$MASS(_{a_ib_i}) = \left(\int_{\min(a_i,b_i)}^{\max(a_i,b_i)} \left(\frac{1}{\sigma} * \sqrt{2\pi}\right) * e^{-\frac{(x-\mu)^2}{2\sigma^2}}\right) * \left(\frac{max-min}{n}\right) \tag{2.22}$$

Mass Distance (MD) of variable a and b is obtained by first calculating the total volume of measurement values bounded between the 2 variables and followed by taking the product over all samples, as shown in Equation (2.23).

$$MD\ (a,b) = \prod_{i=1}^{d} MASS(_{a_ib_i}) \tag{2.23}$$

where d is the number of samples.

2.4. Discussion and Summary

From the review of the literature, it can be seen that the biomedical area is data rich through the development of high throughput technologies such as microarrays, mass spectrometry and from the international genome projects. Development of new computational techniques to analyse these data is vital for progress to be made from bio-information to in the bio-knowledge and followed by drug discovery. One approach involved feature selection and techniques involved evolutionary approaches, rough set theory, various machine learning techniques and hybrids of some of these approaches. However given the characteristics associated with biological datasets, approaches involving traditional statistical approaches and machine learning techniques as described in Section 2.2 may not effective in their analysis.

From a biomedical perspective, groups of features are also known to work together as components in a biological pathway. However, as seen in the review, many existing data analysis approaches in bioinformatics may only involve evaluating each feature separately (univariate analysis) and do not consider possible correlations amongst features nor the joint behavior of a combination of features. There is an increasing need for development of techniques that attempts to address this limitation and where the basis of the selection involved the evaluation of different combinations of features by simultaneously considering two or more selection criteria.

In summary, this chapter has briefly described fundamental concepts associated with DM and FS methods. Existing work related to FS and classification approaches for analysing high dimensional biological data were also outlined. Lastly Section 2.3 presented a review of the various techniques associated with the proposed approaches in this study.

The next chapter describes common elements in this study. These include the seven datasets used to evaluate the approaches in this study, the CV strategy, the process of checking for the state of convergence and termination conditions for the GA.

3. Datasets, Evaluation strategy, Convergence and Termination criteria

This chapter has four main sections that described common elements employed in this study. The first section described the datasets used for evaluating techniques developed in this study, the second section outlined the CV approach, and Section 3 and 4 detailed the process of checking the state of convergence and termination conditions for the GA, respectively.

3.1. Datasets

This section describes 7 biomedical public datasets associated with various diseases, ranging from Ray *et al.* Alzheimer's Disease (AD) (Ray *et al.*, 2007), Alon *et al.* Colon cancer (Alon *et al.*, 1999), Leukemia cancer (Golub *et al.*, 1999), Lung cancer (Gordon *et al.*, 2002) Lymphoma cancer (Alizadeh *et al.*, 2000), Ovarian cancer (Petricoin *et al.*, 2002) and Prostate cancer (Singh *et al.*, 2002).

Table 3-1 showed a summary description of these seven datasets.

Table 3-1 Summary of seven public datasets used in the study

Dataset	Type of data	No of features	No of classes	No of samples	Data type
Ray *et al.* AD (Ray et al., 2007)	Protein immunoassay	120	2	259	Continuous
Alon *et al.* Colon (Alon *et al.*, 1999)	Cancer microarray	2000	2	62	
ALL-AML Leukemia (Golub *et al.*, 1999)		7129	2	72	
Lung (Gordon *et al.*, 2002)		12533	2	181	
Lymphoma (Alizadeh *et al.*, 2000)		4026	2	47	

Prostate (Singh *et al.*, 2002)		12600	2	136	
Ovarian (Petricoin *et al.*, 2002)	Proteomic spectra	15154	2	253	

These datasets have already been used in the evaluation of FS and classification techniques in previous studies associated with bioinformatics (Banerjee *et al.*, 2007; Cao, Lee, Seng, & Gu, 2003; Klassen & Kim, 2009; Ravetti & Moscato, 2008; Ray *et al.*, 2007; Rocha de Paula *et al.*, 2011; Yeung, Bumgarner, & Raftery, 2005). All datasets (except the AD dataset) are taken from Kent Ridge Bio-medical Dataset Repository (J. Li & Liu, 2002). Each dataset is a publicly available dataset. Details of data collection techniques for each dataset can be referred to the original author's paper.

One of the seven datasets (AD) in Table 3-1 is from Alzheimer's disease domain and generated using protein immunoassay technologies, and 5 datasets are associated with cancer, namely Colon, Leukemia, Lung, Lymphoma and Prostate cancer, all of which are generated using microarray technologies. Lastly, the Ovarian cancer dataset is generated using proteomic spectra technologies. The AD dataset consists of a relatively small number of attributes (120 attributes), while each of the remaining seven datasets has a large number of attributes ranging from 4026 to 15154, with the number of samples in these datasets being extremely small in comparison to the number of attributes. For example, the Prostate cancer dataset consists of only 136 samples, with each sample having 12600 attributes. This is a typical example of datasets in the biomedical domain. The samples in all these datasets are classified into 2 classes, diseased versus non-diseased. The attributes are continuous variables and the format of the data files is either in Excel or text format. The following sections describe each of these datasets.

3.1.1. Ray *et al.* Alzheimer's Disease (AD) datasets

The assay dataset used in Ray *et al.*'s experiment (2007) consists of 259 plasma samples from 6 categories, namely Alzheimer disease (AD), Non-demented control (NDC), Other Dementia (OD), Mild Cognitive Impairment (MCI), Other Neurological Disease

(OND) and Rheumatoid Arthritis (RA). Each sample is characterized by measurements associated with 120 known signalling proteins (attributes), saved in a Microsoft Excel file format. Table 3-2 showed the breakdown of information for this dataset. This study used the same training set and test sets as determined in the Ray *et al.*'s study (2007).

Table 3-2 Description of subsets associated with the Ray *et al.*'s dataset (2007)

Dataset: 259 samples 120 attributes	Type of data & Number of samples
Alzheimer disease (AD) (85) Non-demented control (NDC) (79) Other dementia (OD) (11) Mild cognitive Impairment (MCI) (47)	Training set (83) AD: 43 NDC: 40
	AD test set (92) AD: 42 NDC: 39 OD: 11
	MCI test set (47) MCI -> AD: 22 MCI -> OD: 8 MCI -> MCI: 17
Other neurological disease (OND) (21) Rheumatoid arthritis (RA) (16)	Not used for AD classification

- Of the 259 samples, 85 samples belong to the AD group and 79 samples belong to the NDC group. Samples from these two groups are allocated into 2 sets: training and test set. The training set consists of 43 samples belonging to AD group and 40 samples from the NDC group.
- There are two additional test sets used in this study: the AD test set consists of 42 AD, 39 NDC and 11 OD making a total of 92 samples and the MCI test set consists of 47 cases of MCI. In the case of MCI, after 2-6 years of follow-up diagnosis, 22 cases developed to AD, 8 cases developed to OD and 17 cases still remained as MCI, i.e., not developed to AD or OD (Ray *et al.*, 2007).

- According to Ray *et al*. (2007), an additional set, consisting of 21 OND and 16 RA from the 259 samples, was not used for classification.

For more information regarding the methods used to produce the data and the description of the 120 proteins in the dataset, please refer to Ray *et al*. (2007)

3.1.2. Alon *et al.* Colon cancer data

The Colon cancer dataset consists of 62 samples that was analysed using Affymetrix oligonucleotide arrays. Samples were taken from tumours and normal tissues. Each sample has 2000 attributes with continuous values obtained from the microarray analysis. Data is saved in text file format. Table 3-3 shows the detailed breakdown of the dataset.

Table 3-3 Description of subsets of Colon data

Dataset: 62 samples 2000 attributes	Type of data & Number of samples
Tumour colon cancer (T) (40) Normal tissues (N) (22)	Training set (46) T: 30 N: 16
	Test set (16) T: 10 N: 6

As seen in Table 3-3, two groups consisting of 40 tumour (T) tissue samples and 22 normal (N) tissue samples are distributed into a training set consisting of 46 (30 T and 16N) samples, and a test set consisting of 16 (10T and 6N) samples. This distribution of samples in this dataset into the training and test sets followed the same configuration as used in the study conducted by Klassen and Kim (2009).

3.1.3. Leukemia cancer data

Leukemia dataset contains 72 bone marrow samples from acute leukemia patients. It includes samples from Acute Lymphoblastic Leukemia (ALL) and Acute Myelogenous Leukemia (AML), with each sample having 7129 attributes with continuous values. The data is stored in text format. A summary of this dataset is shown in Table 3-4. This study used the same training set and test sets as determined in the study conducted by Golub, *et al*. (1999), J. Li & Liu (2002) and Klassen and Kim (2009).

Table 3-4 Description of subsets of Leukemia data

Dataset: 72 samples 7129 attributes	Type of data & Number of samples
Acute Lymphoblastic Leukemia (ALL) (47) Acute Myelogenous Leukemia (AML) (25)	Training set (38) ALL: 27 AML: 11
	Test set (34) ALL: 20 AML: 14

As seen in Table 3-4, the groups of 47 ALL samples and 25 AML samples from a total 72 samples are allocated into a training set consisting of 38 (27 ALL and 11 AML) samples and a test set consisting of 34 (20 ALL and 14 AML) samples.

3.1.4. Lung cancer data

The Lung cancer dataset contains 181 samples from adenocarcinoma (ADCA) and malignant pleural mesothelioma (MPM) patients. Each sample has 12533 attributes that are continuous values. Dataset is stored in text format. A summary of this dataset is shown in Table 3-5. This study used the same training set and test sets as determined in J. Li and Liu's study (2002)

Table 3-5 Description of subsets of Lung data

Dataset: 181 samples 12533 attributes	Type of data & Number of samples
Adenocarcinoma (ADCA) (150) Malignant Pleural Mesothelioma (MPM) (31)	Training set (149) ADCA: 134 MPM: 15
	Test set (32) ADCA: 16 MPM: 16

As seen in Table 3-5, the groups of 150 ADCA samples and 31 MPM samples from the total 181 samples are distributed into a training set consisting of 149 (134 ADCA and 15 MPM) samples and a test set consisting of 32 (16 ADCA and 16 MPM) samples.

3.1.5. Lymphoma cancer data

Lymphoma cancer dataset contains 47 samples of Diffuse large B-cell lymphoma (DLBCL), including Activated B-like DLBCL (ACL) and Germinal Centre B-like DLBCL (GCL). Each sample has 4026 attributes of continuous values. Dataset is stored in text format. A summary of this dataset is shown in Table 3-6.

Table 3-6 Description of subsets of Lymphoma data

Dataset: 47 samples 4026 attributes	Type of data & Number of samples
Germinal Centre B-like (GCL) (24) Activated B-like (ACL) (23)	Training set (34) GCL: 17 ACL: 17
	Test set (13) GCL: 7 ACL: 6

As seen in Table 3-6, the distribution of samples in this dataset followed the same configuration as used in the study conducted by L. Li, Weinberg, Darden and Pedersen (2001) whereby the groups of 24 GCL samples and 23 ACL samples from a total 47 samples are allocated in the following manner: a training set consisting of 24 (17 GCL and 17 ACL) samples, and a test set consisting of 23 (7 GCL and 6 ACL) samples.

3.1.6. Prostate cancer data

Prostate cancer dataset contains 136 samples from tumour and normal tissues. These samples were analysed using Affymetrix oligonucleotide microarrays resulting in each samples having 12600 attributes with continuous values. Dataset is stored in text format. A summary of this dataset is shown in Table 3-7. This study used the same training set and test sets as determined in J. Li and Liu's study (2002)

Table 3-7 Description of subsets of Prostate data

Dataset: 136 samples 12600 attributes	Type of dataset
Tumour tissues (T) (77) Normal tissues (N) (59)	Training set (102) T: 52 N: 50
	Test set (34) T: 25 N: 9

As seen in Table 3-7, the groups of 77 tumour (T) samples and 59 normal (N) samples from the total 136 samples are allocated in the following manner: a training set consists of 102 (52 T and 50 N) samples, a test set consists of 34 (25 T and 9 N) samples

3.1.7. Ovarian cancer data

The Ovarian cancer dataset contains 253 samples of cancer and normal tissues. Each sample has 15154 attributes (continuous values) was analysed using mass spectroscopy. Dataset is stored in text format. A summary of this dataset is shown in Table 3-8.

Table 3-8 Description of subsets of Ovarian data

Dataset: 253 samples 15154 attributes	Type of dataset
Cancer disease (D) (162) Normal control (N) (91)	Training set (126) N: 45 D: 81
	Test set (127) N: 46 D: 81

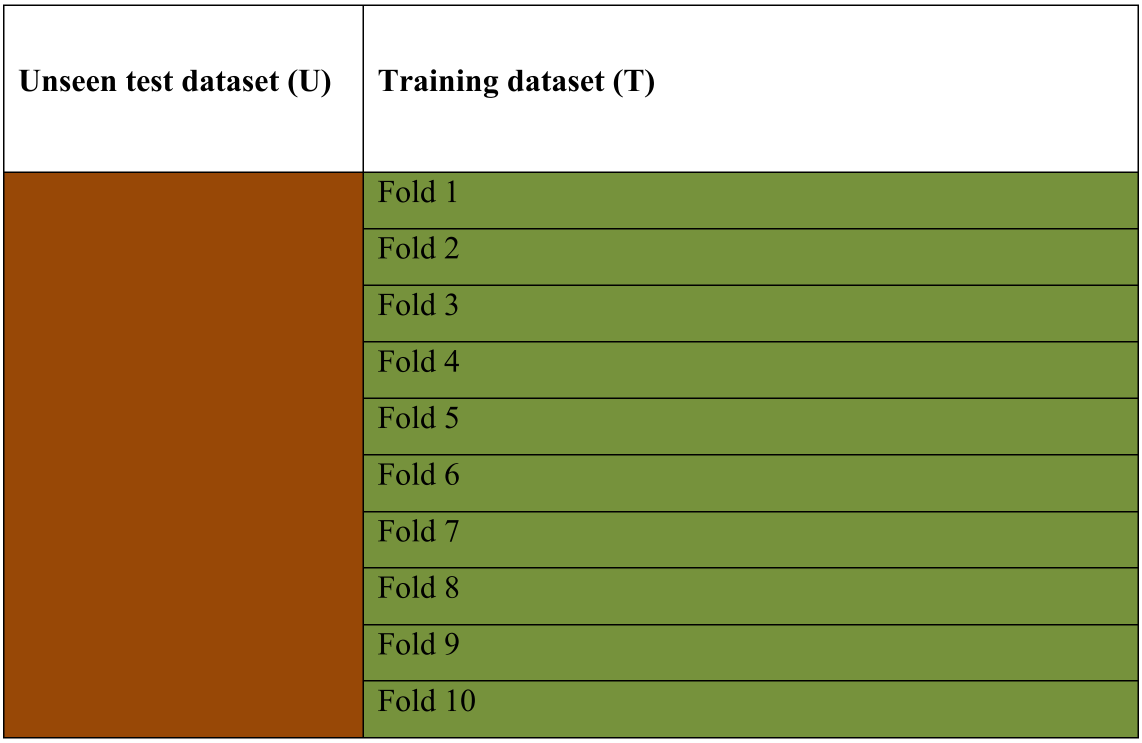
As seen in Table 3-8, the distribution of samples in this dataset followed the same configuration as used in the study conducted by J. Li and Liu (2002) whereby the groups of 91 normal control (N) samples and 162 cancer samples (D) from the total 253 samples are allocated in the following manner: a training set consisting of 126 (45 N and 81 D) samples and a test set consisting of 127 (46 N and 81 D) samples.

This section described the various datasets used to evaluate the techniques developed in this study and the next section will describe the evaluation method for assessing the performance of the developed techniques in this study.

3.2. Evaluation Strategy

Predictive DM is one branch of DM where a model formulated using some existing data is used to predict future behaviour/outcomes. There are a number of ways to measure the performance of these models, namely: classification accuracy, error rates, lift charts (charts are used to measure the performance of the prediction model by plotting the number of true positive predictions against the total number of samples) and ROC curves (charts are used to measure the performance of the prediction model by plotting the number of true positive predictions against the total number of negative predictions) (Witten & Frank, 2005). However, an issue associated with the use of these models for prediction is that while they perform effectively on classifying training data, they may perform badly on future unseen data. Evaluation of these models then becomes important in terms of the reliability of the predicted results, with CV being the most widely used approach for evaluating these models (i.e., the generalization ability of the models) (Witten & Frank, 2005). Other methods which also have been used for model evaluation include holdout and bootstrap 0.632 (Wood, Visscher, & Mengersen, 2007). Cross validation (CV) is a statistical approach that consists of iterations where subsets of the data (training data) are first used to fit a model and followed by the testing of the performance of that model using the rest of the data (validation data). However, if the approach is not carried out properly, selection bias can occur and the resulting classification results can be optimistically biased (Ambroise & McLachlan, 2002). Typically in mining mass throughput data such microarray data, the first step involves employing FS techniques to reduce the number of attributes to a small number. Selection bias occurs if the whole dataset is first used in the FS process and then followed by the CV process. This is due to the fact that the selection of these features already incorporated information on the test set. Thus in order to avoid the selection bias in the process of selecting the training model, "*the test set must play no role in the feature-selection process for an unbiased estimate to be obtained*" (Ambroise & McLachlan, 2002, p. 6566).

Other issues that must be considered as part of the evaluation strategy are stratification, and the number of folds in the CV process, number of repetitions of a CV process and computation resources and lastly, simulation of prediction of new data. Stratification is a process for ensuring that each class associated with the dataset is properly represented, with samples of each class being in the right proportion in both the training and test sets. According to Witten and Frank (2005), a 10-fold is sufficient to obtain the best error estimate. If stratification is incorporated into a 10-fold CV procedure then the evaluation approach is known as a *stratified 10-fold CV*. In addition, consideration must also be given to the number of repetitions of the CV process as a single 10-fold stratified CV will not be able to guarantee a reliable error estimate. Repetitions of CV need considerations of computation resources and lastly, simulation of prediction of new data implies having an untouched validation dataset as this is the only way to simulate prediction of new data. The following section outlines the evaluation strategy, that addressed the issues discussed above and, is used in this study.



Unseen test dataset (U)	Training dataset (T)
	Fold 1
	Fold 2
	Fold 3
	Fold 4
	Fold 5
	Fold 6
	Fold 7
	Fold 8
	Fold 9
	Fold 10

Figure 3-1 General mining structure: the breakdown of a dataset into an unseen test dataset (U) (brown colour) and a training dataset (T) (green colour), in which (T) is further split into 10 folds for 10 fold CV.

The evaluating strategy consists of 3 major steps: 1) partitioning the dataset into a training dataset (T) and an unseen test dataset (U), 2) performing evaluation of 10 fold CV on the training data (T) and 3) performing test classification on the unseen test data (U). The following section describes these steps in details.

Step 1. Partition the full dataset into a stratified training dataset and a stratified unseen test dataset.

- Randomly assign each sample from the dataset into one of two groups: training (T) and unseen test dataset (U). As part of this allocation process, ensure that each class associated with the dataset (e.g. disease and healthy control) is appropriately represented in both the training and unseen test dataset, thus incorporating stratification. In this study, the split ratio (into training and unseen test datasets) and the proportion of samples for each of the classes, e.g., Cancer and Normal in Ovarian cancer dataset, into the training set and unseen test set respectively, from each of the full dataset, followed the configurations used by either the original authors or authors who have also used the same datasets in their subsequent studies.

Step 2. Perform 10 fold CV using the training dataset (T).

- Randomly assign samples from the training dataset (T) into 10 folds to obtain 10 stratified subsets.
- Select 1 subset (fold) as the validation set and use the remaining 9 subsets as a training set. For example, in Figure 3-1, fold 1 may be set to be the validation set and the training set then consists of fold 2 to fold 10.
- Using a selected approach developed in this study and 9 folds, generate the classification model and evaluate its performance using the validation set. This process is repeated 10 times (i.e. 10 folds), with each subset in turn being the validation set and the remaining subsets (9 folds) being a training set. Calculate the performance of 10 fold CV by averaging the classification error rate over the 10 folds.

Step 3. Perform classification on unseen test dataset (U).

- Using the selected approach used in Step 2 and the training data (T), generate the classification model and evaluate its performance on the unseen test dataset (U) to obtain the unseen test classification accuracy. This stage may also be seen as simulating the prediction of new data as the unseen test dataset has been kept totally separate from the training dataset.

The entire classification process (Steps 2 and 3) is repeated 15 times (15 independent runs) which means including 15 times of 10 fold CV. The final training classification accuracy of 15 times of 10 fold CV and classification accuracy of the unseen test dataset are calculated by averaging their respective accuracy rates over the the15 independent runs. Running multiple times is also essential for evaluating the quality and performance of evolutionary algorithm such as GA (Alba, Garcia-Nieto, Jourdan, & Talbi, 2007). For example, 5 independent runs were used in Huerta, Duvalm and Hao's experiment (2006), 10 runs were used in studies (Alba *et al.*, 2007; Bala, Huang, Vafaie, DeJong, & Wechsler, 1995; Kenneth A DeJong & Spears, 1990; Sharpe & Glover, 1999); and 20 runs (Stein, Chen, Wu, & Hua, 2005; Zhang & Sun, 2002). Thus in this study, 15 runs are considered as a sufficient number for estimating the classification accuracy and the performances of the proposed approaches, bearing in mind the tradeoffs – the need for sufficient number of runs for evaluating the performance of proposed techniques and the computational overhead associated with the analysis of high dimensional biological datasets.

3.3. Termination criteria

According to many researchers (Safe, Carballido, Ponzoni and Brignole (2004), Koumousis and Katsaras (2006), Milton (2009) and Ong and Fukushima (2011)), the most common criteria used to terminate GA are: full population convergence to a single solution, fitness of the population has not improved over a pre-define number of consecutive generations, a pre-defined maximum number of generations (or fitness evaluations) have been executed, or the best fitness values found over a number of generations. The following section describes the stopping criteria employed in this study for terminating GA. An additional check using the strategy of detecting the convergence status as described in the previous section 3.3 is also conducted after the GA is deemed to have converged.

The termination criteria employed here for terminating the GA consists of a combination of two conditions: executing for a predefined maximum number of generations and that the fitness of the population did not change over a pre-defined number of consecutive generations. In order to implement this termination approach, for each of generations in the GA, the following calculations are carried out, 1) fitness of each chromosome/individual in the entire population, 2) the maximum fitness of the population, and 3) the average fitness of the population. Whilst a pre-defined maximum number of generations are not reached, the average fitness of the population is checked for any changes (improvement), if it does not change (improve) over a pre-defined number of consecutive generations (100 generations), then the population is considered as having converged and the GA terminates . The choice of 100 consecutive generations for termination is based on the results of parameter tuning that has been completed for all the datasets in this study, as described in Section 5.3. The number of 100 consecutive generations is large enough for avoiding a pre-mature convergence. This termination approach not only avoids premature convergence but also reduce the total computational time because there is no need to keep executing the algorithm when the population in question has already converged (Kumar & Rockett, 2002; Ong & Fukushima, 2011).

The following algorithm describes the procedure of terminating GA.

```
Input
    • Individual fitness in population ($F_{ind}$)
    • Population size ($s$)
    • A pre-defined maximum number of generations ($G_{max}$)
    • A pre-defined maximum number of consecutive generations of
      convergence
      ($C_{max}$)

Output
    • Maximum population fitness ($F_{max}$)
    • Average population fitness ($F_{avg}$)

Steps
  1. Set counter = 0
  2. For 1 to $G_{max}$
       a. Calculate total population fitness ($F_{total}$) = $\sum_1^s F_{ind}$
       b. Calculate $F_{avg} = F_{total} / s$
       c. If $F_{avg}$ does not change then
            • Increase counter by 1
            • If counter = $C_{max}$ then
                  Terminate GA
       d. Else
            • Reset counter back to 0
```

Figure 3-2 Termination procedure for GA

To minimize the likelihood of premature convergence, this study has incorporated the following:

- Selection of parameter settings from one of the four sets of “standard parameter values” for GA from the literature and these have been described in Section 5.3.1. This study has also used tournament selection, elitism and an appropriate crossover probability to ensure a balance between diversity and selection pressure, so as to avoid premature convergence. Details are shown in Table 4-1 and Table 5-2.

- In addition, this study instituted a mechanism consisting of three parts for checking for the occurrence of premature convergence: an approach proposed by Srinivas and Patnaik (1994) for checking for premature convergence and checking that the maximum fitness, at the point of convergence, approaches the theoretical maximum fitness value associated with the different fitness functions and the termination criteria described in Section 3.3 and 3.4 of the thesis respectively. In this check, the function defined by Srinivas and Patnaik approaches zero and the maximum fitness, at the point of convergence, approaches the theoretical maximum fitness value associated with the different fitness functions and termination condition (that is, the fitness of the population did not change over a pre-defined number of consecutive generations).

3.4. Genetic Algorithm and state of convergence

The GA is incorporated in a number of approaches developed in this study. It is important to ensure that the GA has achieved convergence as a premature convergence will result in a local optimal solution instead of a global optimum. To check the algorithm is converging to the global optimum, this study used an approach proposed by Srinivas and Patnaik (1994) to check for premature convergence. This can be done by checking the difference between the average fitness (f_{avg}) and maximum fitness value (f_{max}) of the population after the GA has converged. A plot of the values for $f_{max} - f_{avg}$ is used to detect the state of convergence of GA. That is, the smaller the difference between f_{max} and f_{avg}, the better the global convergence and a better optimal solution obtained from the algorithm, thus avoiding premature convergences (M. Srinivas & Patnaik, 1994).

Figure 3-3 shows an example of the convergence plot for one GA execution over 300 generations. The blue line shows the plot of the maximum fitness of the population for each generation and the red line shows the plot of the values of ($f_{max} - f_{avg}$). The vertical axis on the left-hand side indicates maximum fitness of each generation and the vertical axis on the right hand side indicates the values for ($f_{max}- f_{avg}$). Note that ($f_{max} - f_{avg}$)

approaches to values approximately close to zero around 211 generations. This coincides with the max fitness having a value of 0.954 (1 is maximum). Figure 3-2 also shows a local optima that occurred in the execution of the algorithm over 300 generations. The 2 vertical green lines in Figure 3-2 illustrate the local optimum (a1) found prior to the algorithm reaching global convergence (the vertical blue line). If the algorithm stopped when this local optimum found, then it was a premature convergence, where ($f_{max} - f_{avg}$) approaches to values not so close to zero (a2) and it coincides with the maximum fitness having a smaller value of 0.938 (a1), compared to 0.954 (b1) and ($f_{max} - f_{avg}$) is close to zero (b2) in the case of the global convergence.

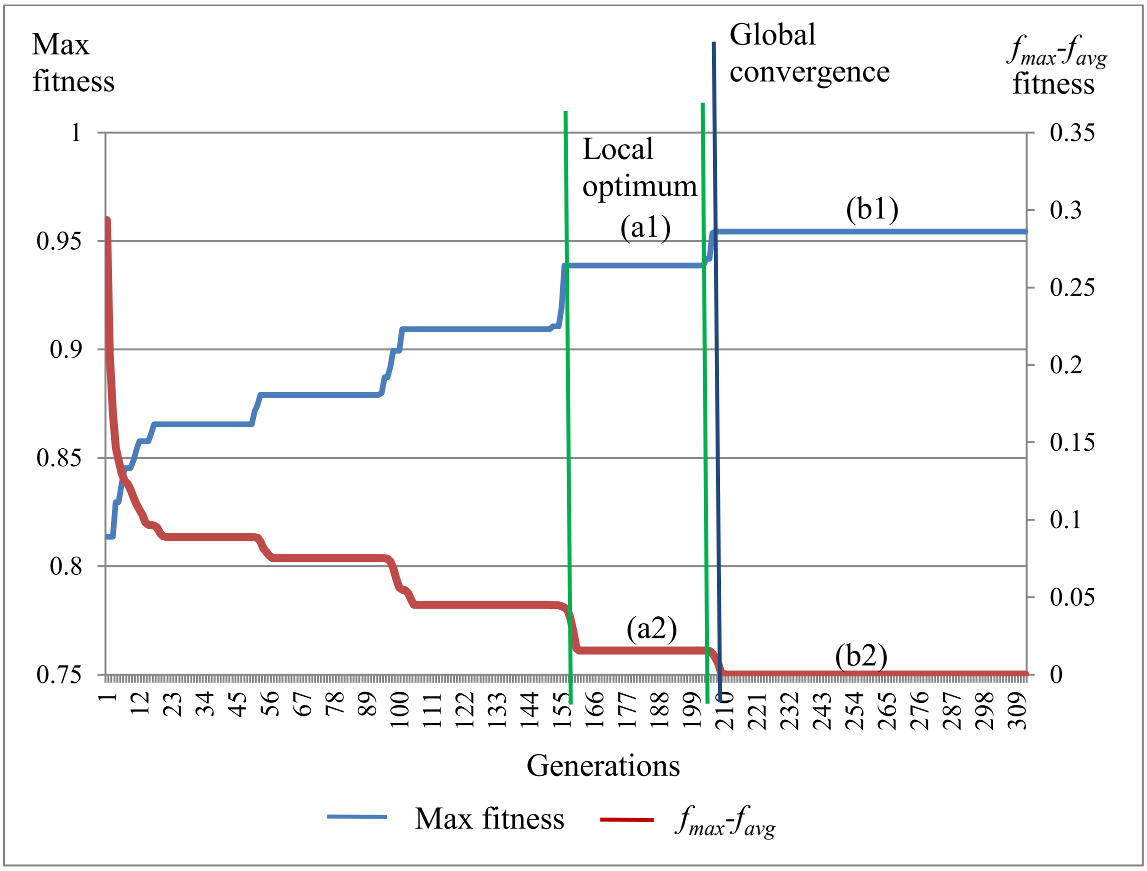


Figure 3-3 An example of a convergence status plot

3.5. Summary

This chapter has described the details of common elements associated with this study. These include: 1) the dataset of AD, Colon cancer, Leukemia cancer, Lung cancer, Lymphoma cancer, Ovarian cancer and Prostate cancer dataset in terms of the training and unseen test sets, 2) the evaluation strategy used to evaluate the proposed approaches, 3) check the state of convergence for GA, and 4) GA termination criteria. These common elements are applied to the evaluation of proposed approaches developed in this study.

In Chapter 4, the proposed approach of incorporating RST into GA for searching optimal feature sets is described. Chapter 4 is the pilot study in this thesis and involved modifying Banerjee *et al.*'s (2007) approach for generating the distinction table.

4. Rough set theory and GA approach (RST-GA)

Chapter 3 described the seven datasets used in this study. It can be seen that these datasets belongs to the category of “binary classification” problems, specifically, normal versus diseased or two variants of diseased samples. Common characteristics of these datasets are very high dimensionality and small number of samples. The challenge when classifying this type of data arises from the limited availability of a small number of samples in comparison to the large number of features associated with each sample. With a large number of features, of which, some maybe redundant or irrelevant, the classification process can be computationally intensive. Furthermore, with a small number of samples, over-fitting in training is likely to occur and can lead to higher classification errors when the trained classification model is used to classify unseen test data (data not used as part of the training).

Chapter 4 is the pilot study in this thesis and involved modifying Banerjee *et al.*’s (2007) approach for generating the distinction table. This chapter is an extended version of the paper “Incorporating genetic algorithm into rough FS for high dimensional biomedical data” (Dang, Lam, & Lee, 2011). It describes the first investigation that was carried out in this study to explore EA-based approaches for FS and classification of such high dimensional biological data. In Section 4.1, a hybrid approach, incorporating GA and RST, for searching for the best subset of optimal features is described. A description of a parameter tuning process for the GA is then outlined in Section 4.3. Using optimal sets of features generated from the proposed approach, classification was carried out using k Nearest Neighbour (k-NN) classifiers to evaluate their performance in classifying unseen test data. Classification results involving classifiers from WEKA (Waikato Environment for Knowledge Analysis) (Hall *et al.*, 2009) are also shown in Section 4.4, and followed by a discussion in Section 4.5.

Please note that in this thesis the term “feature” and “attribute” are used interchangeably and represent the same thing.

4.1. The proposed approach, RST-GA

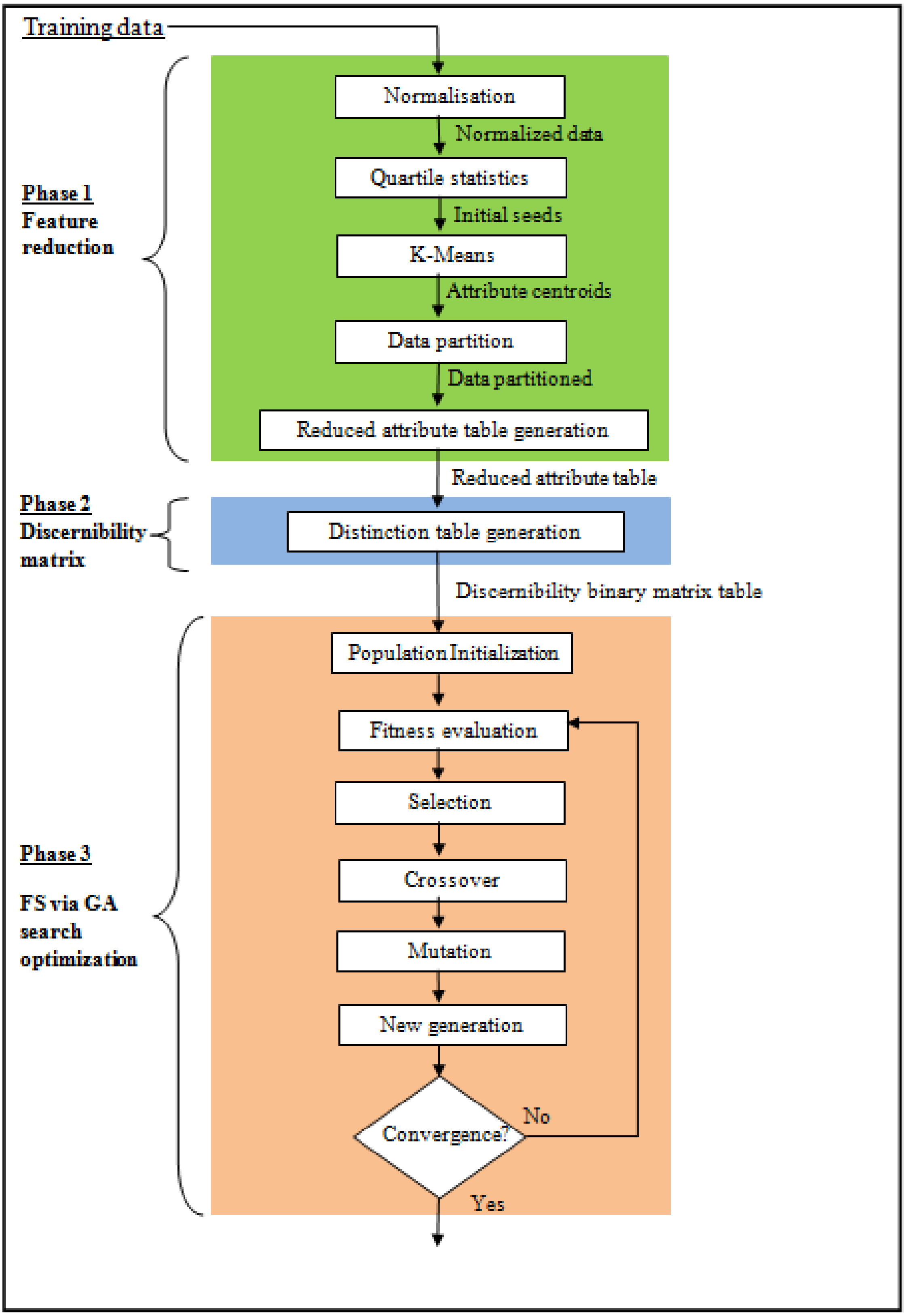


Figure 4-1 Framework of the proposed approach, RST-GA

Figure 4-1 illustrates the framework of the proposed approach, RST-GA, incorporating k-means clustering, RST and GA. As shown in the figure, the proposed approach uses a 3-phased process, consisting of:

Phase 1: This phase carries out the feature reduction step. Owing to k-means being employed to find threshold values associated with each feature, a normalization step was first carried out on all the features. The normalized values are then used to partition each corresponding feature in the process of generating a reduced attribute table. The objective of this step is to do an initial cull, completing a preliminary coarse reduction in redundancy amongst the features.

Phase 2: In this phase, a *distinction table* (Wroblewski, 1995), which is a variant of the discernibility matrix, is constructed using the reduced feature table generated from Phase 1. The *distinction table* is in the form of binary matrix.

Phase 3: GA was employed in the third phase as an optimization method to search for the optimal set of features based on the *distinction table* that has been generated from Phase 2.

RST-GA, is an initial attempt to explore approaches for analyzing high dimensional biological data and is based on Banerjee et al.'s approach, using RST and incorporating GA as a search algorithm However, the proposed approach makes improvements by using quartile statistics and K-means clustering to obtain optimal centroids for partitioning data in the first phase.

The steps associated with each of these three phases are described in the following sections.

4.1.1. Phase 1: Feature reduction

K-means is employed in this phase to generate centroids of each attribute which are subsequently used for its partitioning step. K-means is one of the most popular clustering technique and widely used in the DM community to cluster high dimensional data (Yedla, Pathakota, & Srinivasa, 2010). K-means clustering groups data into separate clusters based on the Euclidean distance between the data points and the centroids (Nazeer & Sebastian, 2009). According to Nazeer and Sebastian (2009), there are 2 steps associated with k-means. The first step is to determine the value of *k* (i.e. the number of clusters) and to initialize each of these cluster centres to a random number. The second step involves the use of a similarity measure (e.g. Euclidean distance measure) to calculate the respective distances of each data point to these centroids and assignment of the data points to the closest centroid. The new centroid for each cluster is re-calculated and the respective distances of each data point to the updated centroids are also re-calculated, and subsequently, data points are re-assigned to the clusters based on the new values of the re-calculated distances of the data points to each of the updated centroids. The process of updating cluster centroids, re-calculating the distance between data points and centroids, and re- assigning data points to the clusters continue until the convergence of clusters takes place, i.e., when there is no more changes to the cluster centroids.

In addition, Visalakshi and Thangavel (2009, p. 168) have also stated that "t*he clustering results can be greatly affected by differences in scale among the dimension from, which the distances are computed"*. Thus to address this issue, a normalization process needs to be carried out to transform raw data consisting of attribute values to a specific range such as [0, 1] prior to employing k-means clustering. In this proposed approach, min-max normalization is first used to normalize the data.

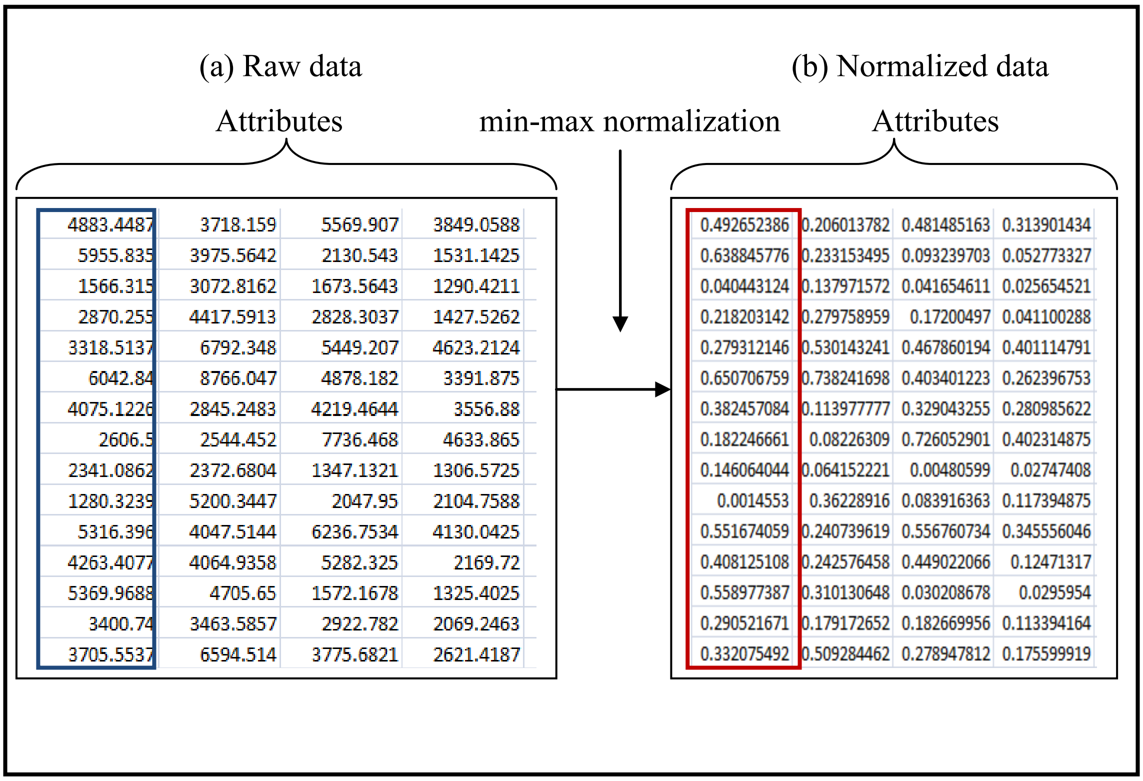
a) Normalization

The min-max normalization method (Han & Kamber, 2006) is applied to the training and test datasets, converting attribute values into the range of [0, 1] using Equation (4.1).

$$a'_j = (a_j\,(x_i) - \min_j) / \max_j - \min_j \tag{4.1}$$

where $\max_j$ and $\min_j$ are respectively, the maximum and minimum expression value of attribute a_j from all samples.

The following figure illustrates an example of normalization.


(a) Raw data
Attributes
min-max normalization
(b) Normalized data
Attributes

4883.4487	3718.159	5569.907	3849.0588
5955.835	3975.5642	2130.543	1531.1425
1566.315	3072.8162	1673.5643	1290.4211
2870.255	4417.5913	2828.3037	1427.5262
3318.5137	6792.348	5449.207	4623.2124
6042.84	8766.047	4878.182	3391.875
4075.1226	2845.2483	4219.4644	3556.88
2606.5	2544.452	7736.468	4633.865
2341.0862	2372.6804	1347.1321	1306.5725
1280.3239	5200.3447	2047.95	2104.7588
5316.396	4047.5144	6236.7534	4130.0425
4263.4077	4064.9358	5282.325	2169.72
5369.9688	4705.65	1572.1678	1325.4025
3400.74	3463.5857	2922.782	2069.2463
3705.5537	6594.514	3775.6821	2621.4187

0.492652386	0.206013782	0.481485163	0.313901434
0.638845776	0.233153495	0.093239703	0.052773327
0.040443124	0.137971572	0.041654611	0.025654521
0.218203142	0.279758959	0.17200497	0.041100288
0.279312146	0.530143241	0.467860194	0.401114791
0.650706759	0.738241698	0.403401223	0.262396753
0.382457084	0.113977777	0.329043255	0.280985622
0.182246661	0.08226309	0.726052901	0.402314875
0.146064044	0.064152221	0.00480599	0.02747408
0.0014553	0.36228916	0.083916363	0.117394875
0.551674059	0.240739619	0.556760734	0.345556046
0.408125108	0.242576458	0.449022066	0.12471317
0.558977387	0.310130648	0.030208678	0.0295954
0.290521671	0.179172652	0.182669956	0.113394164
0.332075492	0.509284462	0.278947812	0.175599919

Figure 4-2 Example of part of normalized data using min-max normalization

As seen in Figure 4-2, each attribute (column) shown in the table of Raw data (Figure 4-2 (a)) consists of values with differences in a wide range (blue box), which are normalized to values between 0 and 1 (red box) shown in the table of Normalized data (Figure 4-2 (b)). Thus, after normalization, values of each attribute are standardized in the same scale (i.e. between 0 and 1).

K-means is also very sensitive to the starting points (i.e. initial centroids) and these subsequently impact greatly on its ability to achieve global versus local optimum in terms of accuracy (i.e. clustering results) and efficiency (i.e. computational time spent to perform clustering) (Bradley & Fayyad, 1998; Nazeer & Sebastian, 2009; Yedla *et al.*, 2010). In this study, a quartile statistics technique is first employed to find more appropriate initial starting centroid values for k-means rather than using random values for initial centroids. The following section describes the quartile statistics procedure for calculating the initial starting centroids for each attribute.

b) Quartile statistics

As seen in Chapter 3, the datasets used in this study are from the bioinformatics domain and belongs to the category of "binary classification" problems, specifically, normal versus diseased or two variants of diseased samples. Values associated with attributes of such datasets typically falls into a number of categories: a normal range where it is considered to be associated with a "non-diseased condition" and a value that's too high or too low may indicate abnormality and that it is associated with a "diseased condition". For example, a measurement associated with blood glucose level that's below 70mg/dl (milligrams per decilitre) is considered to be associated with a low blood glucose condition and a measurement above 180mg/dl is considered to be associated with a high blood glucose condition (*hyperglycemia*) – a condition which is known to be associated with diabetes, while a measurement between 70mg/dl and 180mg/dl is considered to be associated with a normal blood glucose level (*Euglycemia*) (W. L. Clarke et al., 2005).

On the basis of the above characteristic, the approach employs quartile statistics to find three values associated with each attribute: 25th percentile, 50th percentile and 75th percentile; with the value at 50th percentile being used to reflect a value associated with the normal range and the remaining two to reflect values associated with conditions considered to be either too low or too high. These three values are then used as initialization values for centroids of three clusters in the third step – the application of K-means algorithm. The steps used in the calculation of quartile statistics are shown below (Banerjee *et al.*, 2007).

- Sort values associated with each attribute in ascending order.
- Partition the sorted values for each attribute equally into small class intervals (δ).
- Calculate quartile statistics for each attribute to obtain the lower threshold value (Th_l), middle threshold value (Th_m) and upper threshold value (Th_u) using the formula defined in Banerjee *et al.* (2007), and shown as follows.

$$Th_k = L_c + \frac{(R_k - cfr_{c-1})}{fr_c} * \delta \tag{4.2}$$

where L_c is the lower limit of the C_{th} class interval

R_k is the rank of the k_{th} interval value

$R_k = \frac{N*k}{p}$ with p = number of partition

N is a number of objects

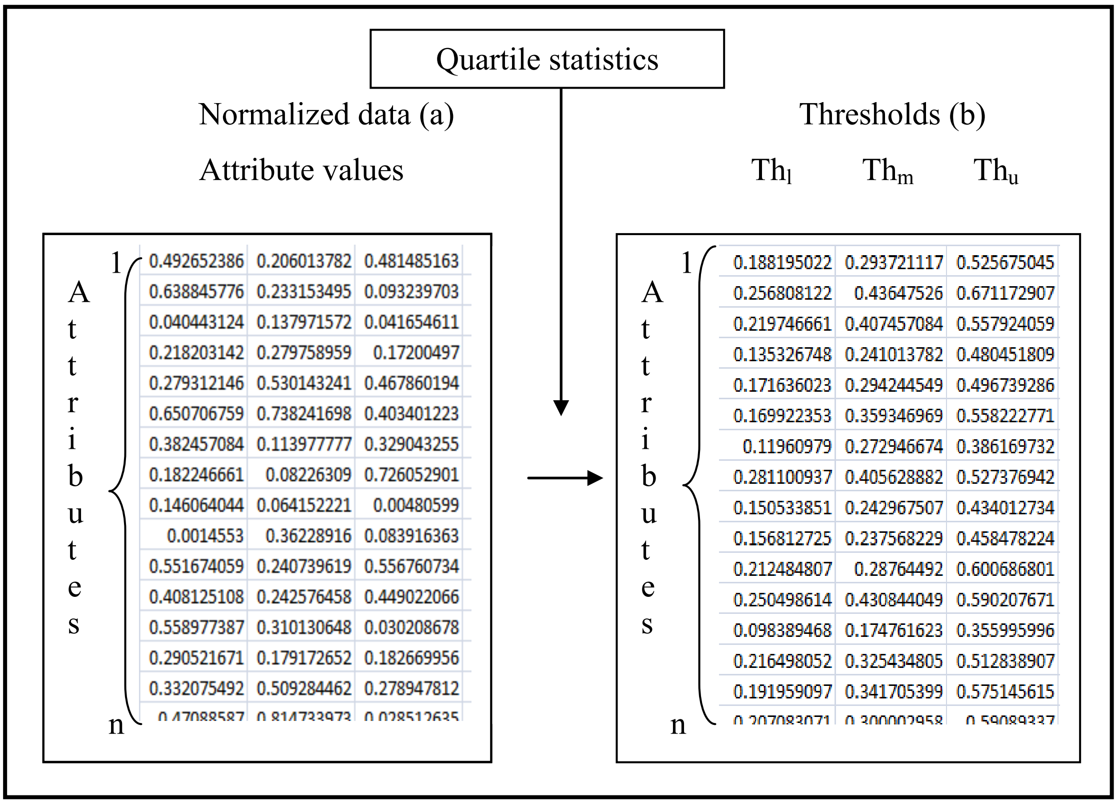
k = k^{th} partition value, k=1, 2, 3 for 4 partitions

cfr_{c-1} is the cumulative frequency of the immediately preceding class interval such that

$$cfr_{c-1} \leq R_k \leq cfr_c$$

fr_c is the class frequency

δ is the class interval width



0.492652386	0.206013782	0.481485163
0.638845776	0.233153495	0.093239703
0.040443124	0.137971572	0.041654611
0.218203142	0.279758959	0.17200497
0.279312146	0.530143241	0.467860194
0.650706759	0.738241698	0.403401223
0.382457084	0.113977777	0.329043255
0.182246661	0.08226309	0.726052901
0.146064044	0.064152221	0.00480599
0.0014553	0.36228916	0.083916363
0.551674059	0.240739619	0.556760734
0.408125108	0.242576458	0.449022066
0.558977387	0.310130648	0.030208678
0.290521671	0.179172652	0.182669956
0.332075492	0.509284462	0.278947812
0.47088587	0.814733973	0.028512635

Th_l	Th_m	Th_u
0.188195022	0.293721117	0.525675045
0.256808122	0.43647526	0.671172907
0.219746661	0.407457084	0.557924059
0.135326748	0.241013782	0.480451809
0.171636023	0.294244549	0.496739286
0.169922353	0.359346969	0.558222771
0.11960979	0.272946674	0.386169732
0.281100937	0.405628882	0.527376942
0.150533851	0.242967507	0.434012734
0.156812725	0.237568229	0.458478224
0.212484807	0.28764492	0.600686801
0.250498614	0.430844049	0.590207671
0.098389468	0.174761623	0.355995996
0.216498052	0.325434805	0.512838907
0.191959097	0.341705399	0.575145615
0.207083071	0.300002958	0.59089337

Figure 4-3 Example of partial table of Th_l, Th_m andTh_u threshold values generated as initial starting points for k-means using the quartile statistics method.

As seen in Figure 4-3, each attribute with its values shown as a row in the table of Normalized data (Figure 4-3(a)) is partitioned into 3 levels of thresholds, lower (Th_l), middle (Th_m) and upper (Th_u) shown as a row in the table of Thresholds (Figure 4-3(b)). The three threshold values associated with each attribute are used as initial centroid points for the K-means clustering step for partitioning each attribute.

c) K-means

The k-means algorithm employed here has been described in Section 2.4.1. Figure 4-4 shows the refinement of the cluster centroids before (initial) and after k-means (final) for one attribute. The initial centroid value (red square) in each cluster (C1, C2, and C3) shifts towards the centre of the cluster, i.e., red squares move to the green triangles which are closer to the centre of the clusters. Subsequently, the final centroids obtained from the k-means step are used in the RST process to produce a *distinction* table.

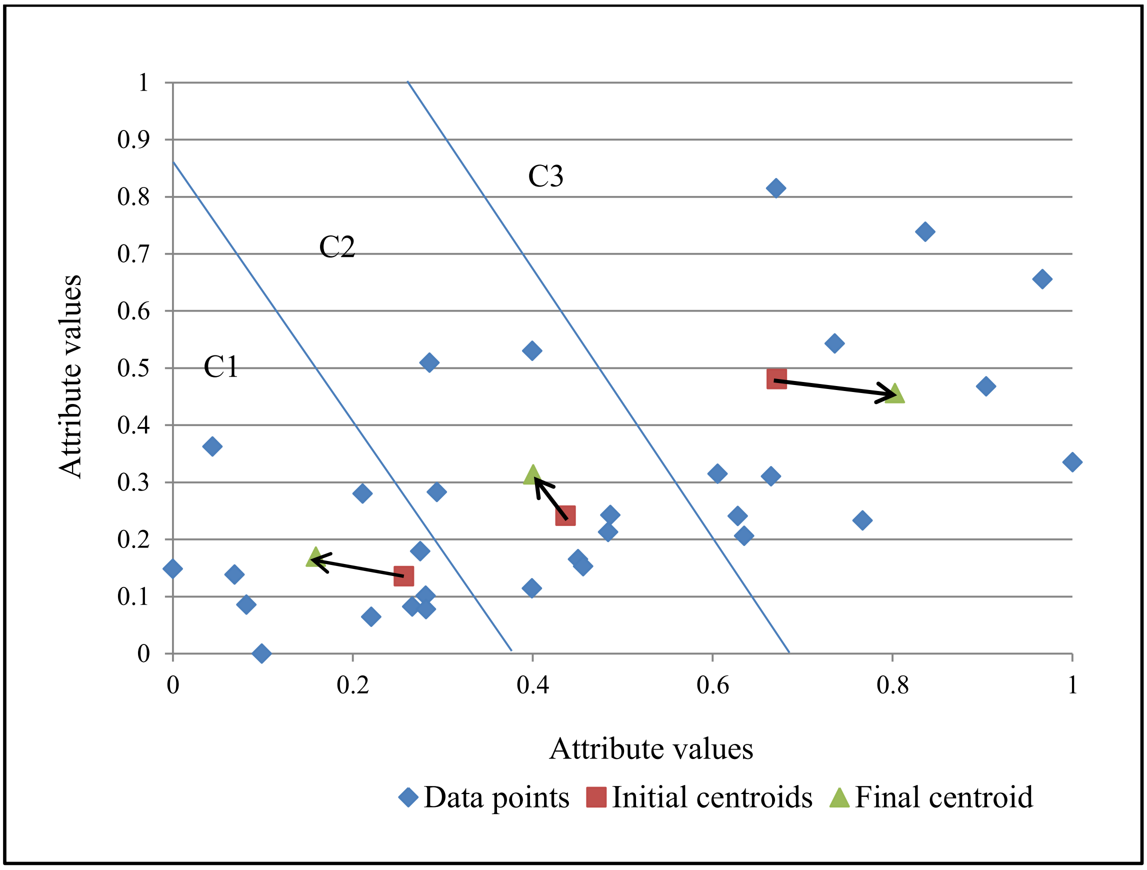


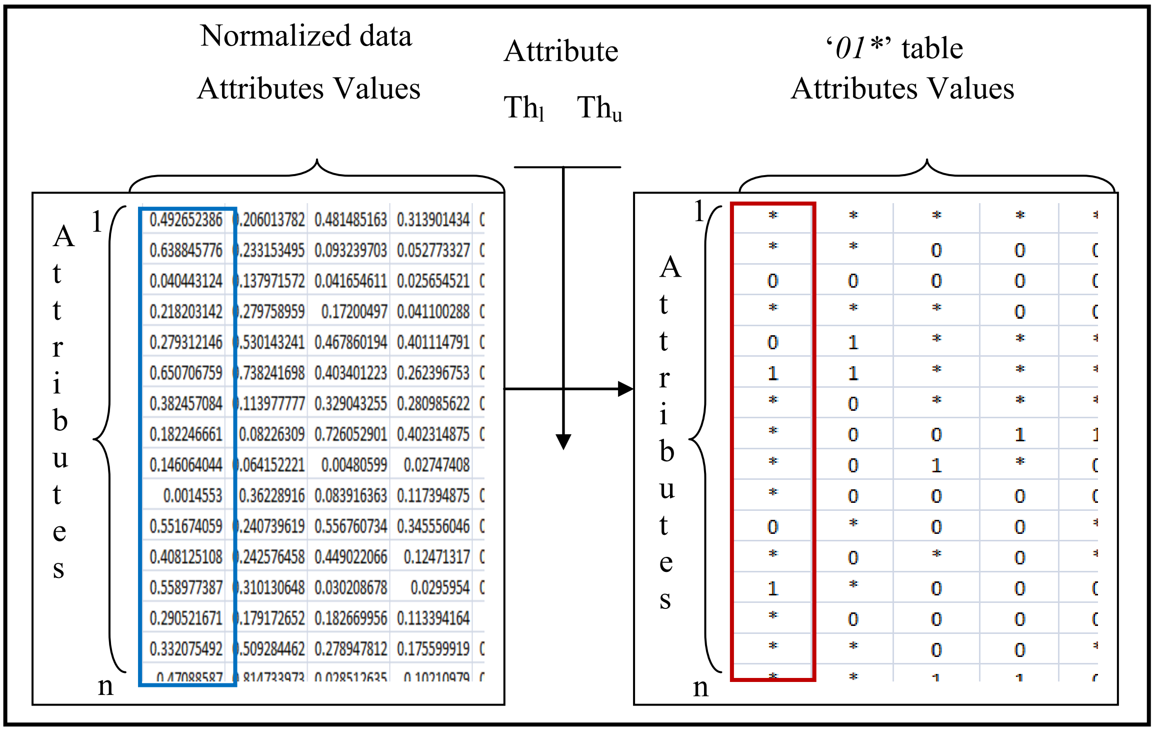
Figure 4-4 Example of cluster centroid positions before and after k-means

d) Partition data

Attributes are considered to be “of interest” if their values have a decisive role in differentiating between individuals belonging to different classes (e.g. diseased vs. non-diseased). Using the previous example, blood glucose level can be an attribute of interest if the task is to decide whether an individual is suffering from diabetes – specifically in the case where the blood glucose level is either very high or very low. This implies that this value associated with this attribute differs between the diseased and non-diseased individuals, with diseased individuals having values outside the norm. Using this rationale, the attributes in the datasets are processed in the following manner: Using the final centroid values obtained in the k-means clustering step, thresholds Th_l and Th_u are assigned as the lower and upper attribute thresholds respectively. These values are subsequently used for transforming the values of each attribute into 0, 1 and "*", with "*" is considered as a "*don't care*" condition (Banerjee *et al*., 2007, p. 625). The implication here is that the range of interest is when the attribute value is at the extreme ends. The following rules are used to process each attribute:

1. If an attribute value is less than or equal to its associated Th_l then assign the value of 0.
2. If an attribute value is greater or equal to its associated Th_u then assign the value of 1.
3. If an attribute value is greater than its associated Th_l and less than its associated Th_u assigned it to "*"

As seen in Figure 4-5, values in each attribute (blue box) are compared to its respective Th_l and Th_u threshold value and has been converted to 0 or 1 or "*" (red box) based the conditions as specified above. As a result, a table of 0, 1 and "*" values are created.



0.492652386	.206013782	0.481485163	0.313901434
0.638845776	.233153495	0.093239703	0.052773327
0.040443124	.137971572	0.041654611	0.025654521
0.218203142	.279758959	0.17200497	0.041100288
0.279312146	.530143241	0.467860194	0.401114791
0.650706759	.738241698	0.403401223	0.262396753
0.382457084	.113977777	0.329043255	0.280985622
0.182246661	0.08226309	0.726052901	0.402314875
0.146064044	.064152221	0.00480599	0.02747408
0.0014553	0.36228916	0.083916363	0.117394875
0.551674059	.240739619	0.556760734	0.345556046
0.408125108	.242576458	0.449022066	0.12471317
0.558977387	.310130648	0.030208678	0.0295954
0.290521671	.179172652	0.182669956	0.113394164
0.332075492	.509284462	0.278947812	0.175599919
0.47088587	.814733973	0.028512635	0.10210979

*	*	*	*
*	*	0	0
0	0	0	0
*	*	*	0
0	1	*	*
1	1	*	*
*	0	*	*
*	0	0	1
*	0	1	*
*	0	0	0
0	*	0	0
*	0	*	0
1	*	0	0
*	0	0	0
*	*	0	0
*	*	1	1

Figure 4-5 Example of a "*01**" table

e) Generate reduced attribute value table

Based on the "*" values in the *"01*"* table generated from the previous step, the average frequency of "*" is computed from the whole dataset (table) and then used as a "*" guided threshold value (Th*$_{avg}$) to eliminate attributes. As mentioned earlier, attributes with a majority of "*" are considered as not being significant in separating the different classes. Therefore, attributes with a total number of "*" greater than or equal to Th*$_{avg}$ are eliminated from the attribute list. As a result, a large number of attributes are eliminated and a reduced attribute value table (A_r) is produced.

4.1.2. Phase 2: Generate a distinction table

In this phase, the reduced attribute table, A_r, is then used to create a *distinction* table. The *distinction* table is a variant of the discernibility matrix which is based on the indiscernibility relation approach. Objects are divided into equivalence classes based on equivalence relations such that two objects are in the same class (equivalence class) if and only if they have the same attribute values (equivalence relation). A discernibility matrix (D_m) is defined as a matrix of *m* rows by *n* columns of an information system (S) of *N* samples and *A* attributes. A discernibility matrix of an information system, D_m (S), with the ith, jth entry (E_{ij}) is defined as $D_m(S)_{E_{ij}} = \{a \in A: a(x_i) \neq (x_j)\}, i, j = 1 \dots m$ (Hoa & Son, 1996). According to Banerjee, *et al.* (2007), a *distinction* table created based on the following criteria is greatly reduced in dimension and the computational time involved is shorter. This distinction table is called a "*d-Distinction*" table, the same name used here as in Banerjee, *et al* (2007, p. 626).

The following rules are used for generating a *d-Distinction* table:

1. Insert 1 for object pairs of different classes having different values (e.g., {0,1} or {1,0}).

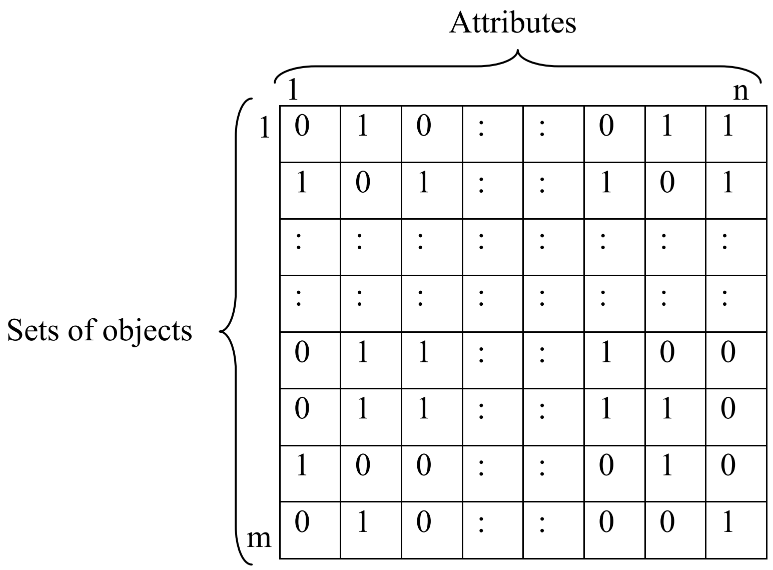
   $$b\big((k,j),i\big) = 1 \text{ if } a_i(x_k) \neq a_i(x_j)$$

2. Insert 0 for object pairs of different classes having the same value (e.g., {1, 1} or {0,0}).

   $$b\big((k,j),i\big) = 0 \text{ if } a_i(x_k) = a_i(x_j)$$

3. Insert 0 for either object of the pair has "*".
4. Object pairs of the same class are ignored.
5. Rows with all 0s are not allowed.

A *d-Distinction* table created using the above criteria has a smaller dimension of m_1 * m_2 in comparison to a discernibility matrix of $(m*(m\text{-}1))/2$, where $m = m_1+m_2$, and m_1, m_2 = the number of samples in class 1 and 2, respectively. For example, let m_1 is 22 and m_2 is 40, therefore $m_1 * m_2$ = 22 * 40 = 880 rows (sets) of objects, which are much less than ((22+40) * (22+40)-1))/2 = (62*61)/2= 1891. Therefore, in terms of search space, it would reduce computational cost when using GA to find the optimal feature subset. The following figure shows an example of a cut down version of *d-Distinction* table.


Attributes
Sets of objects

	1							n
1	0	1	0	:	:	0	1	1
	1	0	1	:	:	1	0	1
	:	:	:	:	:	:	:	:
	:	:	:	:	:	:	:	:
	0	1	1	:	:	1	0	0
	0	1	1	:	:	1	1	0
	1	0	0	:	:	0	1	0
m	0	1	0	:	:	0	0	1

Figure 4-6 An example of d-*Distinction* table

4.1.3. Phase 3: Feature selection via GA search optimization

The first task of phase 3 is to determine the representation of chromosomes. There are different types of chromosome representations. These include binary, integer, real numbers, single character, and permutation representation. One of key components in application of GA is the representation of the solution using chromosomes (Qin, 1999). This is due to the fact that GA searches for solutions (chromosomes) to solve a problem, so it is very difficult for GA to find an optimal solution with an unsuitable chromosome representation for the specific problem. In fact "*the use of different chromosome encoding schemes would lead to different search performances.*" (Chaiyaratana, Piroonratana, & Sangkawelert, 2007, p. 3).

The aim here is to process the *d-Distinction* table for sets of relevant features associated with high dimensional biomedical data using GA. In the studies of Felix and Ushio (1999), Duval & Hao (2010) and Perez and Marwala (2012), strings of *n* binary bits (binary chromosomes), e.g. {0 1 0 1 0 1 0 1 1 1 0 0}, were used to represent solutions for GA. The length *n* is the number of features (genes) in the dataset. That is, a binary chromosome represents a set of features and binary bits (gene values) of 0s in the chromosomes indicate features are not present (not selected for classification), and binary bits of 1s indicate features are present (selected for classification) in relation to the dataset (Banerjee et al., 2007; Deb & Reddy, 2003; Duval & Hao, 2010; Liu & Iba, 2002; Perez & Marwala, 2012; Vafaie & De Jong, 1992). For example, a chromosome {0 1 0 1 0 1 0 1 1 1 0 0} represents a total of 12 genes in the dataset and genes with value of "1" (2nd, 4th, 6th, 8th, 9th and 10th genes) are used for classification, whilst genes with value of "0" are not used. The following sections detail the steps associated with the application of GA.

a) Population initialization

The population of chromosomes is initialized by randomly selecting sets (rows) of objects from a *d-Distinction* table generated from the RST step. The number of sets of objects selected randomly from the distinction table equals the population size. That is, the number of sets of objects are randomly selected depending on the size of the population, e.g., if the size of the population is set to 100 (100 chromosomes) then 100 sets of objects are selected randomly. The following figure describes the algorithm used to initialize the population.

```
Input:
        d-Distinction table ($T_d$) of $m$ rows and $n$ columns
        Size of population ($p$)
Output:
        An initialized population of $p$ rows and $n$ columns
Steps:
   1. Set chromosomes as strings of binary of length $n$
   2. Set initial population of size $p$ ($I_p$) = {∅}
   3. For counter from 1 to $p$
        a. Generate an integer random number ($R_n$) in the range [1, $m$] using a
                RNG
        b. Search indexes of rows (sets of objects) in $T_d$ using $R_n$
        c. Select row[$R_n$] in $T_d$
                d. Store the selected row to $I_p$
```

Figure 4-7 Algorithm for initialisation of population using *d-Distinction* table

b) Fitness evaluation

Fitness function in RST-GA, f, of a chromosome is defined using the formula shown in Equation (4.3).

$$f = w_1 * f_1 + w_2 * f_2 \tag{4.3}$$

Where w_1 and w_2 are the weightings for f_1 and f_2, respectively, with $w_1 + w_2 = 1$

f_1 and $f_2$are the objective function 1 and 2, respectively

f is an overall objective function

The fitness of a chromosome, f, is defined as an aggregation of two objective functions, f_1 and f_2. The objective function f_1 is for maximizing the fitness of chromosomes (sets of features) with the least number of “1”s (features), whilst objective function f_2 is for maximizing the fitness of chromosomes that discerns the most number of objects, i.e, maximizing accuracy. Thus, the objective function, f, guides GA to find an optimal subset of relevant features that has the least number of features but gives higher accuracy in discerning between objects .

Since the objective function is an aggregation of 2 objective functions, f_1 and f_2, associated weightings, w_1 and w_2, are assigned to f_1 and f_2, respectively. These weightings of f_1 and f_2 would affect the search optimization process for finding optimal chromosomes (solutions). Therefore, an empirical experiment for obtaining the appropriate values of w_1 and w_2 was conducted in this study. Different combinations of w_1 and w_2 values were applied, e.g., w_1 = 0.1 and w_2 = 0.9 or w_1 = 0.2 and w_2 = 0.8, etc. As a result, w_1 = 0.9 and w_2 = 0.1 were found to work best for the 2 datasets used in this study. Coincidentally, these two weighting are the same as those used in Banerjee *et al.* (2007)'s experiment and allowed a comparison with results from Banerjee *et al.*'s. As RST-GA incorporated quartile statistics and K-means methods to partition data, an approach different from Banerjee *et al.*, this comparison of results allows an examination of the effectiveness of using quartile statistics and K-means for partition data in phase 1 of the approach.

Equation (4.4) and (4.5) define objective functions f_1 and *f2,* respectively (Banerjee *et al.*, 2007).

$$f_1(\underset{v}{\to}) = \frac{N-L\underset{v}{\to}}{N} \tag{4.4}$$

$$f_2(\underset{v}{\to}) = \frac{C\underset{v}{\to}}{(m_1 * m_2)} \tag{4.5}$$

where

N is the length of the candidate chromosome

$L\underset{v}{\to}$ is a number of "1"s in the candidate chromosome

$m_1$and m_2 are the number of objects belonging to class 1 and 2, respectively

$C\underset{v}{\to}$ is a number of objects distinguished by the candidate chromosome.

The following Figure 4-8 describes the algorithm used to calculate the fitness of chromosomes.

```
Input:
        d-Distinction table ($T_d$) of $m$ rows and $n$ columns
        Initial population ($p$)
Output:
        Fitness of chromosomes as an array of $p$ rows and $n$ columns
Steps:
   1. Set Size = size of population, $p$
   2. Set weighting for $w_1$ and $w_2$ with $w_1+w_2=1$
   3. Set $f$ = fitness of chromosome
   4. Set fitness population ($F_p$) = {∅}
```

```
5. For counter from 1 to Size
    a. Calculate objective function $f_1$ using  Equation (4.4)
    b. Calculate objective function $f_2$ using Equation (4.5)
    c. Calculate fitness of chromosome $f$ using Equation (4.3)
    d. Store $f$ into fitness population ($F_p$)
```

Figure 4-8 Algorithm for fitness calculation using f_1, f_2 and f objective functions

c) GA operators

Selection, crossover and mutation operators for binary-value encodings are used in the proposed approach. Tournament selection is a simple but efficient operator that has been commonly used in the GA (Miller & Goldberg, 1995). In the proposed approach, the tournament selection is employed to select 2 chromosomes from the population for crossover operation. Two chromosomes are selected randomly from the population of size *k*, a fitter chromosome is then selected for crossover. The tournament selection has been described in Section 2.3.4.1.

Single point crossover (Back, Hoffmeister, & Schwefel, 1991) is a technique that can be applied to binary value encodings to exchange parts of two chromosomes at a randomly selected crossover position. That is, 2 selected parents are split into 2 parts at the crossover position and then the second part of the 2 parents is inter-changed to produce 2 offspring. Single point crossover has been described in Section 2.3.4.1 and is employed in this study to recombine the chromosomes using the probability rate (P_c) as listed in Table 4-1.

Bit-flip mutation is the most common mutation operator used for binary encoded chromosomes. The bit value of a gene is flipped, i.e., if the bit value is 0 then change it to 1 and vice versa, independently based on a predefined mutation rate, (Eiben & Smith, 2007). As a result, mutated offspring are produced by the process of mutation. The bit-flip mutation has been described in Section 2.4.4.1 and is employed in this study to modify the chromosomes using the mutation rate (P_m) as listed in Table 4-1.

d) New population generation

The 2 best chromosomes are selected from the pool of parents and resulting offspring obtained from the previous step involving selection, crossover and mutation. These are then placed into the new population. Also a single elitist strategy is employed in this study to allow the best candidate solution in the previous generation to be retained and placed into the new generation to improve the search in evolutionary algorithms (Ahn & Ramakrishna, 2010). The process of selection, crossover and mutation continue until the generation of the new population is completed. The following figure describes the procedure to generate a new population.

```
Input:
        Chromosome population ($p$)
        Fitness population ($F_p$)
        Crossover probability ($P_c$)
        Mutation probability ($P_m$)
        Elite chromosome (Elite)

Output:
        New population ($N_p$)
Steps:
        1. Set Size = size of population, $p$
        2. Set new population ($N_p$) = {∅}
        3. Store Elite into $N_p$
        4. For counter from 1 to ½ Size
            a. Select 2 parent chromosomes from $p$
                • Perform tournament selection to select $parent_1$, $parent_2$
            b. Create 2 offspring chromosomes using $parent_1$ and $parent_2$
                b1. Generate a random number ($R_n$) in the range [0, 1] using RNG
                b2. If   $P_c \geq R_n$
                    • Perform one point crossover on 2 parents to produce 2 offspring
                b3. If   $P_m \geq R_n$
                    • Perform bit-flip mutation on each bit of offspring
                b4. Evaluate fitness of parent and offspring chromosomes
            c. Store the best 2 chromosomes into $N_p$
```

Figure 4-9 Algorithm for generating a new population

e) Checking for convergence in NSC-GA

The process of fitness evaluation, selection, crossover, mutation and new generation is repeated until the convergence of population in fitness takes place or a predefined maximum number of generations have been executed. The procedure of verifying the convergence status and terminating the GA have been described in Section 3.3 and 3.4, respectively. Upon the convergence the fittest chromosome (optimal solution) is selected.

In order to evaluate the proposed approach, a suitable set of parameter values are needed for all the parameters associated with GA. The following section describes the parameter tuning process to obtain the best parameter set for GA.

4.1.4. Parameter tuning for the GA

Convergence of fitness in the GA is important as premature convergence will result in a local optimal solution. Parameter tuning for the GA is necessary to ensure that the algorithm has executed using the best parameter setting, owing to the fact that the crossover probability (P_c) and the mutation probability (P_m) are vital for the optimal performance of the GA (M. Srinivas & Patnaik, 1994). Parameter tuning for P_c and P_m in this study is completed using the approach proposed by Srinivas and Patnaik (1994). This process involves varying different values of P_c and P_m, and observing the difference value between the average fitness (f_{avg}) and the maximum fitness value (f_{max}) of the population to verify the convergence status for GA (i.e. local or global convergence). As a result of this parameter tuning, the best parameter values of P_c and P_m are found to be 0.7 and 0.03, respectively. The complete set of parameter values used to run RST-GA in this study is shown in Table 4-1.

Table 4-1 A set of parameters used to run GA

Parameters	Values / Methods
Population Size	100
Chromosome length - Binary Encoding	The number of reduced attributes (genes)
Pc	0.7
Pm	0.03
Generation	1000
Selection	Tournament
Crossover	Single point
Mutation	Bit-flip
Elitist	Single

4.2. *Experiment results*

The proposed approach was evaluated using both the Colon and Leukemia cancer datasets described in Chapter 3. For each dataset, 15 independent runs of the proposed approach were executed using the respective training data. The optimal set obtained for each run was used to construct corresponding k-NN classifier with k = 1, 3, 5, and 7. The selected optimal set of features is evaluated using 10 fold CV strategy described in Section 3.2 and then further evaluated using unseen test datasets.

Steps in conducting one run of the experiment involved:

- Invoke the algorithm using an input training file and parameter setting file. The training dataset and unseen test dataset are prepared based on the procedure outlined in Section 3.1 and 3.2.
- Use the optimal feature set, to construct corresponding k-NN classifier with k = 1, 3, 5, and 7. Record results. Repeat this step using another training fold until 10 fold CV has been completed.
- Use the optimal feature set, to construct corresponding k-NN classifier with k = 1, 3, 5, and 7 to classify the unseen test dataset respectively.

The classification results for classifying the unseen test data using each of these k-NN classifiers were recorded and shown in Table 4-2. The following sections detail the results obtained from applying the approach on each of the two datasets.

4.2.1. *Alon et al.* Colon cancer data

Banerjee, *et al*. (2007) split the colon cancer dataset, with 50% for training and 50% as the unseen test dataset. Each of these two datasets consists of 20 Cancer (C1) samples and 11 Normal (C2) samples. In this study, the Colon cancer dataset was partitioned in the same way as that of Banerjee, *et al.*'s (2007).

Using the parameter settings in Table 4-1 and the training data for the Colon cancer dataset, the optimal subset of features from each of the 15 independent runs of RST-GA were obtained, evaluated using 10 fold CV evaluation strategy described in section 3.2, as well as tested on the unseen test set. The k-NN classifier with different k values of 1, 3, 5 and 7 were used to classify the Colon unseen test dataset. The classification results obtained using the optimal set from each of the 15 independent runs on the unseen test data are shown in Table 4-2.

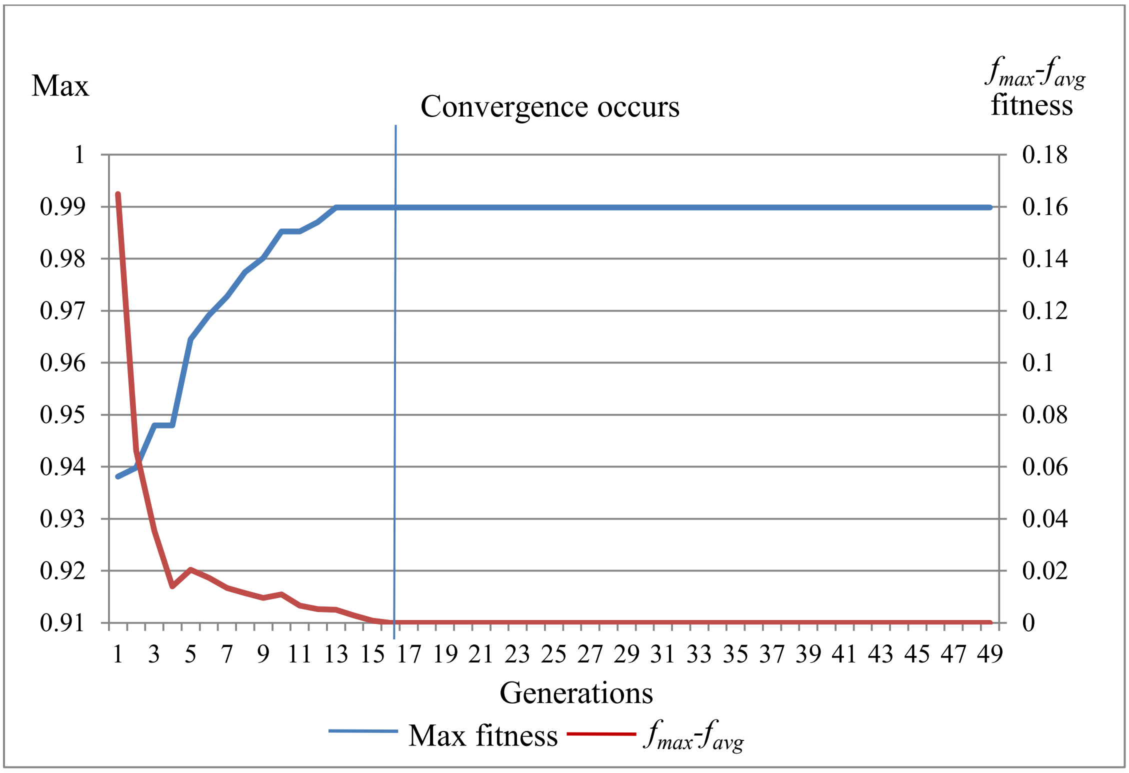


Figure 4-10 A typical convergence plot for maximum fitness and ($f_{max}-f_{avg}$) associated with the Colon dataset

A convergence plot from one of the typical runs is shown in Figure 4-10. As seen in this figure, the algorithm converged to a global optimum with the maximum fitness of 0.99 (approaching the theoretical maximum fitness of 1). The values of (f_{max} – f_{avg}) was relatively high (values shown on the right-hand vertical axis) in the earlier generations (<5) and it decreases to values very close to zero after 17 generations. This value coincides with the maximum fitness value of 0.99. Note that the convergence after 17th generations is due to the initial population of individuals being selected from the *d-Distinction* table, which consists of chromosomes (binary strings) that have already been processed for redundancy reduction of features.

Premature convergence occurs when the evolutionary algorithm (e.g. genetic algorithms) gets stuck in local optima and returns suboptimal solutions (Vanaret, Gotteland, Durand, & Alliot, 2013). On achieving global convergence, the population is genotypically very similar, thus individuals in the population has very similar fitness value. The state of convergence of each of the 15 runs of RST-GA is evaluated using the check mechanism outlined in Section 3.4, and typically, a plot like the graph in red in Figure 4-10 is obtained for f_{max} - f_{avg}. In addition, to gain a understanding of the behaviour of RST-GA across the 15 run, a whisker plot for the maximum fitness value of all 15 runs is shown in Figure 4-11 . At the point of convergence, the fitness value approaches the theoretical maximum fitness of 1 and the spread of the fitness value is very small across the 15 runs for each of those generations, thus showing global convergence.

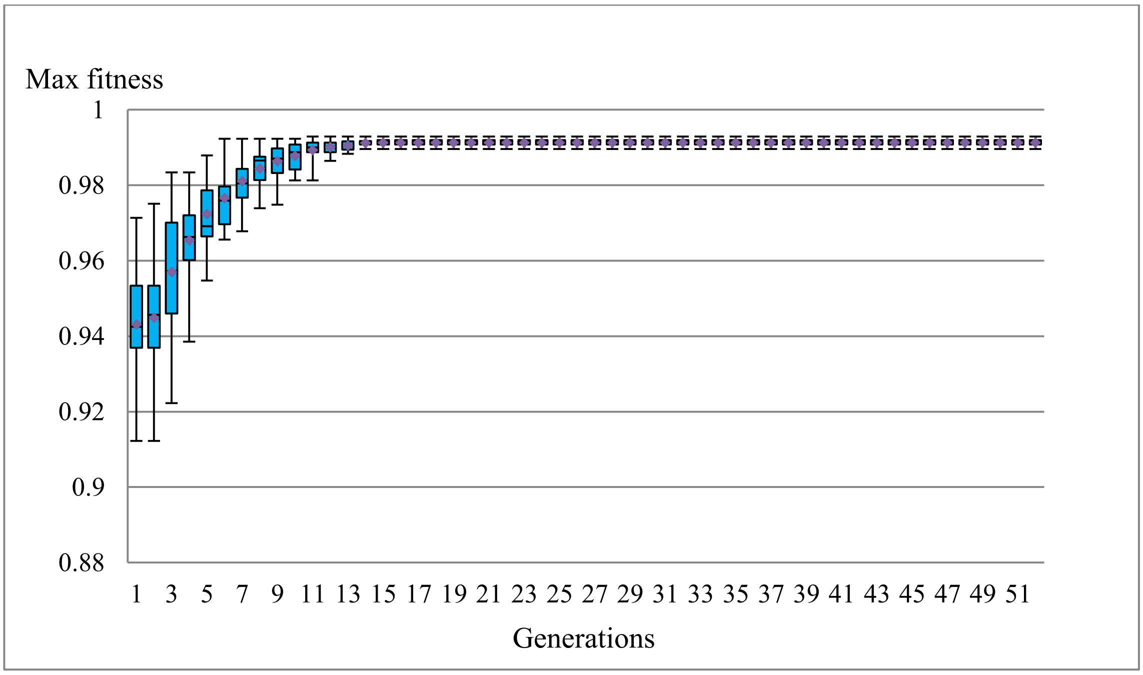
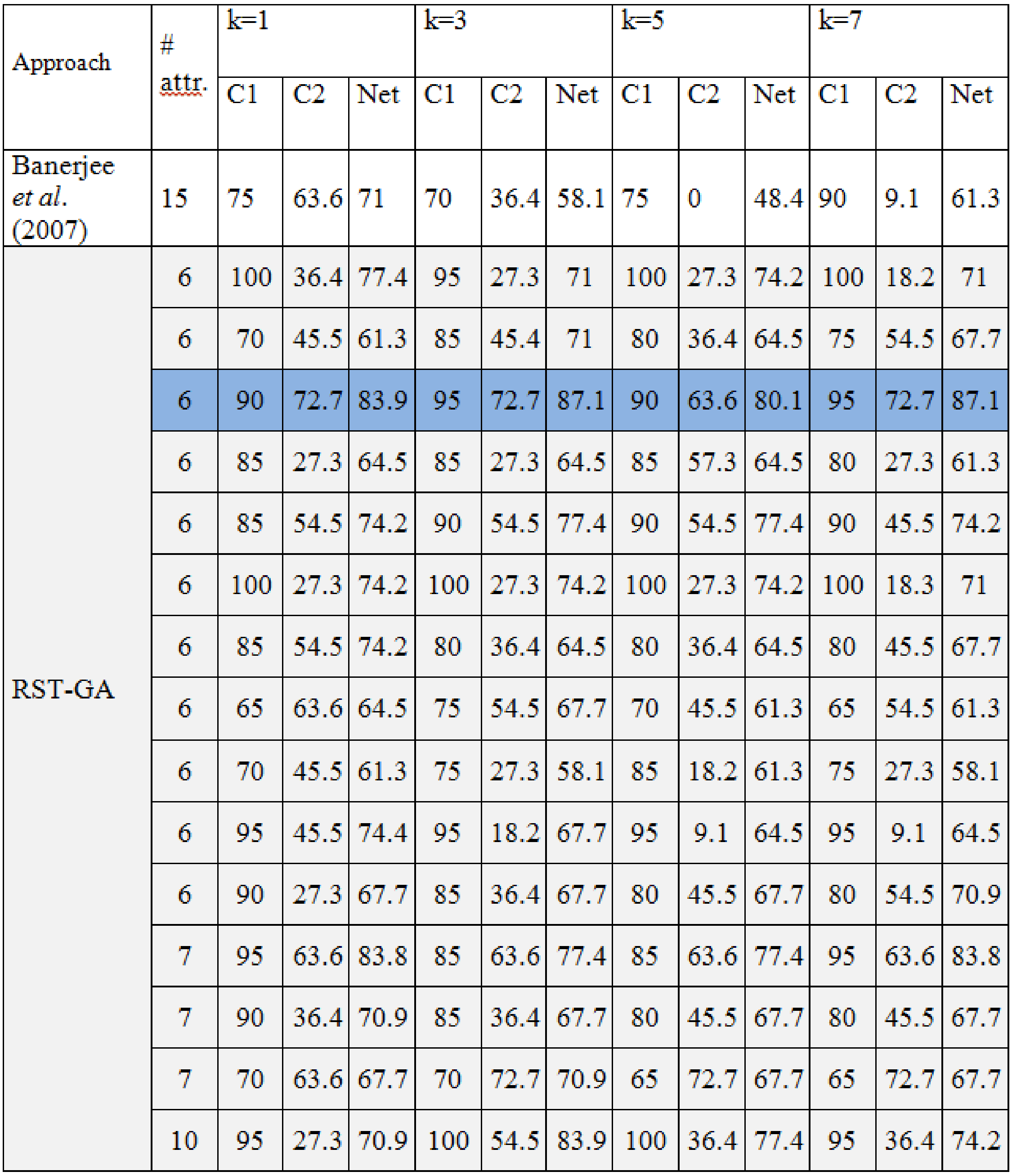


Figure 4-11 A whisker plot for maximum fitness for 15 runs

The fitness maximum value obtained at convergence for each of the 15 run of RST-GA for the Colon cancer dataset are 0.9907, 0.9917, 0.991, 0.992, 0.9907, 0.9914, 0.9914, 0.9922, 0.9929, 0.9917, 0.992, 0.9898, 0.9896, 0.9902 and 0.9836 respectively. The first value of 0.9907 is associated with Set 1 in Table 4-3, 0.9917 with Set 2, and the mapping of the sets following this order until with Set 15 being mapped to 0.9836.

Table 4-2 Results associated with RST-GA (proposed approach) and from Banerjee, *et al*. (2007) using k-NN classifiers, with k=1, 3, 5, and 7 on the Colon unseen test set



Approach	# attr.	k=1			k=3			k=5			k=7		
		C1	C2	Net	C1	C2	Net	C1	C2	Net	C1	C2	Net
Banerjee *et al.* (2007)	15	75	63.6	71	70	36.4	58.1	75	0	48.4	90	9.1	61.3
RST-GA	6	100	36.4	77.4	95	27.3	71	100	27.3	74.2	100	18.2	71
	6	70	45.5	61.3	85	45.4	71	80	36.4	64.5	75	54.5	67.7
	6	90	72.7	83.9	95	72.7	87.1	90	63.6	80.1	95	72.7	87.1
	6	85	27.3	64.5	85	27.3	64.5	85	57.3	64.5	80	27.3	61.3
	6	85	54.5	74.2	90	54.5	77.4	90	54.5	77.4	90	45.5	74.2
	6	100	27.3	74.2	100	27.3	74.2	100	27.3	74.2	100	18.3	71
	6	85	54.5	74.2	80	36.4	64.5	80	36.4	64.5	80	45.5	67.7
	6	65	63.6	64.5	75	54.5	67.7	70	45.5	61.3	65	54.5	61.3
	6	70	45.5	61.3	75	27.3	58.1	85	18.2	61.3	75	27.3	58.1
	6	95	45.5	74.4	95	18.2	67.7	95	9.1	64.5	95	9.1	64.5
	6	90	27.3	67.7	85	36.4	67.7	80	45.5	67.7	80	54.5	70.9
	7	95	63.6	83.8	85	63.6	77.4	85	63.6	77.4	95	63.6	83.8
	7	90	36.4	70.9	85	36.4	67.7	80	45.5	67.7	80	45.5	67.7
	7	70	63.6	67.7	70	72.7	70.9	65	72.7	67.7	65	72.7	67.7
	10	95	27.3	70.9	100	54.5	83.9	100	36.4	77.4	95	36.4	74.2

Table 4-2 shows the results obtained via RST-GA and those obtained in Banerjee, *et al*.'s study (2007). The column headings C1 and C2 in the table stand for classification accuracy (%) on the Colon unseen test dataset for the Cancer class and Normal class, respectively. The column heading "Net" stands for the overall classification (%) for all classes on the Colon unseen test dataset and "Net" is calculated using Equation (4.3).

$$\text{Net (\%)} = \frac{TP + TN}{TP + FP + TN + FN} \tag{4.3}$$

where *TP* is true positive for correct prediction to C1 class

TN is true negative for correct prediction to C2 class

FP is false positive for incorrect prediction to C1 class

FN is false negative for incorrect prediction to C2 class

From 15 independent runs, RST-GA found 15 sets of features consisting of 1 set of 10 features, 3 sets of 7 features and 11 sets of 6 features. Each of the 15 sets is used to train a k-NN classifier with k = 1, 3, 5, and 7 and each row of Table 4-2 is associated with the classification results obtained on the unseen test dataset. The row highlighted (in blue) in Table 4-2 shows the highest classification accuracy obtained for classifying the unseen test set. It can be seen that the proposed algorithm found a smaller set with 6 genes that gave a higher classification accuracy in comparison to those involving the larger set of 15 features reported by Banerjee, *et al.* (2007). It is not possible to evaluate whether there are any commonality in the sets of features found by RST-GA, with the set obtained by Banerjee *et al.* (2007) as the list of their 15 features is not listed in their paper.

In addition to examining the importance of the classification performance of a set of features, its relevance to its corresponding domain is crucial. Table 4-3 lists the selected genes by accession numbers for the 15 sets found by the proposed approach. As seen from the table, some genes are common across a number of these sets (e.g. H08393 are found in set 1, 3, 5, and 12). These are coded in the same colour in the table for ease of identifying them in the different sets.

Table 4-3 List of genes associated with the Colon Cancer dataset for each of the 15 optimal sets of features obtained from 15 independent runs of RST-GA

Set #1	Set #2	Set #3	Set #4	Set #5	Set #6	Set #7	Set #8
M76378	X86779	U31248	T93589	R28373	M76378	R46753	R46753
M23254	R46753	L08069	M88279	M80815	M23254	M23254	H87344
L08069	H87344	H49870	M22632	L08069	L08069	T59162	M23254
H49870	M80815	M18216	U31248	H49870	R55310	L08069	T59162
M22538	H45977	M22538	L08069	D13665	L33930	R55310	M80815
H08393	T54303	H08393	T54303	H08393	H64807	T54303	T54303

Set #9	Set #10	Set #11	Set #12	Set #13	Set #14	Set #15
R46753	T63508	T71025	H86060	T71025	X63432	H64398
J03210	X86779	M76378	T71025	M76378	T71025	R28373
M80815	M76378	M23254	X87159	M23254	R46753	T48014
H45807	M22632	T74896	T52003	T74896	T74896	T53412
R55310	U31248	R93337	M76378	R93337	R93337	J03210
T54303	T54303	R55310	H58397	R55310	R55310	H73758
			H08393	T54303	T54303	X79981
						M92287
						M84721
						M33210

As seen in Table 4-3, the feature set obtained from each of 15 runs of the RST-GA is different, with only a small number of features in common across the different sets. This is due to a characteristic associated with feature selection methods, namely, the stability of feature selection methods. Stability is a term used to describe the sensitivity of a feature selection algorithm to small variations in the training data and in the settings of the algorithmic parameters, resulting in different feature sets being produced by the algorithm. Small variations in the training data include using a different partition of data samples, reordering of samples and adding/removing a few samples. In addition, in stochastic algorithms, using different random seeds and different parameter values will also result in different results from the algorithm. Both Rough Set theory and the GA are algorithms known to have feature selection instability.

An important point to note is that each execution of the RST-GA approach consists of three phases, and the application of GA is only in the third phase. The first two phases of RST-GA involved application of Rough Set Theory to generate the d-Distinction table, used as input, in the third phase (i.e. the GA phase). The initialization of population in the third phase (i.e. GA phase) involved 100 individuals randomly selected from the d-*Distinction* table (e.g. in the case of the Colon Cancer dataset, size of the d-*Distinction* table = 480). Thus the input to the GA phase is different in each of the 15 runs (besides having the random seed being different). Given that the GA is a feature selection instability method, different results is obtained in different runs since the input data is different. Other potential causes for the feature selection instability here is due to redundancy of features in high dimensional biological datasets, where multiple features contribute to the same diseased effect and with the availability of only a small number of samples in relation to the high number of features as exemplified by microarray datasets.

Set #3 ("Set #3" column highlighted Table 4-3) is one of the 15 sets selected using the proposed approach which gave the highest classification accuracy on the unseen test data. This set consists of 6 genes which have been reported in biomedical literature as being associated with cancer and other diseases. U31248 (*Human zinc finger protein (ZNF174) mRNA*) is related to the expression of colon tissues (Williams, Khachigian, Shows, & Collins, 1995). L08069 (*Human heat shock protein, E. coli DnaJ homologue mRNA*) is not only "shown to increase tumorigenicity in rat colon cancer" (GSAEmulator, n.d.), but also associated with tumour development in human (Diesinger *et al.*, 2002). H49870 (*yo24h10.s1 Soares adult brain N2b5HB55Y*) is involved in the detection of over-expression for olon cancer disease (Laping, 1999). The M18216 (*Human nonspecific crossreacting antigen mRNA*) is considered as a major component of Carcinoembryonic antigen involved in expression of lung cancer, tumour specimens, and tumour cell lines at mRNA levels (Hasegawa *et al*., 1993), and also increasing level of expression in Colon cancer (Hinoda *et al*., 1997). H08393 (*yl92a10.s1 Soares infant brain 1NIB)* is involved in the process of degrading activity of Colon cells. It is also one of 66 differently expressed genes for Colon cancer data (Shaik & Yeasin, 2007). M22538 (*Human nuclear-encoded mitochondrial NADH-ubiquinone reductase 24Kd subunit mRNA*) is involved in schizophrenia, bipolar disorder, and Parkinson disease (Nishioka *et al*., 2010) and is the only feature of this set that has not been shown to have an established linked to some form of cancer. This result may be the trigger for biological studies to include this feature for subsequent investigations.

Sensitivity and specificity associated with classification are two measures that are of great interest to the biomedical community in their efforts to find biological markers (also known as biomarkers) and to assess the utility of these biomarkers as to how well they can predict relevant outcomes. Sensitivity represents the probability of correctly diagnosing a condition (i.e. the proportion of truly affected (i.e. diseased) in a sample population that is identified by the test as being diseased). On the other hand, specificity represents the proportion of truly non-diseased that the test identified as such. Ideally, a biomarker should have high sensitivity and high specificity – resulting in the majority of the truly *at-risk* cases being correctly identified, and the majority of the truly *not-at-risk* cases also correctly identified as not having the diseased condition.

From Table 4-2, the k-NN classification results associated with each of the 15 sets of features mostly showed high sensitivity but low specificity, implying the majority of the truly *at-risk* cases will be correctly identified, but the majority of the truly *not-at-risk* cases will also be incorrectly identified as *at-risk*. For example, in the case of the highlighted row, sensitivity is 90% but specificity is only 72.7% for k = 1. A further investigation was carried out using the same set of 6 genes and 22 different classifiers from WEKA software (Hall *et al*., 2009) to classify the unseen test dataset. WEKA is a data mining software program that has been developed and maintained by WEKA team since 1994 (Markov & Russell, 2006). WEKA consists of a large number of classifiers that can be used to analyse datasets and perform classification. WEKA classifiers used in the thesis are categorized into six types of classifiers including Function, Bayes, Lazy, Meta, Rules and Tree classifiers (Hall *et al*., 2009). Function classifiers are simple and used for attributes with all numeric values, and a "linear boundaries between classes" strategy is used for classifying data. Bayes classifiers are implemented based on Bayes's rule for probability and use density estimators to map attributes to the probability. Lazy classifiers are simple and use a distance function to measure the distance between data points and classify data. Meta classifiers use weighting or voting or ensemble schemes to classify data, for example, AdaBoost classifier classifies data based on the class with highest total weight (Witten & Frank, 2005). Rules classifiers use "*a separate-and-conquer*" strategy to identify rules for classifying data (Beasley, Martin, & Bull, 1993, p. 171). Tree classifiers use "*the simple divide-and-conquer*" strategy to generate decision trees for classifying data (Beasley et al., 1993, p. 159) . Further details of WEKA classifiers can be found in Witten and Frank (2005) and Hall *et al*. (2009). The aim here is to see if this trend (as in the case of k-NN) in terms sensitivity and specificity is a result of using a specific classifier, in this case k-NN classifiers. The classification results for the 22 classifiers constructed using the same set of 6 features as that in training the k-NN classifiers are shown in Table 4-4. Note that Multilayer Perceptron, Decorate, Random Committee and Random Forest are classifiers that may return (slightly) different results from different runs, thus, these classifiers were executed 10 times with different seeds and results with * is an average on these 10 executions.

As seen in Table 4-4, mixed results were obtained. Using KStar (in bold) on the unseen test set produced results showing high sensitivity and high specificity (90%). However, there are also other classifiers showing behaviour similar to that of the k-NN classifiers. Also interestingly, there are a number of classifiers demonstrating higher specificity (shaded cells) than sensitivity. These results demonstrated that the use of specific classifiers may have an impact on the sensitivity and specificity. Thus in a DM analysis for finding suitable sets of biological markers, a number of classifiers should be used instead of just using one. This will avoid cases of missing out on sets of features with high discriminatory capabilities that should be further investigated in early diagnostic test developments but have been rejected on the basis of their sensitivity/specificity relating to a specific classifier.

Table 4-4 Results of classification for the 6 selected genes (highlighted in blue in Table 4-2) with 22 WEKA classifiers on the Colon unseen test set

Classifier	Set of 6 genes	
	C1	C2
SMO	70	81.8
Simple Logistic	65	100
Logistic	65	100
Multilayer Perceptron	83.5*	89*
Bayes Net	85	18.2
Naïve Bayes	80	63.6
Naïve Bayes Simple	80	63.6
Naïve Bayes Up	80	63.6
IB1	90	72.7
KStar	**90**	**90.9**
LWL	70	63.6
AdaBoost	80	63.6
ClassVia Regression	80	63.6
Decorate	85*	58.2*
Multiclass Classifier	65	100
Random Committee	77.5*	55.4*
j48	90	54.5
LMT	65	100
NBTree	80	54.5
Part	90	54.5
Random Forest	79*	59.9*
Ordinal Classifier	90	54.5

4.2.2. Leukemia cancer data

Using the same approach as outlined in Section 4.2, 15 independent runs involving RST-GA were carried out using the parameter settings in Table 4-1 and the Leukemia training dataset. The optimal subsets of features were obtained and evaluated using 10 fold CV evaluation strategy described in Section 3.2, as well as tested on the unseen test dataset. The k-NN classifier associated with each of the optimal subsets of features and with k values of 1, 3, 5 and 7 were used to classify the Leukemia cancer unseen dataset. A convergence plot from one of the 15 independent runs is shown in Figure 4-12 and the classification results of the 15 runs are shown in Table 4-5.

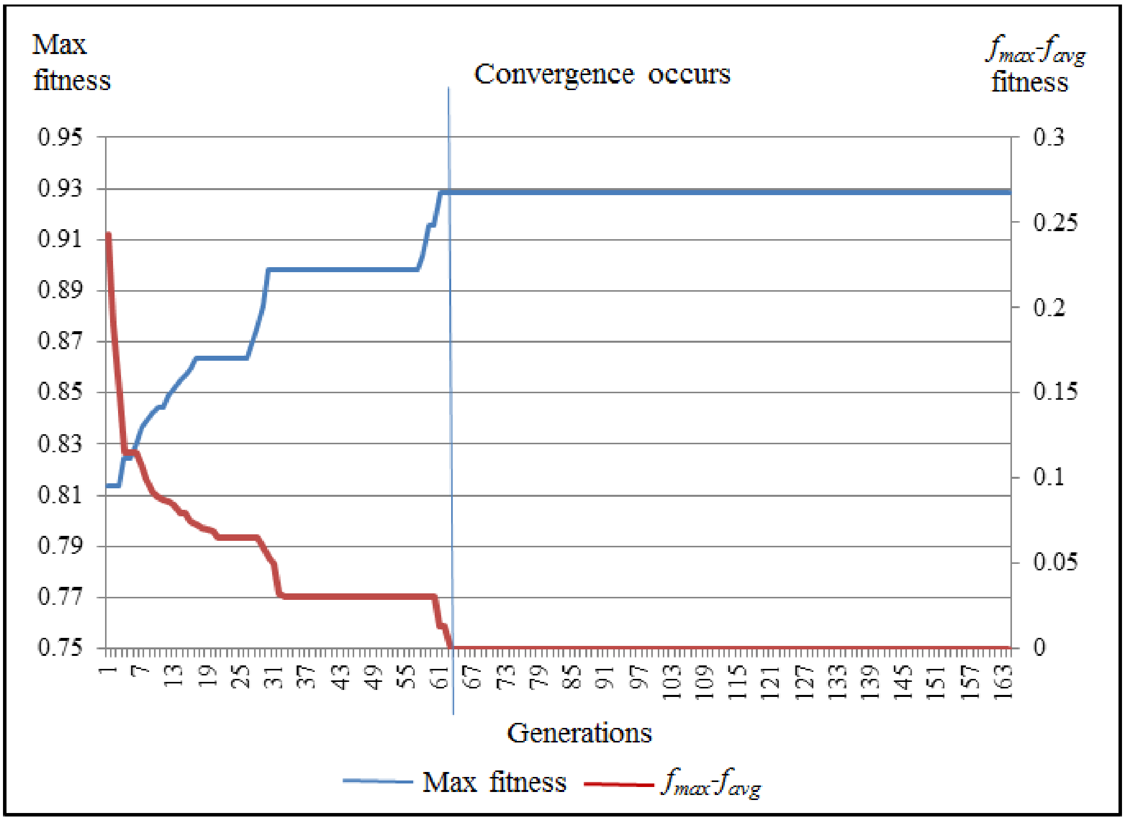


Figure 4-12 A typical state of convergence plot for maximum fitness and ($f_{max} - f_{avg}$) values associated with the Leukemia cancer dataset

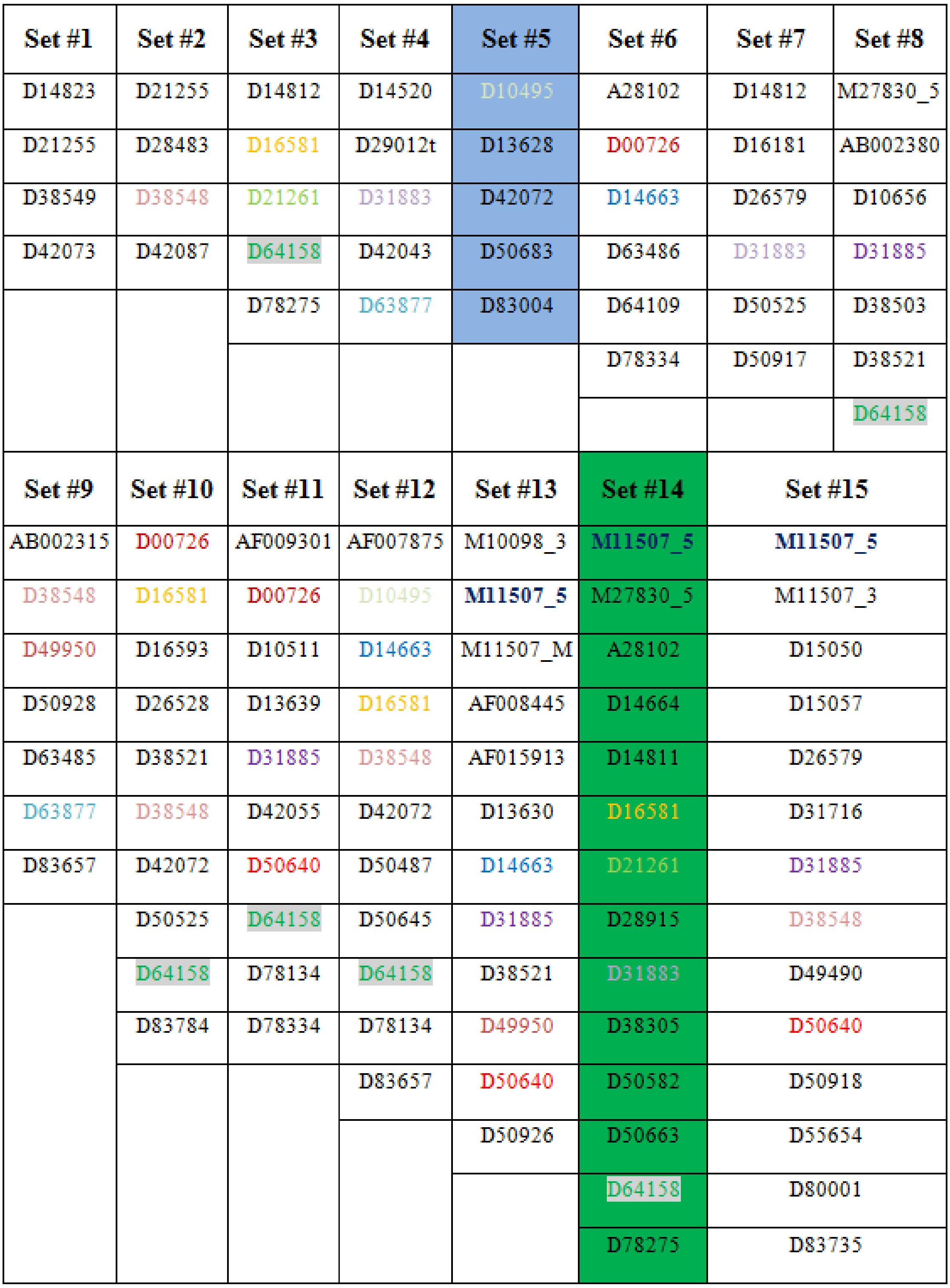
As seen in Figure 4-12, the algorithm converged to a global optimum with the maximum fitness value of 0.928. The value for ($f_{max} - f_{avg}$) was relatively high (values shown on the right-hand vertical axis) in the earlier generations (<7) and it decreases to values approximately close to zero around 64 generations. This value coincides with the maximum fitness having a value of 0.928 (1 is the maximum). Similar to the Colon dataset, the convergence is also quick although it occurred after 64 generations in comparison with 17 generations for the Colon dataset. This is due to the fact that samples in the Leukemia dataset have a larger number of genes (features), 7129, compared to 2000 features for the Colon cancer data. The 15 optimal sets of features obtained from 15 runs for the Leukemia cancer data using RST-GA, consisted of 2 sets of 4, 6, 7, 10 and 14 features, 3 sets of 5 features, 1 set of 11 features, and 1 set of 12 features. Each row in Table 4-5 is associated with classification results for the unseen test dataset using the k-NN classifier constructed from one of the 15 sets of features. The proposed algorithm found a set of 5 and 14 genes (rows highlighted in blue and green in the table, respectively) that gave a similar classification accuracy compared to those involving the larger set of 19 genes reported by Banerjee, *et al.*(2007). It is not possible to evaluate whether there are any commonality in the sets of features found by RST-GA, with the set obtained by Banerjee *et al.* as the list of their 19 features is not listed in their paper. Table 4-5 shows the classification accuracy for the unseen test data for the classifier associated with each of the 15 sets. The lists of genes associated with each set are shown in Table 4-6 by their accession number. Again, the differences between the sets of selected features from each of the 15 runs are due to the same reasons as outlined in the analysis of the Colon Cancer dataset.

As seen from Table 4-5, the k-NN classification accuracies associated with classifiers of the set of 5 genes (the row highlighted in blue) and 14 genes (the row highlighted in green) obtained from RST are compatible with the classification accuracies associated with classifiers of the set of 19 features reported in Banerjee, *et al*. (2007).

Table 4-5 Results for RST-GA (proposed approach) and Banerjee, *et al.*(2007) using for k-NN classifier with k=1, 3, 5, and 7 on the Leukemia unseen test set. The column heading "Net" stands for the overall classification (%) for all classes on the Leukemia unseen test dataset and "Net" is calculated using Equation (4.3).

Approach	# attr	k=1			k=3			k=5			k=7		
		C1	C2	Net	C1	C2	Net	C1	C2	Net	C1	C2	Net
Banerjee *et al.* (2007)	19	90	50	73.5	90	57.1	76.5	95	14.3	61.7	100	14.3	64.7
RST-GA	4	70	7.1	44.1	55	21.4	41.2	55	14.3	38.2	90	0	51.9
	4	85	35.7	64.7	85	35.7	64.7	90	28.6	64.7	95	35.7	70.6
	5	90	28.6	64.7	100	21.4	67.6	100	35.7	73.5	100	35.7	73.5
	5	65	42.9	55.9	90	14.3	58.8	95	21.4	64.7	95	21.4	64.7
	5	100	35.7	73.5	100	28.6	70.6	100	21.4	67.6	100	14.3	64.7
	6	75	42.9	61.8	85	0	50	95	0	55.9	96.3	45.5	55.9
	6	80	50	67.6	95	42.9	73.5	95	35.7	70.6	100	28.6	70.6
	7	70	50	61.8	95	28.6	67.6	80	35.7	61.8	85	7.1	52.9
	7	95	35.7	70.6	100	21	67.6	100	28	70.6	100	21.4	67.6
	10	95	28.6	67.6	95	14.3	61.8	95	21.4	64.7	96.3	27.3	61.7
	10	85	50	70.6	90	7.1	55.8	85	7.1	52.9	95	0	55.9
	11	75	42.9	61.8	95	35.7	70.6	95	35.7	70.6	95	28.6	67.6
	12	65	57.1	61.7	75	57.1	67.7	70	50	61.7	90	50	73.5
	14	85	50	70.6	95	57.1	79.4	95	50	76.5	95	50	76.5
	14	75	64.3	70.6	85	64.3	76.5	85	50	70.6	85	57.1	73.5

Table 4-6 List of 15 sets of genes selected for the Leukemia cancer dataset using RST-GA.



Set #1	Set #2	Set #3	Set #4	Set #5	Set #6	Set #7	Set #8
D14823	D21255	D14812	D14520	D10495	A28102	D14812	M27830_5
D21255	D28483	D16581	D29012t	D13628	D00726	D16181	AB002380
D38549	D38548	D21261	D31883	D42072	D14663	D26579	D10656
D42073	D42087	D64158	D42043	D50683	D63486	D31883	D31885
		D78275	D63877	D83004	D64109	D50525	D38503
					D78334	D50917	D38521
							D64158

Set #9	Set #10	Set #11	Set #12	Set #13	Set #14	Set #15
AB002315	D00726	AF009301	AF007875	M10098_3	M11507_5	M11507_5
D38548	D16581	D00726	D10495	M11507_5	M27830_5	M11507_3
D49950	D16593	D10511	D14663	M11507_M	A28102	D15050
D50928	D26528	D13639	D16581	AF008445	D14664	D15057
D63485	D38521	D31885	D38548	AF015913	D14811	D26579
D63877	D38548	D42055	D42072	D13630	D16581	D31716
D83657	D42072	D50640	D50487	D14663	D21261	D31885
	D50525	D64158	D50645	D31885	D28915	D38548
	D64158	D78134	D64158	D38521	D31883	D49490
	D83784	D78334	D78134	D49950	D38305	D50640
			D83657	D50640	D50582	D50918
				D50926	D50663	D55654
					D64158	D80001
					D78275	D83735

Table 4-6 lists the selected genes by accession numbers for the 15 sets found by RST. The highlighted columns (“Set #5” in blue and “Set #14” in green) in the table are the corresponding set of 5 and 14 genes associated with classifiers that produced compatible unseen test classification accuracies in rows highlighted in blue and green, respectively, in Table 4-5. As seen from the table, some genes are common across a number of these sets, e.g., D64158 are found in Set# 3, 8, 10, 11, 12 and 14 (shaded cells). These common genes are coded in the same colour in the table for ease of identifying them in the different sets.

As seen in from Table 4-6, Set #5 ("Set #5" column highlighted in Table 4-6) is one of the 15 sets obtained using the proposed approach, which gave a high classification accuracy with a smaller number of features. This set consists of 5 genes which have been reported in the literature as being associated with cancer: D10495 (*protein kinase C delta-type*) is the gene whose expression is commonly down-regulated in acute Adult T-cell leukemia (ATL) (Tsukasaki *et al.*, 2004), D13628 (*Angiopoietin 1*) is the gene that is over-expressed in *extramedullary plasmacytomas* (Hedvat *et al.*, 2003), D42072 (*Neurofibromatosis type 1 (NF1)*) is known as an autosomal disorder gene and highly associated with malignancy (Suzuki *et al.*, 1995), D50683 (*Alteration of the transforming growth factor β (TGFB)*) is a down-regulated gene that modifies expression and effects of TGFB in pancreatic carcinomas (Albrechtsson, Axelson, Heidenblad, Ludmilagorunova, & Höglund, 2001) and D83004 (*ubiquitin-conjugating enzyme E2*) is one of Atherosclerotic phenotype determinative genes that can be used in diagnosis, treatment and drug screening methods for Atherosclerosis (West, Nevins, Goldschmidt, & Seo, 2005). No existing information found to indicate its role in terms of this disease.

Also seen in Table 4-5, similar to the Colon cancer data, the k-NN classification results associated with the selected set of 5 and 14 genes mostly showed high sensitivity but low specificity when classifying the unseen test data. For example, the k-NN classifier involving the set of 5 selected genes showed 100% classification accuracy for C1 and 35.7% for C2 (i.e. high sensitivity and low specificity) . Further investigation, similar to that conducted with the Colon cancer dataset, was also carried out here using the sets of 5 and 14 genes to train 22 classifiers from WEKA and then to use them to classify the Leukemia unseen test dataset. The classification results of 22 classifiers are shown in Table 4-7.

Table 4-7 Results of classification for the 5 and 14 selected genes (highlighted in blue and green, respectively, in Table 4-6) with 22 WEKA classifiers on the Leukemia unseen test set

Classifier	Set of 5 genes		Set of 14 genes	
	C1	C2	C1	C2
SMO	100	14.3	95	64.3
Simple Logistic	100	14.3	80	50
Logistic	100	14.3	90	64.3
Multilayer Perceptron	100*	7.1*	85*	64.3*
Bayes Net	95	42.9	75	71.4
Naïve Bayes	100	14.3	80	57.1
Naïve Bayes Simple	100	14.3	80	57.1
Naïve Bayes Up	100	14.3	80	57.1
IB1	100	35.7	85	50
KStar	95	14.3	100	42.9
LWL	95	42.9	70	35.7
AdaBoost	100	42.9	95	50
ClassVia Regression	95	42.9	80	57.1
Decorate	100*	24.6*	86.7*	62.7*
Multiclass Classifier	100	14.3	90	64.3
Random Committee	100*	13.4*	95.5*	45*
j48	100	14.3	70	57.1
LMT	100	14.3	80	50
NBTree	100	14.3	80	57.1
Part	100	14.3	70	57.1
Random Forest	100*	17.3*	95.5*	46.4*
Ordinal Classifier	100	14.3	70	57.1

Again, it can be seen that the classification results of 22 WEKA classifiers constructed using the selected set of 5 and 14 genes showed mixed results in terms of sensitivity and specificity. The 22 classifiers associated with the set of 5 genes showed that they can classify the diseased instances (C1) very well and are very poor in classifying the non-diseased cases (C2) (i.e. high sensitivity and low specificity). For the classifiers associated with the set of 14 genes, most showed similar trends as the k-NN classifiers but one classifier (Bayes Net) showed similar specificity and sensitivity.

4.3. Discussion

As described in the previous section, the optimal set of features, generated from each independent RST-GA run, is then used with the training set to produce the various k-NN classifiers. These classifiers are then used to classify the unseen test set, with the classification results reported in Table 4-2 and Table 4-5 for the Colon cancer and the Leukemia datasets respectively. As shown in Table 4-3 and Table 4-6 Table 4-6, the optimal set of features obtained from each independent run of RST-GA has differences and varying degrees of overlap in terms of the selected features. This is a typical outcome when using a non-deterministic approach such as RST-GA. Potential benefits include: 1) the generations of smaller sets of features, (e.g. sets ranging from 4 to 14 features in comparison to the original dimensionality such as 7129 features in the Leukemia dataset), with high discriminatory capabilities that can be further investigated for early diagnostic test developments, and 2) examination of the overlap between the sets of features which then can lead to construction of feature sets for further investigations.

A number of observations emerged from examining the classification results in Table 4-2 and Table 4-5 relating to the issues of sensitivity and specificity of a classifier associated with selected set of features. First, it can be seen that different classifiers, trained using the same set of features, can produce different values for these two measures in their evaluation of a test dataset. Second, as demonstrated in the analysis involving the Leukemia dataset, sets with different number of features (e.g. set of 5 and set of 14 genes) when used to train the same classifier will also produce different values for these two measures in their evaluation of a test dataset. For example in the case of the Bayes Net classifier, when trained with the set of 5 genes, the classification result showed high sensitivity (95%) and low specificity (42.9%) and when trained with the set of 14 genes, the sensitivity and specificity is not too different (75% versus 71.4 %). Yet, most of the remaining 21 classifiers when trained with the same set produced classification results that showed high sensitivity and low specificity. This implies that decisions, in terms of evaluating sets of features to be further investigated in early diagnostic test developments, need to take into consideration these observations - to avoid eliminating potential sets of features during an early stage of investigation.

The proposed approach, RST-GA can be used as an exploratory tool in terms of the generation of multiple optimal sets with the most relevant features. By utilizing these sets of features to train multiple classifiers and followed by classification on unseen test datasets would provide biomedical researchers with more information about selecting potential sets for further investigation. The comparison of classification results of these different optimal sets with different classifiers in conjunction with domain knowledge could be the starting basis for further investigations and developments leading to development of panels of biomarkers related to a disease.

However, given the feature instability nature associated with RST-GA, resulting in feature sets obtained from each different runs of the RST-GA on a specific dataset being different and only having a small number of common features across the different sets. From examining the analysis involving two datasets, it was obvious that the degree of feature instability across different runs could be significant and the approach may not be most ideal to explore biomedical data for finding potential biomarkers. The decision was then to explore approaches that could work better with evolutionary approaches and with minimal feature instability.

4.4. Summary

This chapter describes the proposed approach of a hybrid algorithm (RST-GA) which incorporates GA and RST for finding the optimal subset of significant features. The approach utilizes the k-means clustering for getting the initial cluster centroids of each attribute for RST, the rough set-based approach for generating sets of good candidate solutions, and GA for finding the reducts (optimal subsets of features). The evaluation process used the same Colon and Leukemia cancer datasets as in Banerjee, *et al*. (2007). The set of 6 genes and 5 genes for Colon and Leukemia cancer data respectively, produced from the proposed approach, have similar classification results in comparison to those obtained by Banerjee, *et al*. (2007) using a larger number of features.

In the next chapter, an approach of incorporating the NSC algorithm with GA for searching for an optimal shrinkage threshold value that leads to the selection of an optimal set of features will be described.

5. Incorporating NSC and GA, NSC-GA

5.1. Introduction

Chapter 4 described an initial attempt in this study to develop FS techniques incorporating the use of evolutionary algorithms and RST for analysis of high-dimensional biomedical data. One limitation of this approach is that the result of the optimal set of features obtained is not constant in every independent run. This is a typical issue when employing a non-deterministic algorithm such as GA. To ensure less variability in the optimal set of features from each independent run, a deterministic method can be incorporated in the approach. This chapter is an extended version of the paper "NSC-GA: Search for Optimal Shrinkage Thresholds for Nearest Shrunken Centroid" (Dang, Lam, & Lee, 2013). It describes the second approach in this study that incorporates EA and a deterministic algorithm for analysing biological data. This hybrid approach incorporates the NSC method (Tibshirani *et al.*, 2002) and GA to automatically search for an optimal range of shrinkage threshold values for the NSC. The optimal shrinkage thresholds obtained are used in NSC to obtain a set of features. The feature sets obtained using this hybrid approach has less variability as in NSC, shrinkage threshold values with small differences map to the same feature set .

The NSC method, with its most well-known software implementation being known as Prediction Analysis for Microarrays (PAM), has been widely used as a FS and classification method for high dimensional biomedical data in numerous studies (Bair & Tibshirani, 2004; Klassen & Kim, 2009; Lee et al., 2005; Ravetti & Moscato, 2008; Ray et al., 2007; K. Y. Yeung & R. E. Bumgarner, 2003). A shrinkage threshold value must also be provided to the NSC method as input and normally, this is selected manually by executing the NSC method many times using a number of predetermined shrinkage threshold values. The optimal shrinkage threshold value is then obtained by minimizing the cross-validated error rate on the training data. This process can be time-consuming and the optimal shrinkage threshold value may be limited by the granularity of the predetermined values.

The selection of a shrinkage threshold value is crucial as the NSC works on the principle of shrinking the relative difference between the class centroid and the overall centroid of all classes, moving the class centroid towards the overall centroid of all classes using the shrinkage threshold value.

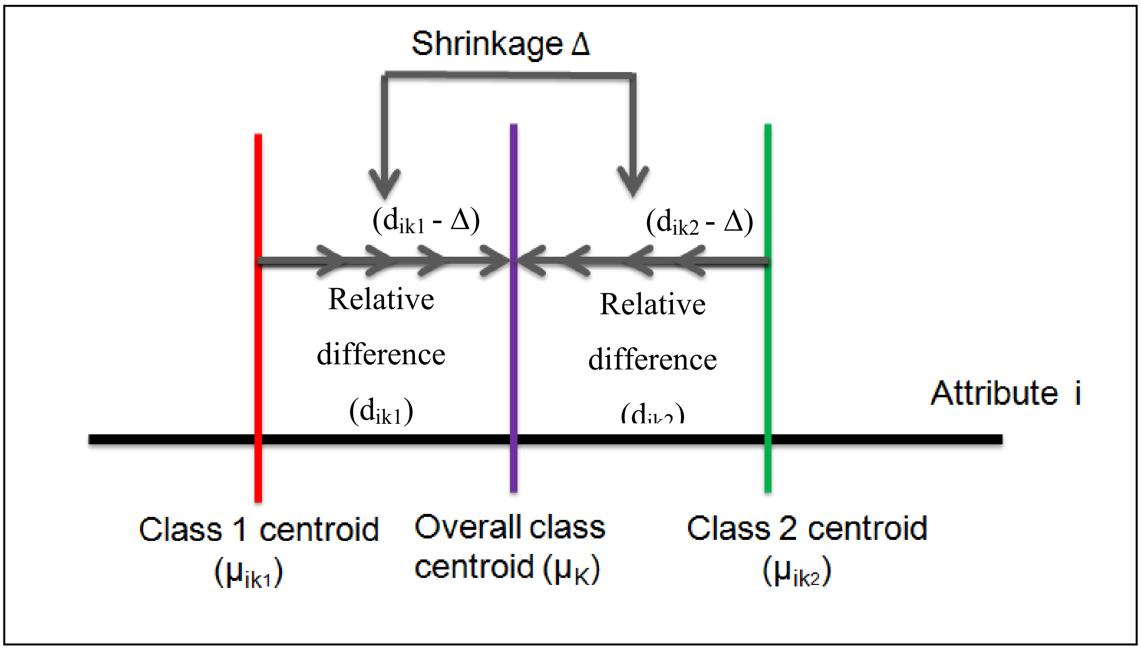


Figure 5-1 Shrinkage of threshold of 2 class centroid toward overall centroid in NSC

As seen in Figure 5-1, the class centroid μ_{ik1} and μ_{ik2} (associated with Class 1 and Class 2 respectively) of attribute *i* are shrunk toward overall class centroid (μ_K) by a shrinkage threshold (Δ) value iteratively. The relative difference, $d_{ik1}$and d_{ik2}, is the distance between the class centroid, μ_{ik1} and μ_{ik2}, and the overall centroid, μ_k, respectively. If the relative difference of an attribute is shrunk to zero for all associated classes, then it is considered as **not** an important attribute and is eliminated (i.e. class centroids and overall class centroid are not different). Attributes with at least one positive relative shrunken class centroid are considered as important attributes and are selected (i.e. class centroids and overall class centroid are different).

The shrinkage threshold value for NSC is important in terms of FS and classification as it affects the selection of features. Using inaccurate shrinkage threshold values will lead to irrelevant features being selected and subsequently will lead to a lower classification accuracy. Two approaches, CV (Tibshirani et al., 2002; S. Wang & Zhu, 2007; K. Yeung & R. Bumgarner, 2003) and empirical approach (Klassen & Kim, 2009; Levner, 2005; Ray *et al.*, 2007) are normally used to find the shrinkage threshold values. With the CV approach such as 10 fold CV, the dataset is divided randomly equal into 10 parts, each part consists of approximate proportion of a number of samples and classes. One part takes turn to be the test set while the other 9 parts are used as the training set. The procedure is repeated 10 times to obtain the prediction error rate for each time. The overall prediction error rate is then calculated by averaging the errors from all iterations. The selected optimal shrinkage threshold value is based on the CV prediction errors associated with the different shrinkage threshold values. The shrinkage threshold value that gives the minimum CV prediction error is selected as the optimal shrinkage threshold value. For example, in Tibshirani, Hastie, Narasimhan and Chu's study (2002) the optimal shrinkage threshold value was chosen based on the average errors of a 10 fold CV resulting in a set of 43 genes that was associated with the minimum CV errors.

With the empirical approaches (Klassen & Kim, 2009; Levner, 2005; Ray *et al.*, 2007), the optimal shrinkage threshold was selected based on the lowest classification error over a range of shrinkage thresholds. For example, in Levner's study (2005), experiments were first carried out with 20 different shrinkage threshold values in the range of [0.5, 10] with increments of 0.5. This study also experimented with another 200 different shrinkage threshold values in the range of [0.5, 10] with increments of 0.05, and obtained the same classification results. In general, CV and empirical approaches for determining the optimal shrinkage threshold value are based on "trial and error". However, such shrinkage threshold values may not be precisely tuned for the specific dataset for obtaining optimal classification results. This is due to the fact that it is limited in terms of exploring the search space of shrinkage threshold values in relation to the dataset. It is vital to address the issues described above. Thus, a new approach incorporating GA for automatically searching for the optimal shrinkage threshold for the NSC is proposed in this study.

Besides investigating evolutionary approaches for obtaining the shrinkage threshold values, similarity measures used in NSC is another area of investigation. The investigation is structured in the following way: the investigation of evolutionary approaches for the NSC and followed by investigation of the impact of different similarity measures. Chapter 5 and 6 described investigations involving GA and Memetic algorithm and followed by the description of investigations if similarity measures in Chapter 7.

The following section describes the proposed approach (NSC-GA) involving GA. Section 5.3 describes the parameter settings for GA. Using seven datasets described in Section 3.1, the performance of the proposed approach is examined using the evaluation strategy as described in Section 3.2. The evaluation results are reported in Section 5.4 and the summary is in Section 5.5.

5.2. The proposed approach, NSC-GA

Figure 5-2 illustrates the framework of the proposed approach, NSC-GA that incorporates NSC and the GA to search for the best optimal range of shrinkage thresholds for the NSC algorithm. The basic concepts of NSC (Tibshirani et al., 2002) and GA (Goldberg, 1989) algorithms have already been reviewed in Section 2.3.3, and 2.3.4.1, respectively.

The two main steps are:

Step 1: This step carries out the procedure of automatic calculation of Th_{max}. This procedure is performed once only at the beginning of the proposed approach, NSC-GA, to obtain $Th_{max.}$

Step 2: The GA is employed in this step as an optimization method to search for optimal sets of shrinkage thresholds for NSC algorithm that lead to the selection of optimal subsets of features. Also in this step, the NSC algorithm is employed as a fitness evaluator to evaluate the fitness of each chromosome in terms of the number of selected features and its training classification accuracy.

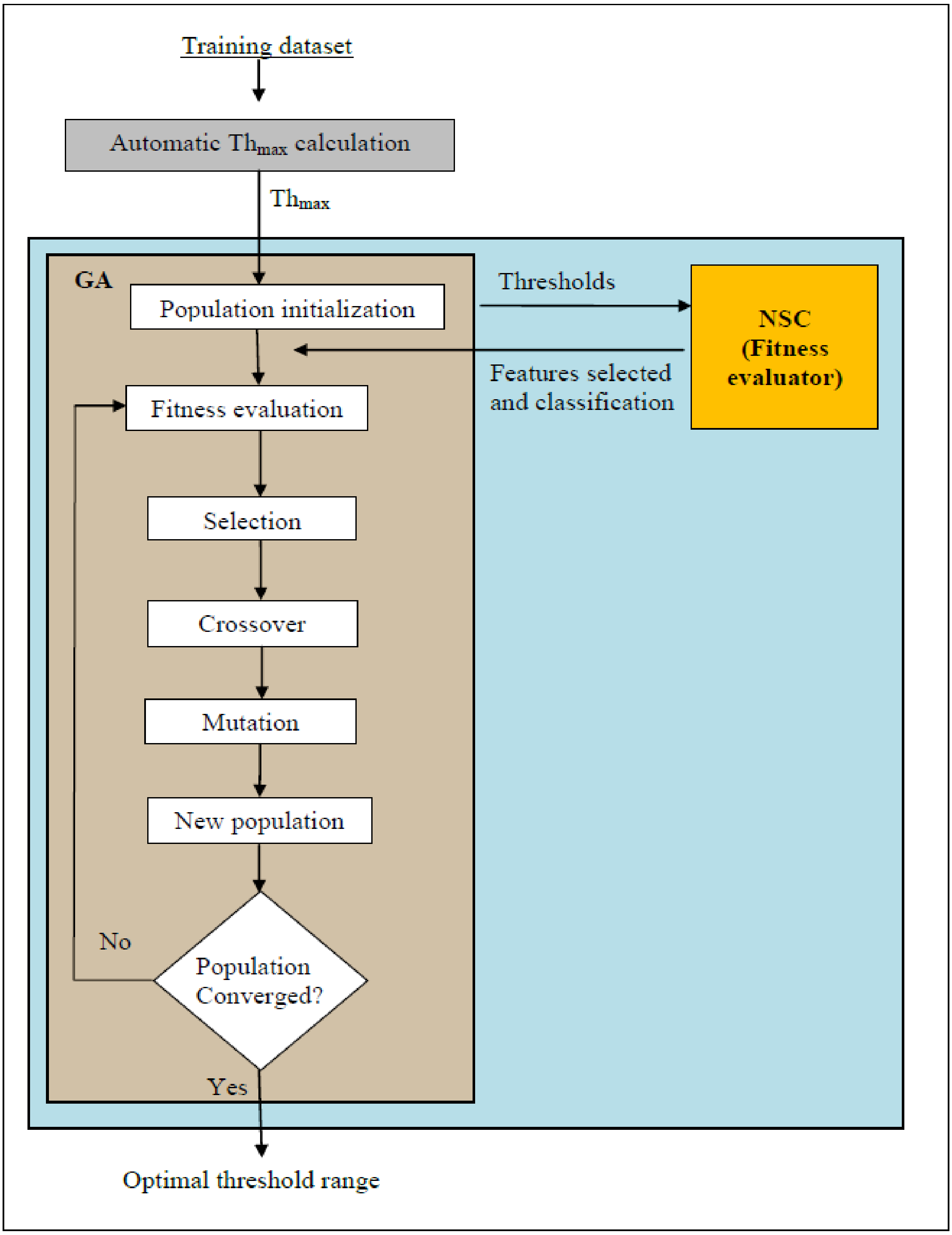


Figure 5-2 Framework of the proposed approach, NSC-GA

5.2.1. Issues related to the proposed approach, NSC-GA

Encoding chromosomes, estimating the initial range of values for the shrinkage threshold and fitness evaluation are the issues that need to be first addressed in NSC-GA. The following section describes these issues.

5.2.1.1. Encoding chromosomes

The aim of the proposed approach is to optimize a range of shrinkage threshold values consisting of real numbers for NSC. The most appropriate encoding representation for chromosomes in this study would be real-encoding. Each chromosome, consists of number of genes, representing a range of *n* shrinkage threshold values, e.g. {1.23 0.56 4.23 5.32 6.0 0.87 in the case of $n = 6$}. This allows the optimization of a range of shrinkage threshold values and the use of the GA crossover operator. Without using crossover to recombine chromosomes, GA would rely solely on a mutation operator and has a higher probability of being stuck in a local optimum (Back *et al.*, 1991).

5.2.1.2. Estimate initial range of values for shrinkage thresholds

Shrinkage threshold values of chromosomes are generated randomly using a RNG. Theoretically, shrinkage thresholds can be in the range [0, ∞]. However, in practice, there is a finite number of attributes associated with the dataset to be analysed. The lower limit ($\mathrm{Th}_{\mathrm{lower}}$) associated with shrinkage thresholds is a value where all attributes from the dataset are selected and the maximum value ($\mathrm{Th}_{\mathrm{max}}$) is a value where only 1 attribute is selected. The value $\mathrm{Th}_{\mathrm{lower}}$ is 0. Thus shrinkage threshold values in the range [0, $\mathrm{Th}_{\mathrm{max}}$] map to the search space of sets of features in NSC. In the proposed approach, a chromosome is a range of shrinkage thresholds, each shrinkage threshold maps to a subset of features, therefore each chromosome maps to a number of subsets of features. This mapping is different from the commonly used binary representation in FS in which, a chromosome is a string of binary (bit) of 0 and 1, and each gene (bit) value maps to 1 feature, with each chromosome mapping to only 1 set of selected features.

To illustrate the impact of $\mathrm{Th}_{\mathrm{max}}$ in the "time-to-convergence" in NSC-GA, Figure 5-3 and Figure 5-4, respectively, showed examples of convergence plots of fitness from executing NSC-GA with and without employing Th_{max} when analysing the AD training data.

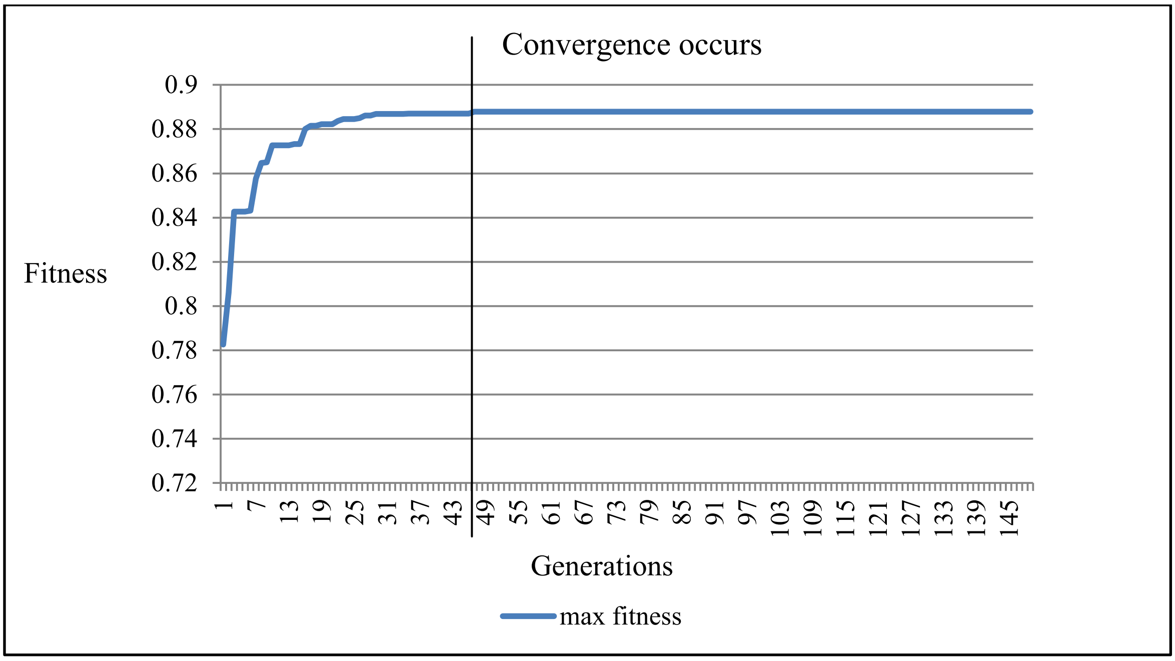


Figure 5-3 Example of convergence plot for AD training dataset with the application of Th_{max} calculation using NSC-GA

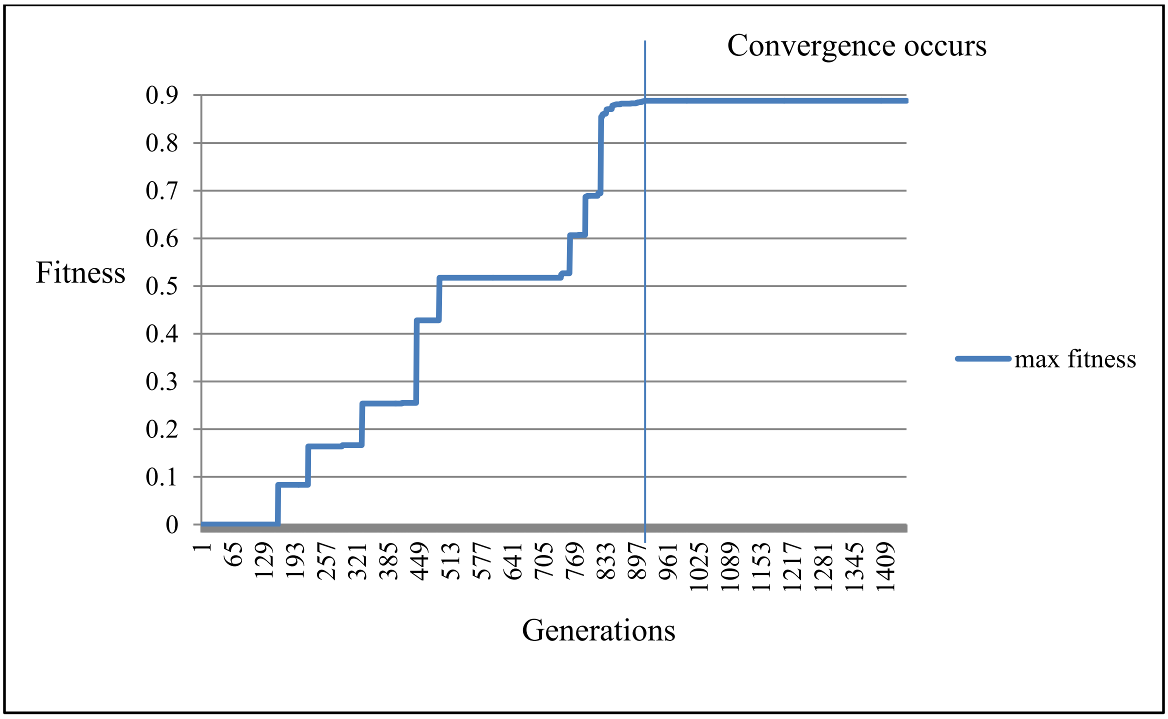


Figure 5-4 Example of convergence plot for AD training dataset without the application of Th_{max} calculation and $Th_{upper} > Th_{max}$

As seen in Figure 5-3 and Figure 5-4, the algorithm achieved the same maximum fitness of 0. 887. However, it can be also seen that, with the application of the Th_{max}, the algorithm reached convergence much quicker, at the 47th generation in comparison to the 916th generation for the algorithm that did not use Th_{max}. In this instance, the approach that did not use Th_{max} required more than 20 times the number of generations compared to the one using Th_{max}. The approach with the application of Th_{max} reached the fitness of 0.78 in the 1st generation, while the other approach (without Th_{max} calculation) required 820 generations before reaching the same fitness value of 0.78. The algorithm without the application of Th_{max} spent much more computational time to obtain the same result as that of the one with Th_{max} calculation. This is not only unnecessary but also contradictory to attempts by many previous researchers whom have tried to develop algorithms or strategies to improve computational time for GA (Li and Love (1997), Ahujaa and Orlinb (2000), Ilonen, Kamarainen and Lampinen (2003), and Snyder and Daskin (2006).

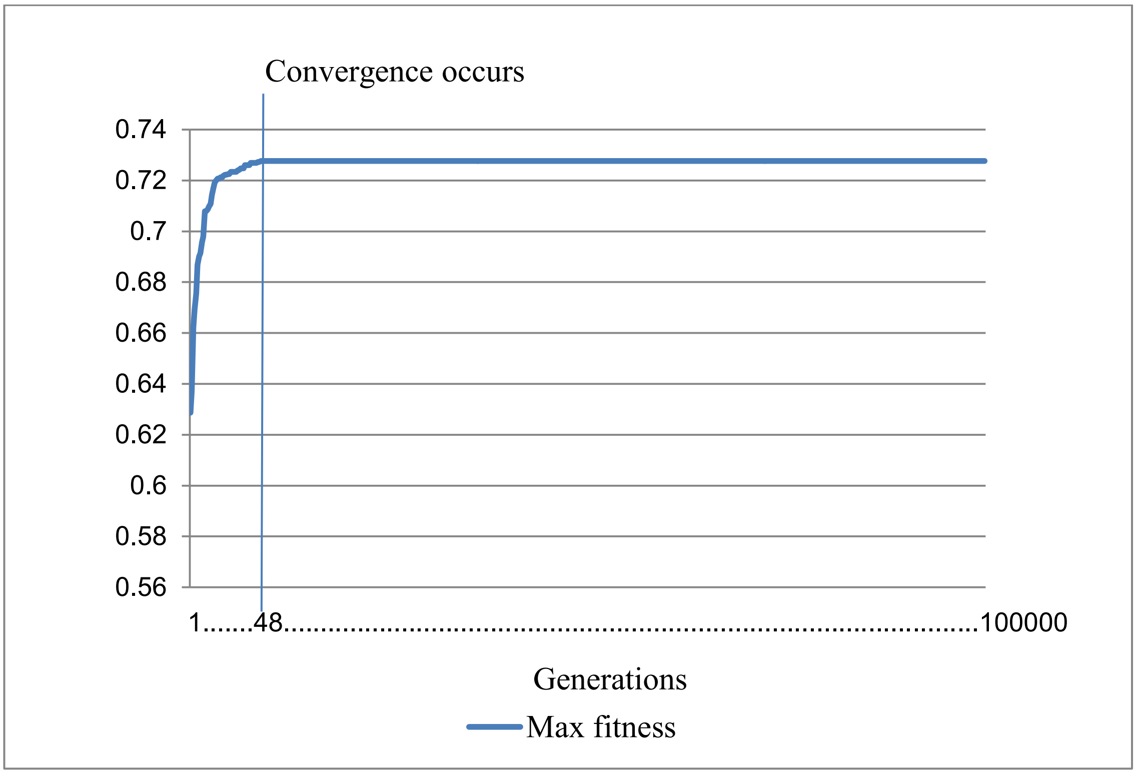


Figure 5-5 Example of convergence plot for AD training dataset without the application of Th_{max} calculation and the value of $Th_{upper} < Th_{max}$

Figure 5-5 shows an example of the convergence plot of fitness for analysing the impact of Th_{max}, using the same Ray *et al.* training dataset as before and where the upper limit of shrinkage threshold value is less than the associated of Th_{max} value (i.e. the initial population of chromosomes is initialized to be in the range [0, th_{upper}] (i.e. $th_{upper} < Th_{max}$). As seen in this figure, the algorithm is stuck in a local optima and premature convergence occurred. A maximum fitness of 0.727 is obtained after running for a very large number of generations (100,000). The optimal shrinkage threshold obtained has resulted in a set of 48 features with corresponding classification accuracy of 86.95% on the unseen test set. In comparison, as demonstrated in Figure 5-3, a fitness of 0.887 resulting in a set of 11 features with corresponding classification accuracy of 89.49% on unseen test data in the case involving the use of Th_{max}.

Th_{max} is a simple procedure that involved a single iteration to calculate Th_{max} and the computational time to obtain Th_{max} is up to around 1 second for each of seven datasets using a personal computer i7, CPU speed of 3.4 GHz with 16 GB memory, Windows 7 and NetBeans 7.2. The calculation of Th_{max} needed to be carried out once only for each dataset. For example, it took 0.0105 seconds to obtain Th_{max} for the Ray *et al.* AD training dataset, 0.167 seconds for the Colon cancer dataset, 0.41 second for Leukemia, 0.180 second for Lymphoma, 0.95 second for Lung cancer, 0.971 second for Prostate cancer and 1.06 seconds for Ovarian cancer dataset. It can also be seen that the application of Th_{max} in the proposed GA based approach maximizes the performance of the algorithm, resulting in a global convergence using less computational time.

5.2.1.3. Fitness evaluation using NSC as a fitness evaluator

The NSC algorithm as described in Section 2.3.3 was implemented and employed as a fitness evaluator in this approach to evaluate the fitness of the chromosomes using the training dataset.

5.2.2. Steps in the proposed approach, NSC-GA

The following sections describe the steps in NSC-GA.

5.2.2.1. Step 1: Th_{max} calculation

To find the value of Th_{max} for the dataset in question, the approach estimates the value of Th_{max} using the procedure shown in Figure 5-6.

```
Input
        Training dataset ($Ts$)
Output
        $\text{Th}_{\text{max}}$ value
Steps
   1. Generate a real random number ($Rn$) in the range [0,1] as an initial shrinkage
        threshold seed using RNG
   2. Set $\text{Th}_{\text{max}} = Rn$
   3. Perform NSC FS on Ts using $\text{Th}_{\text{max}}$ to select a number of features ($N$)
   4. Loop while $n \neq 1$
        a. If no feature selected $n = 0$
              • Generate $Rn$
              • Decrease $\text{Th}_{\text{max}}$ by $Rn$, $\text{Th}_{\text{max}} = \text{Th}_{\text{max}} - Rn$
              • Perform NSC FS on Ts using updated $\text{Th}_{\text{max}}$ to select $n$
        b. Else
              • Generate $Rn$
              • Increase $\text{Th}_{\text{max}}$ by $Rn$, $\text{Th}_{\text{max}} = \text{Th}_{\text{max}} + Rn$
              • Perform NSC FS on Ts using updated $\text{Th}_{\text{max}}$ to select $N$
   5. Return $\text{Th}_{\text{max}}$
```

Figure 5-6 Algorithm for calculating Th_{max}

As seen in the algorithm in Figure 5-6, the value of Th_{max} is adjusted up or down using steps of values associated with random numbers in the range [0, 1]. This process repeats until Th_{max} reaches the value that results in only one feature being selected using NSC, i.e., Th_{max} of the training dataset has been determined.

5.2.2.2. Step 2: GA search optimization

The following section describes steps involving the application of GA in NSC-GA.

a) Population initialization

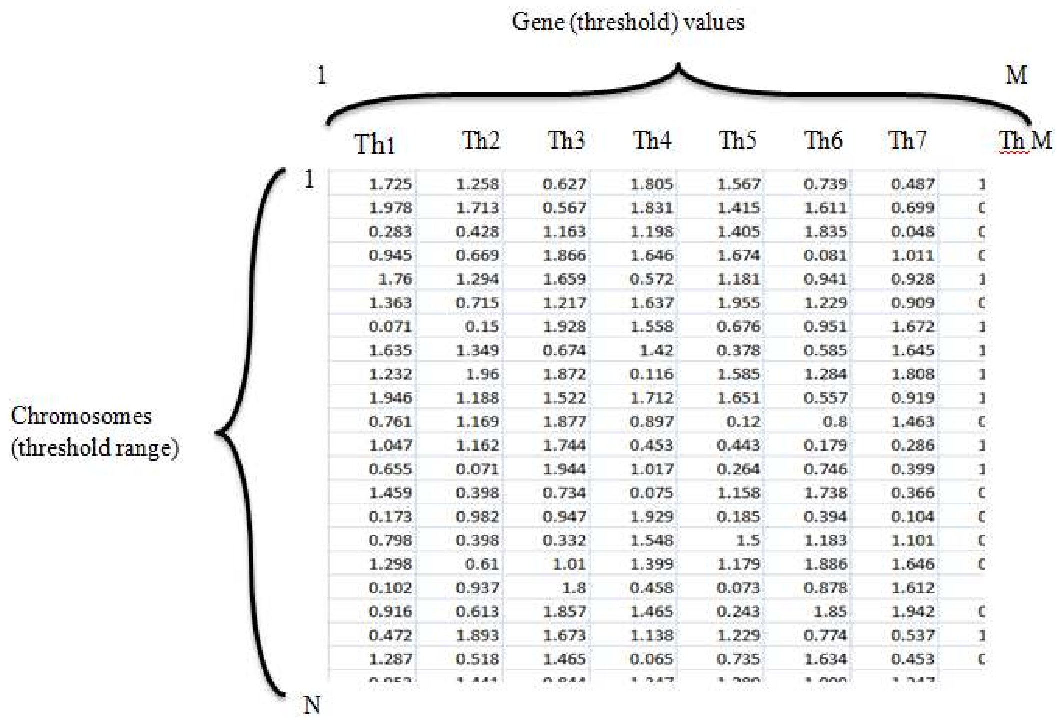
After Th_{max} has been calculated, a population of chromosomes is then initialized. Each shrinkage threshold in a chromosome (essentially each chromosome represents a range of values for shrinkage thresholds) is initialized to a real number, generated randomly in the range of [0, Th_{max}] using a RNG. The number of shrinkage threshold values in the chromosome equals to the length (*n*) of the chromosome, i.e. the number of genes in the chromosome. Theoretically, *n* can be as large as ∞, but in this proposed approach, *n* = 10 is chosen. That is, each chromosome consists of 10 shrinkage thresholds. The size of 10 is chosen empirically to balance the computational time and obtaining the optimal shrinkage threshold. For example, a chromosome of size 10 is illustrated as a range of 10 real numbers, as follows.

2.312	3.523	1.133	1.034	2.334	9.234	0.211	5.354	8.142	10.299

Figure 5-7 describes the algorithm used to initialize the population and Figure 5-8 shows an example of an initial population.

```
Input:
        $Th_{max}$
        Length of chromosome, $n$
        Size of population, $p$
Output:
        An initialized population of $p$ rows and $n$ columns
Steps:
    1. Set population ($I_p$) as a 2 dimensional array of $p$ rows and $n$ columns of real
       numbers
    2. Set $I_p = \{\emptyset\}$
    3. For counter1 from 1 to $p$
        3.1 For counter2 from 1 to n
            a. Generate a real random number ($R_n$) in the range [1, $Th_{max}$] using a
               RNG
            b. Store $R_n$ to $I_p$[counter1][counter2]
```

Figure 5-7 Initial population algorithm using RNG


Gene (threshold) values
1
M
Chromosomes
(threshold range)
N

	Th1	Th2	Th3	Th4	Th5	Th6	Th7	Th M
1	1.725	1.258	0.627	1.805	1.567	0.739	0.487	1
	1.978	1.713	0.567	1.831	1.415	1.611	0.699	0
	0.283	0.428	1.163	1.198	1.405	1.835	0.048	0
	0.945	0.669	1.866	1.646	1.674	0.081	1.011	0
	1.76	1.294	1.659	0.572	1.181	0.941	0.928	1
	1.363	0.715	1.217	1.637	1.955	1.229	0.909	0
	0.071	0.15	1.928	1.558	0.676	0.951	1.672	1
	1.635	1.349	0.674	1.42	0.378	0.585	1.645	1
	1.232	1.96	1.872	0.116	1.585	1.284	1.808	1
	1.946	1.188	1.522	1.712	1.651	0.557	0.919	1
	0.761	1.169	1.877	0.897	0.12	0.8	1.463	0
	1.047	1.162	1.744	0.453	0.443	0.179	0.286	1
	0.655	0.071	1.944	1.017	0.264	0.746	0.399	1
	1.459	0.398	0.734	0.075	1.158	1.738	0.366	0
	0.173	0.982	0.947	1.929	0.185	0.394	0.104	0
	0.798	0.398	0.332	1.548	1.5	1.183	1.101	0
	1.298	0.61	1.01	1.399	1.179	1.886	1.646	0
	0.102	0.937	1.8	0.458	0.073	0.878	1.612	
	0.916	0.613	1.857	1.465	0.243	1.85	1.942	0
	0.472	1.893	1.673	1.138	1.229	0.774	0.537	1
	1.287	0.518	1.465	0.065	0.735	1.634	0.453	0

Figure 5-8 An example of an initial population

Figure 5-8 shows an example of an initial population of N chromosomes by M shrinkage threshold values. That is, each chromosome consists of M shrinkage thresholds. After the population has been initialized, the next step is to employ the NSC algorithm as a fitness evaluator to evaluate the fitness of each chromosome.

b) Fitness evaluation

GA was described in Section 2.4.4 and implemented, and employed in this study. The GA here uses an objective function to optimize the search for finding optimal shrinkage thresholds that leads to the selection of the smallest set of features with the highest classification accuracy. The objective function, *f*, is an aggregation of two fitness functions, f_1 and f_2, calculated using Equations (5.1), (5.2) and (5.3).

$$f = f_1 + f_2 \tag{5.1}$$

$$f_1 = (N_{total} - N_{att}) / N_{total} \tag{5.2}$$

$$f_2 = \frac{TP+TN}{TP+FP+TN+FN} \qquad (5.3)$$

where N_{total} is the total number of attributes (features) of the dataset

N_{att} is the number of attributes selected by NSC

f_2 is an overall training classification accuracy for the selected set of attributes using NSC.

TP is true positive for correct prediction to the disease class

TN is true negative for correct prediction to the normal class

FP is false positive for incorrect prediction to disease class

FN is false negative for incorrect prediction to normal class

f_1 is computed based on the number of attributes selected over the total of number of attributes in the training dataset. That is, the smaller the set of features, the higher the fitness value for f_1. Thus f_1 is designed for evaluating the fitness of a shrinkage threshold that leads to a minimum number of attributes.

f_2 is computed based on the classification accuracy, associated with the training data, in the form of *TP* and *TN* over a total number of samples in the training dataset (i.e. *TP*, *FP*, *TN* and *FN*). Thus f_2 is designed for evaluating the fitness of a shrinkage threshold that leads to the maximum classification accuracy.

Since each chromosome (range) consists of a number of shrinkage threshold values, therefore the overall fitness of a chromosome is calculated, as the average of fitness values associated with each of the shrinkage thresholds in the chromosome, using Equation (5.4).

$$Fitness_{Ind} = \sum_{i=1}^{M} f_{th} / M \qquad (5.4)$$

where *M* is the number of shrinkage thresholds in a chromosome.

c) GA operators

Selection, crossover and mutation operators for real encodings are used in NSC-GA. The same tournament selection procedure employed in the RST-GA approach proposed in Chapter 4 is also employed here to select the parent chromosomes for crossover.

The same single point crossover employed in Chapter 4 is used here to recombine the 2 selected parent chromosomes to produce 2 offspring chromosomes using the probability of crossover (P_c) listed in Table 5-4.

Uniform mutation (Eiben & Smith, 2007) is used for real-encoded chromosomes. Uniform mutation has been described in Section 2.4.4.1 and is employed here to modify offspring chromosomes using the mutation rate (P_m) listed in Table 5-4. Uniform mutation modifies a chromosome by replacing its gene value with a mutated number, N_{mut} , which is calculated using Equation (5.5).

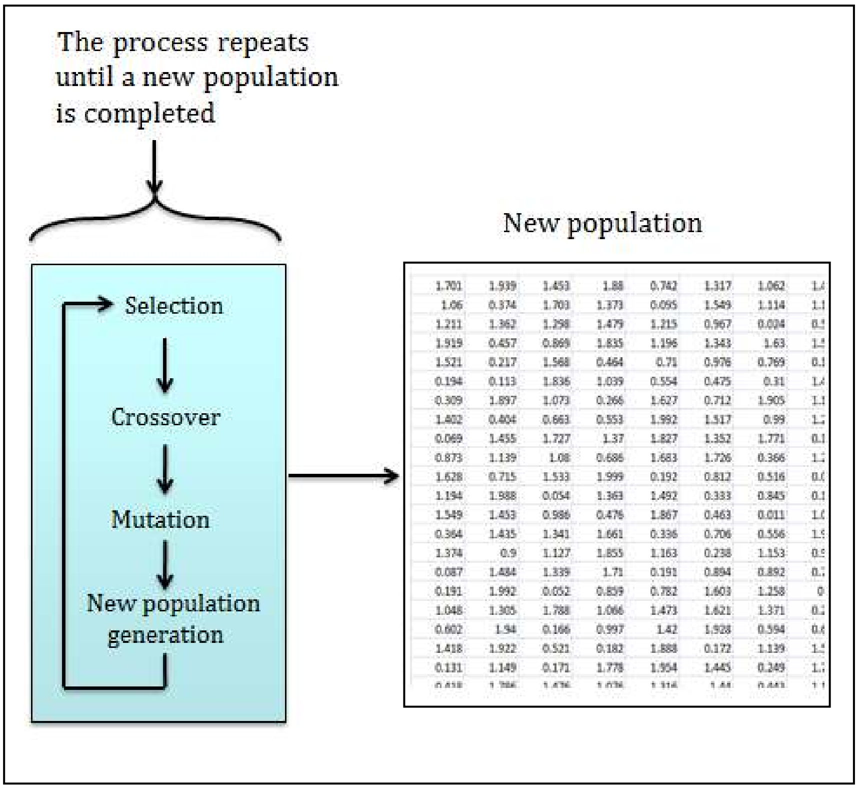
$$N_{mut} = L_b + (R_n * (U_b - L_b)) \tag{5.5}$$

where L_b is lower bound of chromosome, R_n is a random number generated by RNG, U_b is upper bound of chromosome.

d) Generation of New population

Two parents and 2 offspring chromosomes from the previous step involving selection, crossover and mutation are evaluated for their fitness and the best 2 chromosomes are selected and placed into the new population. A single elitist strategy is also employed to allow the best candidate solution in the previous generation to be retained and placed into the new generation to improve the search in evolutionary algorithms (Ahn & Ramakrishna, 2010). The process of selection, crossover and mutation iterates until the generation of the new population is completed. The procedure for generating a new population in NSC-GA is the same as the one in RST-GA (Figure 4-8), except that uniform mutation instead of the bit flip mutation is employed here.

The following figure shows an example of the process of selection, crossover, mutation, in new population generation.


The process repeats until a new population is completed
Selection
Crossover
Mutation
New population generation
New population

1.701	1.939	1.453	1.88	0.742	1.317	1.062
1.06	0.374	1.703	1.373	0.095	1.549	1.114
1.211	1.362	1.298	1.479	1.215	0.967	0.024
1.919	0.457	0.869	1.835	1.196	1.343	1.63
1.521	0.217	1.568	0.464	0.71	0.976	0.769
0.194	0.113	1.836	1.039	0.554	0.475	0.31
0.309	1.897	1.073	0.266	1.627	0.712	1.905
1.402	0.404	0.663	0.553	1.992	1.517	0.99
0.069	1.455	1.727	1.37	1.827	1.352	1.771
0.873	1.139	1.08	0.686	1.683	1.726	0.366
1.628	0.715	1.533	1.999	0.192	0.812	0.516
1.194	1.988	0.054	1.363	1.492	0.333	0.845
1.549	1.453	0.986	0.476	1.867	0.463	0.011
0.364	1.435	1.341	1.661	0.336	0.706	0.556
1.374	0.9	1.127	1.855	1.163	0.238	1.153
0.087	1.484	1.339	1.71	0.191	0.894	0.892
0.191	1.992	0.052	0.859	0.782	1.603	1.258
1.048	1.305	1.788	1.066	1.473	1.621	1.371
0.602	1.94	0.166	0.997	1.42	1.928	0.594
1.418	1.922	0.521	0.182	1.888	0.172	1.139
0.131	1.149	0.171	1.778	1.954	1.445	0.249

Figure 5-9 An example of creating a new population

e) Checking for convergence in NSC-GA

The process of fitness evaluation, selection, crossover, mutation and new population generation is repeated until the convergence of fitness takes place or if a predefined maximum number of generations have been executed. The procedure of verifying the convergence status and terminating the GA have been described in Section 3.3 and 3.4, respectively. Upon convergence, the fittest chromosome (optimal solution) is selected.

The process of determining parameter settings used in NSC-GA is described in Section 5.3.

5.3. Parameter settings for GA

As discussed previously, one of the aims of this study is to investigate the incorporation of evolutionary algorithms in developing techniques for analysing high dimensional biological datasets. An aspect associated with the use of evolutionary algorithms involves finding appropriate values for its parameters (Eiben & Smith, 2003). Population size, crossover and mutation probability rate are some of the crucial parameters which affect the performance of evolutionary algorithms where their specific values may cause the algorithm either to converge to a local (premature convergence) or global optimal solution. This process is known as parameter tuning, in which appropriate parameter settings are determined to ensure that the algorithm will perform at its best (M. Srinivas & Patnaik, 1994).

A traditional parameter tuning approach empirically finds a set of parameter values which is then subsequently applied to the evolutionary algorithm for processing the various problem instances. This is usually a very time-consuming and hard task (Nannen, Smit, & Eiben, 2008) as there are many choices of values associated with the parameters and little is known about the effect of these parameter values on the performance of the algorithm. Often this process is guided by conventions (e.g. low mutation rate), ad hoc choices and experimental comparisons carried out on a limited scale. Dovgan, Tušar and Filipic (2011) carried out experiments, comparing parameter tuning methods for evolutionary algorithms and their findings showed that "*there is no best value for each parameter, but there are wide ranges of good parameter values*" (Franken *et al*., 2011, p. 2), and that there's some value in conducting parameter tuning. Recently a study by Fraser and Arcuri (2011) confirmed that parameter tuning can have an impact on the performance of the evolutionary algorithm but if this is not performed properly, it is highly likely to result in obtaining parameter configurations which are worse than values already found in the literature. One of the main conclusions from their study is that "*using default values coming from the literature is a viable option*" (Fraser & Arcuri, 2011, p. 26), specifically in the case where parameter tuning is expensive and the investigation is focused on examining the performance of new techniques (rather than comparisons between techniques). They argued that it would make more sense to use the available time for analysing a larger number of case studies than to spend that time on parameter tuning.

Section 5.3.1 describes the procedure for selection of the parameter settings which will be employed in approaches in the remaining part of this thesis. The outcomes of this process are presented in Section 5.3.2 with a summary of the results listed in Table 5-3.

5.3.1. Parameter settings

In order to find an appropriate set of values for the GA parameters, empirical parameter tuning involving 4 sets of commonly used parameter settings from the literature are conducted. The parameters that are tuned (highlighted in blue in Table 5-2) include population size, crossover probability rate (P_c), and mutation probability rate (P_m), with each set of these values being taken from DeJong (1975), Grefenstette (1986), Goldberg (1989), and Alander (1992) respectively (Table 5-2). These sets of parameter settings have been widely used in applications involving the GA and have been considered as "standard parameter values" for the GA (Harik & Lobo, 1999). The aim of the parameter tuning process in this study is to determine which of these four sets of "standard parameter values", will encourage more exploration and less exploitation in the population, and achieves an appropriate balance between selection pressure and diversity so that global convergence can be achieved in a reasonable amount of time.

In other words, during parameter tuning, the evaluation is on the suitability of each the sets of parameter values for one algorithm using specific datasets and the end product is a specific set of parameter values. Suitability is measured in terms of achieving maximum fitness at convergence and the required computation time. For example, we can have the situation where there are two sets where the algorithm can converge to the same maximum fitness but one set provides a lower selection pressure than the other and so may take a longer computation time to reach convergence. Obviously in this case, the set of parameter values that allowed the algorithm to converge to the same maximum fitness and using a shorter computation time will subsequently be chosen in the application of the algorithm to solve the specific problem. Thus, computation time in obtaining maximum fitness at convergence is used to evaluate one set of parameter values against another set.

The tuning process involved separate trials that employ each of the 4 sets of parameter settings in the NSC-GA and for all the seven datasets described in Section 3.1. The results obtained from these experiments are analyzed and the set of parameter values which gave the best results in terms of computational time for convergence and maximum fitness were then selected to be used in the proposed approaches in this study.

An entire tuning trial involving each of the 4 parameter sets using each of the 7 datasets (i.e. 4 x (1 x7) = 28 runs) takes 119,339 minutes to complete. Owing to this lengthy computational time, three independent trials of the tuning process are carried out. The following table, Table 5.1, shows the computational time of using a personal computer i7, CPU speed of 3.4 GHz with16 GB memory, Windows 7 and NetBeans 7.2 for one trial involving each parameter setting for each dataset.

Table 5-1 Computational time spent for one trial involving the 4 parameter settings

Datasets	Running time (minutes)			
	Dejong (1975)	Grefensette (1986)	Goldberg (1989)	Alander (1992)
Ray *et al.* AD	102	45	45	64
Alon *et al.* Colon	2241	644	724	795
Leukemia	2844	2314	1969	4895
Lung	11288	7721	6764	10816
Lymphoma	3377	1456	1086	1625
Ovarian	11180	7721	8596	10026
Prostate	5473	5352	3436	6740

Table 5-2 Four sets of parameter settings used in the tuning process

	Dejong	Grefensette	Goldberg	Alander
Population size	50	30	30	50
P_c	0.06	0.9	0.6	0.5
P_m	0.001	0.01	0.033	0.002
Maximum generations	5000			
Selection	Tournament			
Crossover	Single point			
Mutation	Uniform			
Elitist	One			

5.3.2. Results of parameter tuning

The following section presents the results from one typical independent trial. For each of the seven datasets, NSC-GA is applied using each of the 4 sets of parameter settings shown in Table 5-2 for one typical trial. Plots demonstrating convergence of fitness associated with each of the 4 sets of parameter settings as well as a typical plot of the state of convergence associated with one set of parameter settings is shown for each dataset. A summary of the results for the 3 independent trials is also shown in Table 5-3.

5.3.2.1. Ray *et al.* AD data

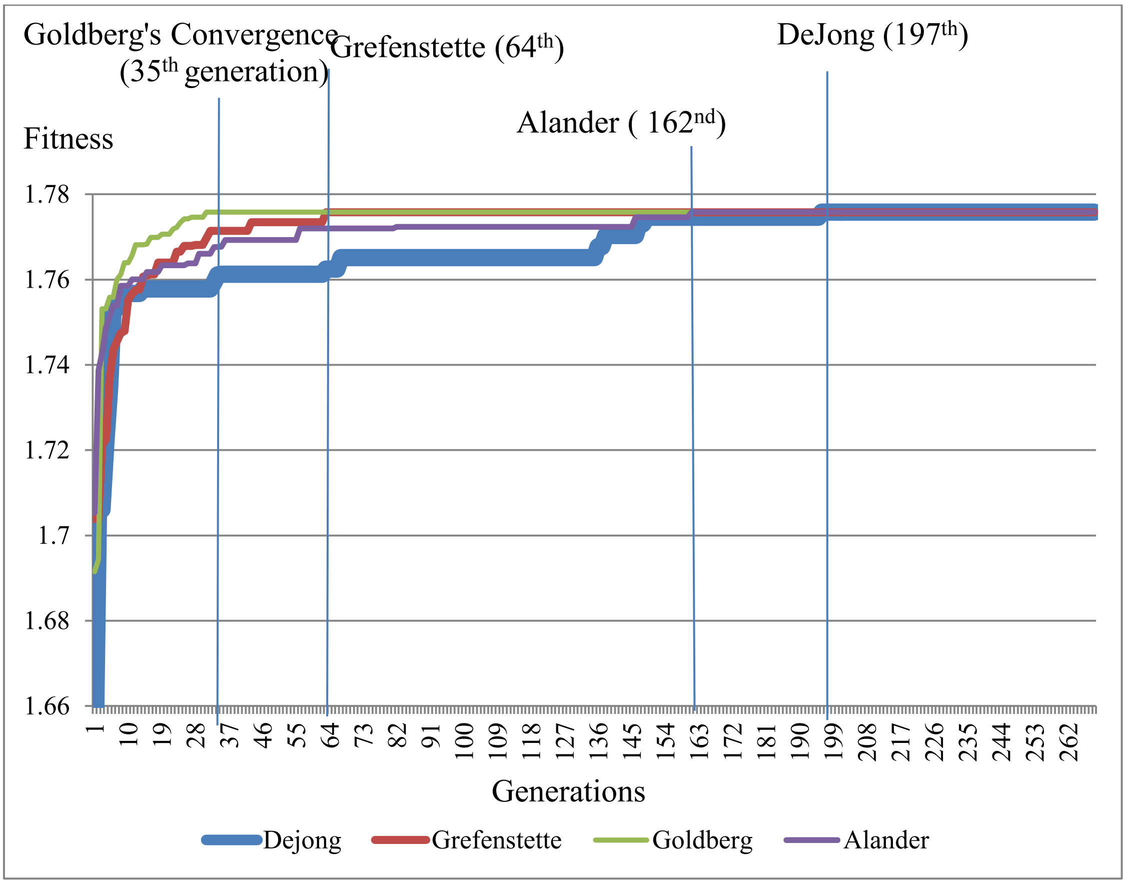


Figure 5-10 Typical convergence of fitness plots for AD data associated with each of the 4 different parameter settings from DeJong, Genfenstette, Goldberg and Alander

Figure 5-10 shows the convergence of fitness plots from running NSC-GA on the AD dataset using each of the 4 parameter settings from Table 5-2. The algorithm converged to the maximum fitness of 1.775 for each of the 4 sets of parameter settings. However, the convergence associated with each of the 4 sets of parameter settings occurred at different generations. With DeJong's set of parameter settings, convergence occurred after 197 generations and with Grefensette's and Alander's set of parameter settings, the convergence occurred after 64 and 162 generations, respectively. Whilst with Goldberg's set of parameter settings, convergence occurred after 35 generations. With this dataset, the algorithm using Goldberg's parameter settings outperformed the other 3 sets of parameter settings in terms of obtaining the global optimum with the same maximum fitness value and a quicker convergence.

The state of convergence associated with each run of the algorithm using one of the 4 sets of parameter settings is monitored using Srinivas and Patnaik (1994)'s method, see Section 3.3. The red line plot in Figure 5-11 shows the difference between the maximum and average fitness of the population (i.e. $f_{max} - f_{avg}$) and demonstrates the state of convergence of the algorithm associated with using Goldberg's parameter settings. The value of $(f_{max} - f_{avg})$ is expected to be very small for a population that has converged to a global optimum than for a population with members spread over the entire search space. That is, a value closer to 0 would imply a convergence closer to the global optimum.

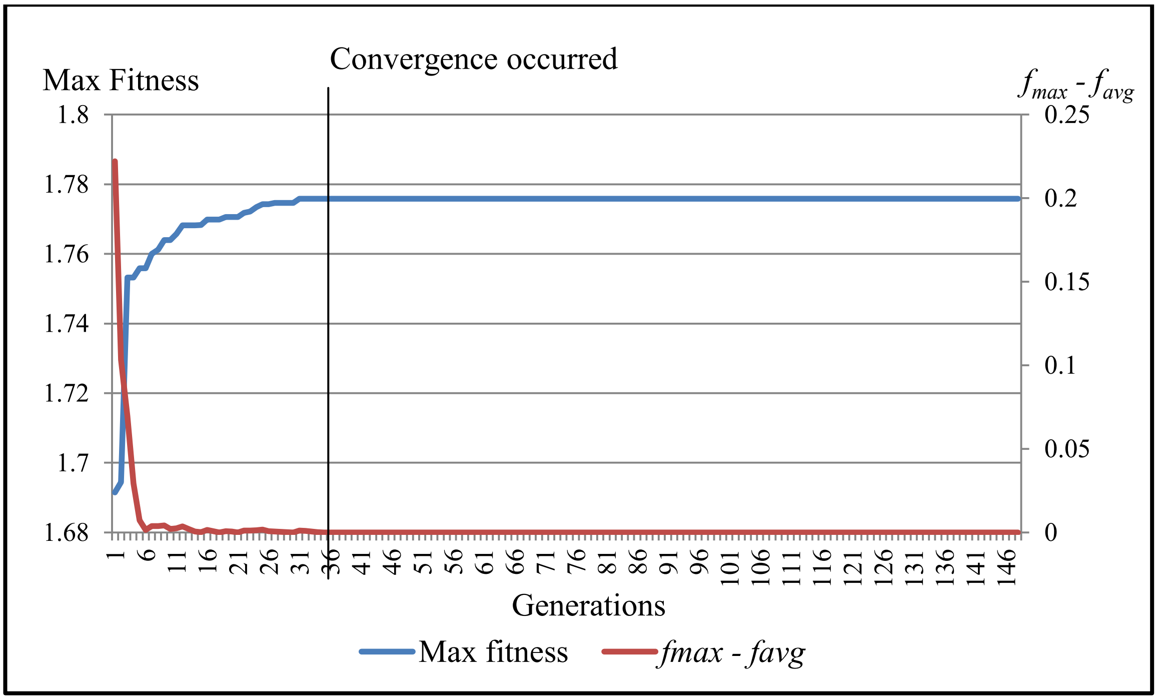


Figure 5-11 An example plot of the state of convergence: $(f_{max} - f_{avg})$ versus generations (using Goldberg's set of parameter settings)

As seen in Figure 5-11, the value for ($f_{max} - f_{avg}$) was relatively high (values shown on the right-hand vertical axis) in the earlier generations (<10) and it decreases to values very close to zero around 35th generations. This coincides with the maximum fitness of 1.775.

The following figure shows the state of convergence associated with one run of the algorithm for each of the 4 sets of parameter settings using Srinivas and Patnaik (1994)'s method.

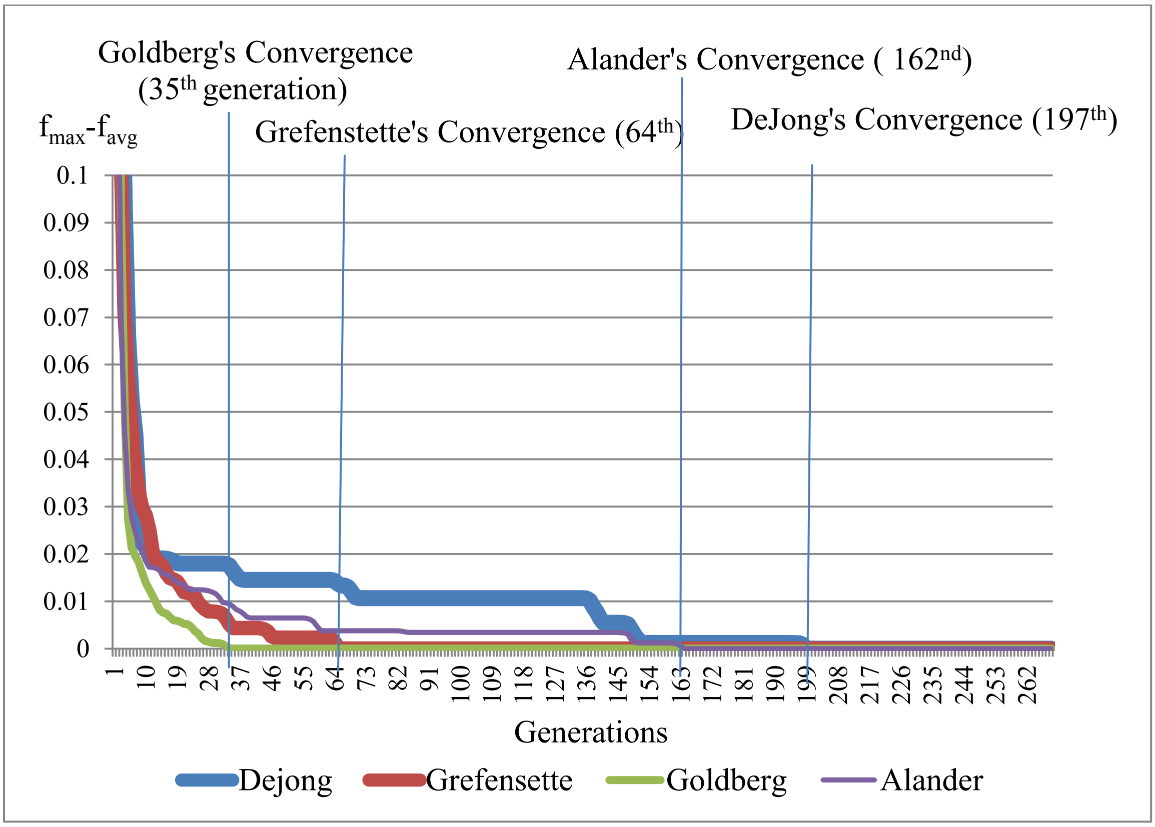


Figure 5-12 An example plot of the state of convergence: ($f_{max} - f_{avg}$) versus generations for 4 sets of parameter settings

As seen in Figure 5-12, the value for ($f_{max} - f_{avg}$) was relatively high for each set of parameter settings in the earlier generations (<10) and it decreases to 0 around 35 generations for Goldberg's parameter settings, 64 generations for Grefensette's, 162 generations for Alander's and 197 generations for Dejong's .

5.3.2.2. Alon *et al.* Colon cancer data

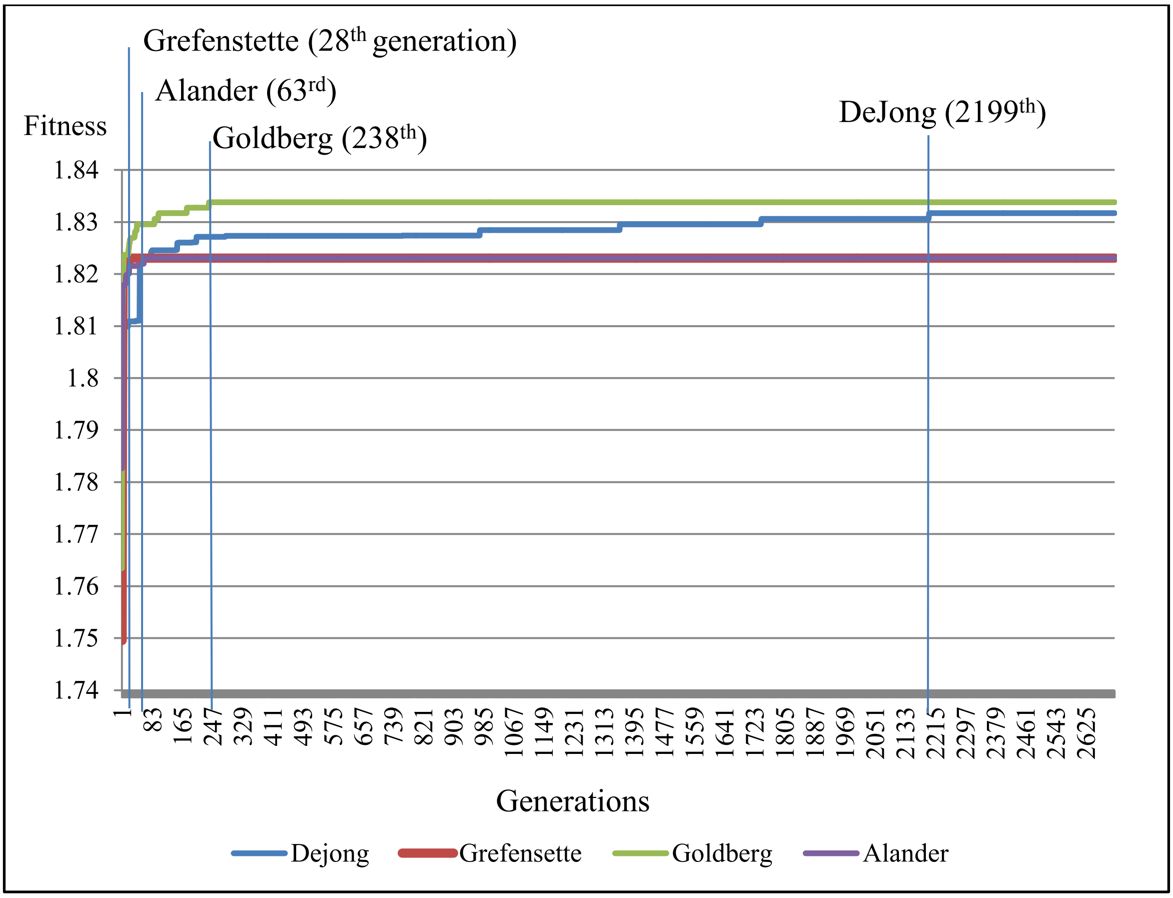


Figure 5-13 Typical convergence of fitness plots for Colon cancer data associated with each of the 4 different parameter settings from DeJong, Grefenstette, Goldberg and Alander.

Figure 5-13 shows the convergence of fitness plots of running NSC-GA on the Colon cancer dataset for each of the 4 sets of parameter settings. The algorithm converged with different maximum fitness values and involved a different number of generations for each of the 4 sets. With Goldberg's parameter settings, convergence occurred after 238 generations with the maximum fitness of 1.833, whilst with the other 3 sets of parameter settings, their maximum fitness at convergence is lower than Goldberg's value. In terms of the number of generations required for convergence, the algorithm took 2199 generations using DeJong's set of parameter settings. In the case of using Grefenstette's and Alander's set of parameter settings with the Colon cancer dataset, the algorithm required a smaller number of generations (in comparison to Goldberg's set) to converge. Therefore the algorithm with Goldberg's set of parameter settings outperformed the other 3 sets of parameter settings in terms of obtaining a higher maximum fitness.

5.3.2.3. Leukemia cancer data

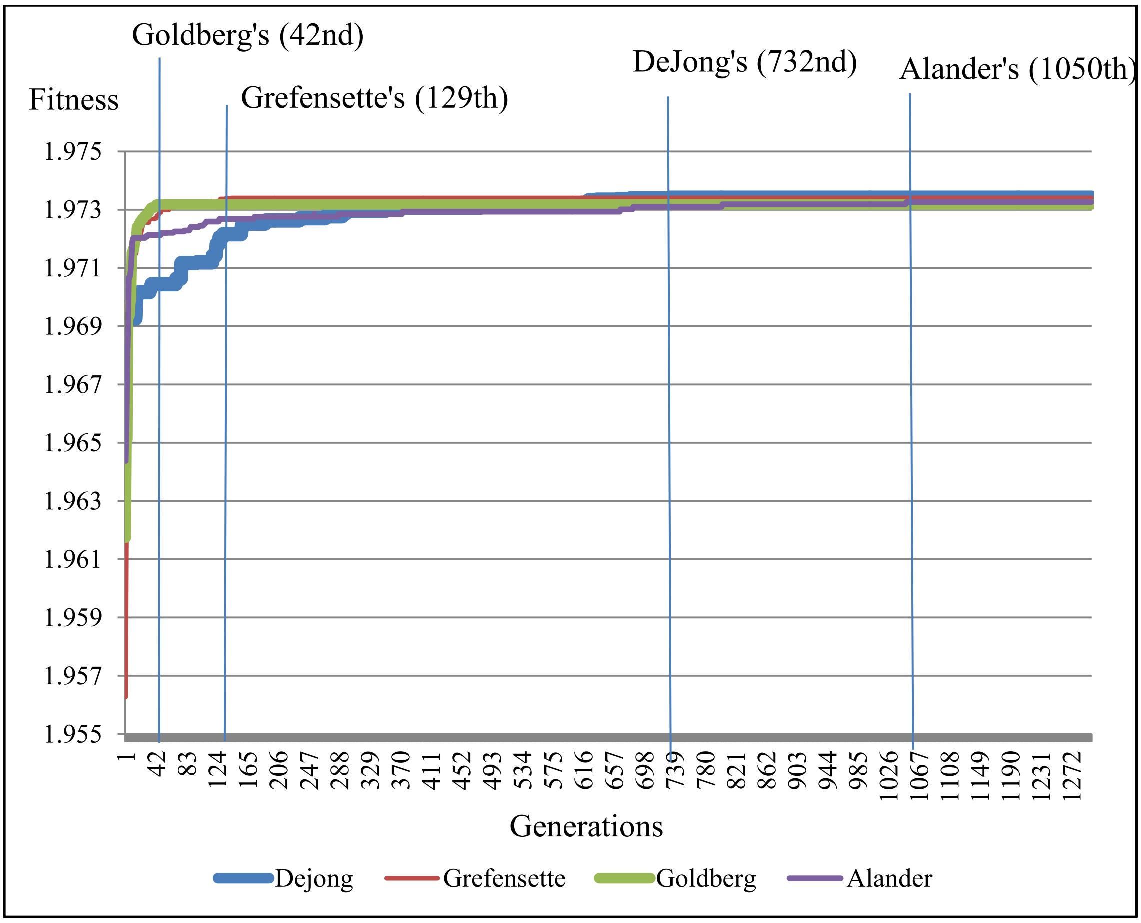


Figure 5-14 Typical convergence of fitness plots for Leukemia cancer data associated with parameter settings from DeJong, Grefenstette, Goldberg and Alander

Figure 5-14 shows the convergence of fitness plots of running NSC-GA on the Leukemia cancer dataset for each of the 4 sets of parameter settings. The algorithm converged to the global optimum with the same maximum fitness of 1.973 for each of the 4 sets of parameter settings. However, the number of generations that the algorithm has to run to achieve convergence is different. With Goldberg's set of parameter settings, convergence occurred after 42 generations, with Grefenstette's set of parameter settings, convergence occurred after 129 generations, with DeJong's set, convergence occurred after 732 generations and lastly with Alander's set, convergence occurred after 1050 generations. Therefore the algorithm with Goldberg's parameter settings outperformed the other parameter settings in terms of obtaining the global optimum with the same maximum fitness but used less computational time.

The state of convergence associated with each run of the algorithm using one of the 4 sets of parameter settings for the Leukemia cancer data is also monitored using the method proposed by Srinivas and Patnaik (1994). The plots obtained here for each of the 4 sets of parameters are similar in nature to that shown in Figure 5-11 and Figure 5-12.

5.3.2.4. Lung cancer data

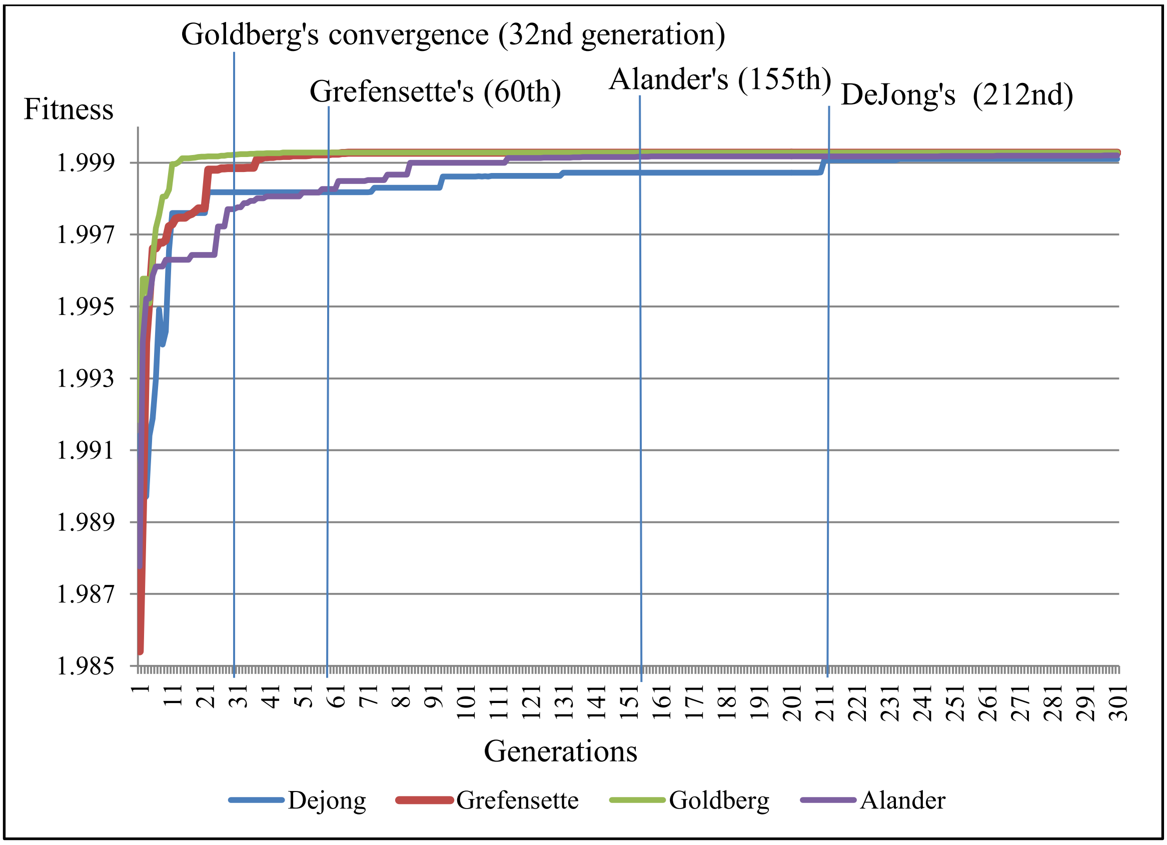


Figure 5-15 Typical convergence of fitness plots for Lung cancer data with 4 different parameter settings from DeJong, Grefenstette, Goldberg and Alander

The convergence of fitness plots from running NSC-GA on the Lung cancer dataset for each of the 4 sets of parameter settings is shown in Figure 5-15. It can be seen that the algorithm converged with the same maximum fitness of 1.999 for each of the 4 sets of parameter settings. However, these convergence started at different generations; with

Goldberg's set of parameter settings, the convergence occurred after 32 generations, with Grefenstette's set of parameters, the convergence occurred after 60 generations, DeJong's set parameters, the convergence occurred after 212 generations and Alander's set of parameters, the convergence occurred after 155 generations. With this dataset, the algorithm using Goldberg's set of parameter settings achieved the global optimum with the same maximum fitness.

The state of convergence associated with each run of the algorithm using one of the 4 sets of parameter settings for the Lung cancer data is again monitored. The plots obtained here for each of the 4 sets of parameters are similar in nature to that shown in Figure 5-11 and Figure 5-12.

5.3.2.5. Lymphoma cancer data

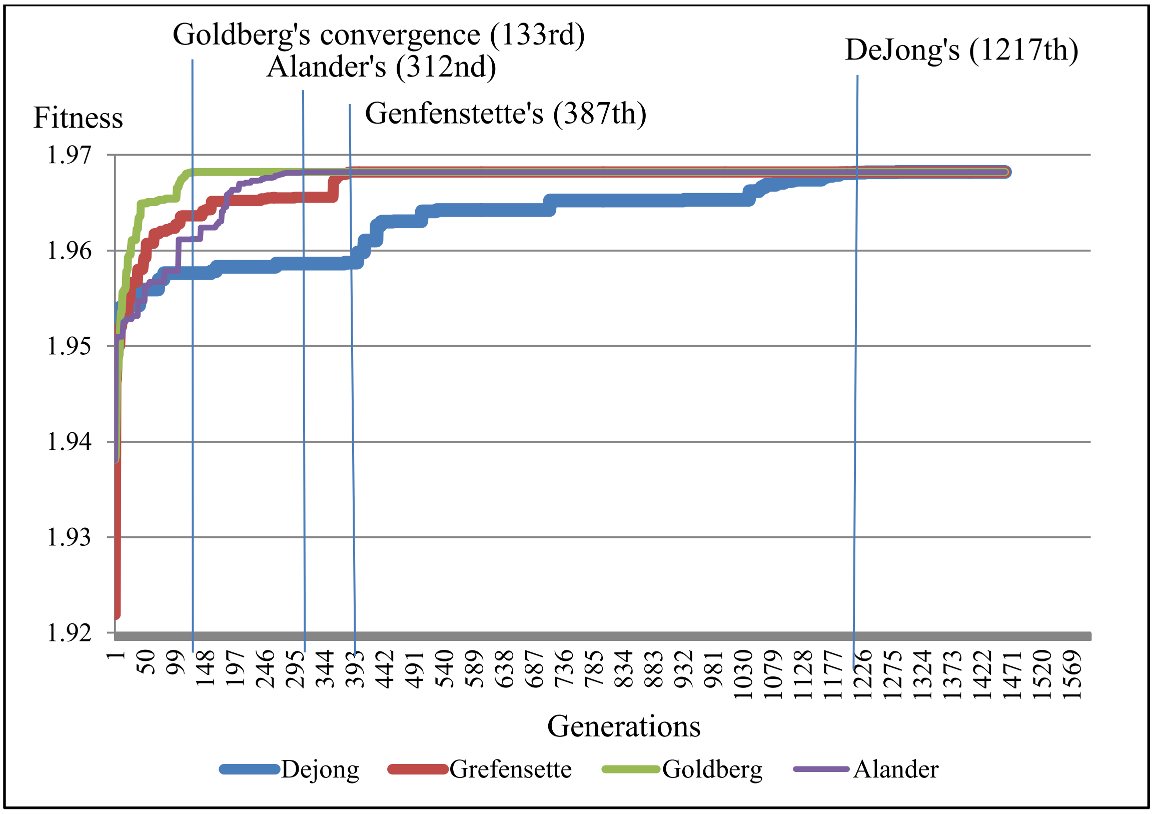


Figure 5-16 Typical convergence of fitness plots for Lymphoma cancer data for each of the 4 sets of parameter settings: DeJong, Grefenstette, Goldberg and Alander.

Figure 5-16 shows the convergence of fitness plots of running NSC-GA on the Lymphoma cancer dataset and each of the 4 sets of parameter settings. The algorithm converged, producing the same maximum fitness of 1.968 for each of the 4 sets of parameter settings. However, the number of generations required for convergence differs, with Goldberg's set of parameter settings, convergence occurred after 133 generations, with Alander's set of parameter settings, convergence occurred after 312 generations, with Grefensette's set of parameter settings, convergence occurred after 387 generations and lastly with DeJong's, the convergence occurred after 1217 generations. Again, running the algorithm with Goldberg's set of parameter settings on this dataset resulted in obtaining the global optimum.

5.3.2.6. Ovarian cancer data

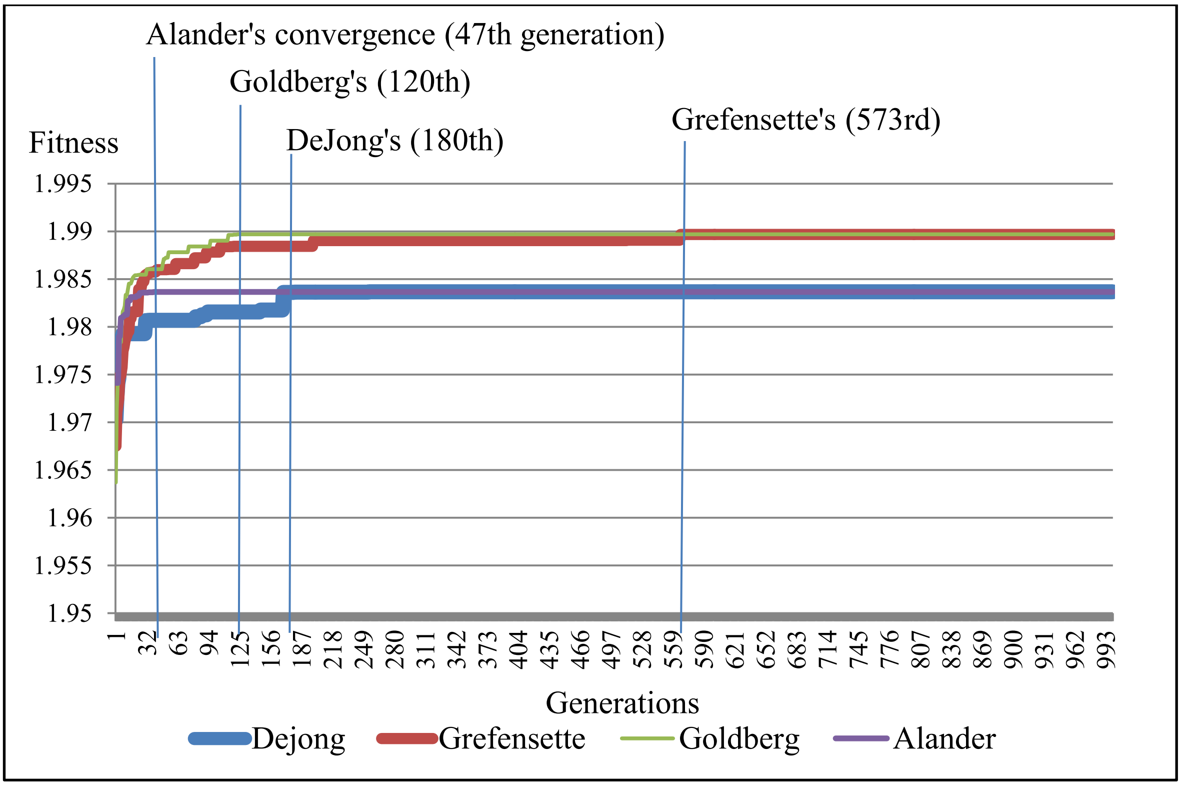


Figure 5-17 Typical convergence of fitness plots for Ovarian cancer data using each of the 4 different parameter settings from DeJong, Grefenstette, Goldberg and Alander

From Figure 5-17, it can be seen that the algorithm converged to different maximum fitness values for each of the 4 sets of parameter settings on the Ovarian cancer dataset. With Goldberg's and Grefensette's set of parameter settings, both achieved a maximum fitness of 1.989 but convergence occurred after 120 and 573 generations, respectively. Whilst with DeJong's and Alander's sets of parameter settings, convergence occurred after 180 and 47generations, respectively and both with a maximum fitness value smaller than 1.989. In this instance, the algorithm using Goldberg's set of parameter settings obtains a higher fitness value.

5.3.2.7. Prostate cancer data

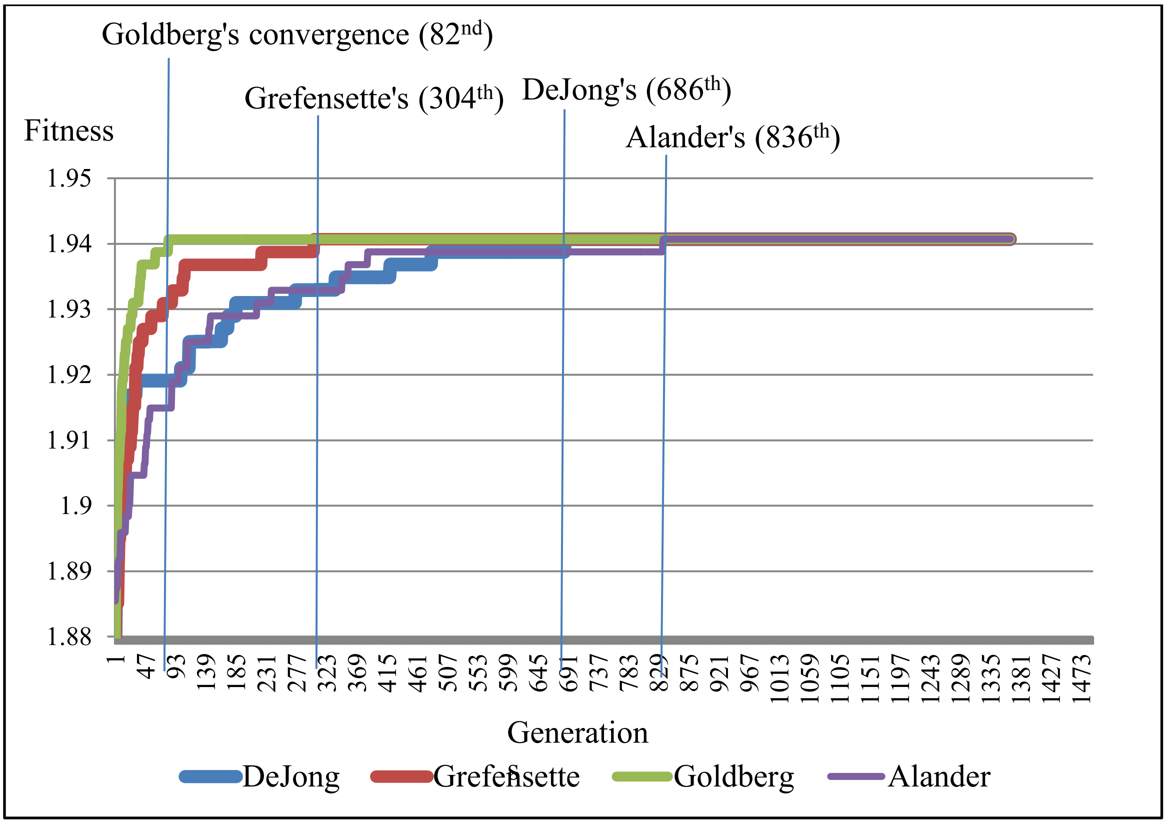


Figure 5-18 Typical convergence of fitness plots for Prostate cancer data with 4 different parameter settings from DeJong, Grefenstette, Goldberg and Alander

As seen in Figure 5-18, the algorithm converged to the global optimum with the same maximum fitness of 1.94 for each of the 4 sets of parameter settings. Again, the algorithm using Goldberg's set of parameter settings outperformed the other three sets of parameter settings in terms of computational time (faster convergence).

The results of parameter settings using the 4 parameter settings of DeJong, Grefensette, Goldberg and Alander are summarized in Table 5-3.

Table 5-3 Summary of results of running the algorithm using each of the 4 sets of parameter settings for 3 independent runs

Dataset	Parameter settings	Maximum fitness obtained		
		Run #		
		1	2	3
AD	Dejong	√	√	√
	Grefensette	√	√	√
	Goldberg	√	√	√
	Alander	√	√	√
Colon	Dejong			
	Grefensette		√	√
	Goldberg	√	√	√
	Alander			√
Leukemia	Dejong	√	√	
	Grefensette	√	√	√
	Goldberg	√	√	√
	Alander	√		
Lung	Dejong	√	√	√
	Grefensette	√	√	
	Goldberg	√	√	√
	Alander	√	√	√
Lymphoma	Dejong	√		
	Grefensette	√	√	√
	Goldberg	√	√	√
	Alander		√	
Ovarian	Dejong			
	Grefensette	√		√
	Goldberg	√	√	√
	Alander			
Prostate	Dejong	√	√	√
	Grefensette	√	√	√
	Goldberg	√	√	√
	Alander	√	√	√

From Table 5-3, it can be seen that the algorithm, using Goldberg's set of parameter settings, consistently achieves maximum fitness in each of the three runs for all of the seven datasets. In comparison, the algorithm using any one of the remaining three sets of parameters only achieve similar results in some of the runs for some of the seven datasets. In addition, on examining Table 5-1, the algorithm using Goldberg's set of parameter settings also consistently used least computation time to achieve convergence. Based on these observations, Goldberg's set of parameter settings is considered to be more suitable than the other three sets of parameter values.

Note that while the number of generations has been provided in terms of when convergence starts to occur, it is not used to measure the performance of the algorithm. It is only used as an indicative measure of computation time associated with a set of values for GA parameters used by the algorithm against the seven datasets in this study as these seven datasets are of different complexity (varying from 120 variables in the AD dataset to 15154 in Ovarian cancer dataset).

Given the lengthy computational time associated with the tuning process (see Table 5-1) and the aim of the study is to explore the feasibility of incorporating evolutionary approaches for finding interesting biomarkers that can differentiate between two classes (e.g. diseased vs. healthy) of biological data, a "*near optimal*" set of parameter settings that can be applied across a range of datasets and algorithms is acceptable. This is unlike the case where the aim is related to comparisons between the performances of one evolutionary algorithm against another evolutionary algorithm (that is, to show the performance of one evolutionary algorithm as being superior), where it is then important to ensure parameters associated with each of these algorithms are optimally tuned. Arcuri and Fraser (2011) has argued that, in the case where parameter tuning is expensive and the investigation is focused on examining the performance of new techniques, using a set of default values is acceptable. Hence, Goldberg's set of parameter settings will subsequently be used in the evolutionary-based approaches described in Chapter 5, 6 and 7 in this study. The following table shows the complete set of parameter settings.

Table 5-4 Parameter settings used in Chapter 5, 6 and 7

Parameters	Values/operators
Population Size	30
Chromosome length - Real encoding	10
Crossover probability (P_c)	0.6
Mutation Probability (P_m)	0.033
Maximum generations	1000
Selection	Tournament
Crossover	Single Point
Mutation	Uniform
Elitist	Single

Note that the set of parameters shown in Table 5-4 is the same as the one (Goldberg's) in Table 5-2, except for the maximum number of generations which is now set for 1000 instead of 5000. This is due to the fact that with the Goldberg's parameter settings, the algorithm obtains the global optimum with the maximum fitness in less than 1000 generations for all seven datasets, thus allowing some savings in computational time. As the approach also checks the state of convergence, the number of generations in specific instances can be varied if required. The set of parameter setting is to be selected on the basis that it consistently allows the algorithm to converge with maximum fitness using less computation time.

5.4. Experiment results

The proposed approach was evaluated using seven datasets: AD, Colon, Leukemia, Ovarian, Lymphoma, Lung and Prostate cancer datasets. For each dataset, 15 independent runs of NSC-GA were executed using the respective training data and parameter values shown in Table 5-4. For each run, 10 fold CV strategy described in Section 3.2 was employed to evaluate the selected feature sets. The optimal set of features was then used to construct the NSC classifier to classify the unseen test data associated with the dataset. The classification results for classifying the unseen test data were recorded and the average classification result from 15 independent runs was calculated. The following sections detail the results obtained from applying NSC-GA on each of the seven datasets. Where appropriate, the comparison of the performance of the proposed algorithm with existing work is based on classification accuracy and the selected feature sets.

5.4.1. Ray *et al.* Alzheimer's Disease (AD) data

As mentioned previously in Section 3.1.1, this dataset consists of 120 attributes. The training set consists of 43 AD and 40 NDC samples and 2 test sets: the AD test set consists of 42 AD, 50 NAD samples and the MCI test set consists of 22 AD and 25 NAD samples. More details about this dataset can be found in Section 3.1.

The optimal shrinkage thresholds obtained upon convergence were used to evaluate the training dataset first, and then applied to the unseen test dataset using the NSC classifier. A convergence plot from one of the typical runs is shown in and results are in Table 5-5.

Table 5-5 Classification results for the AD data using NSC-GA approach and from Ray *et al*. (2007)

Approach	Alzheimer		
	AD		MCI
	No of attributes	Average classification accuracy on unseen test dataset (%)	Average classification accuracy on unseen test dataset (%)
Proposed approach NSC-GA	11	89.49	79
NSC (Ray *et al*., 2007)	18	89	81

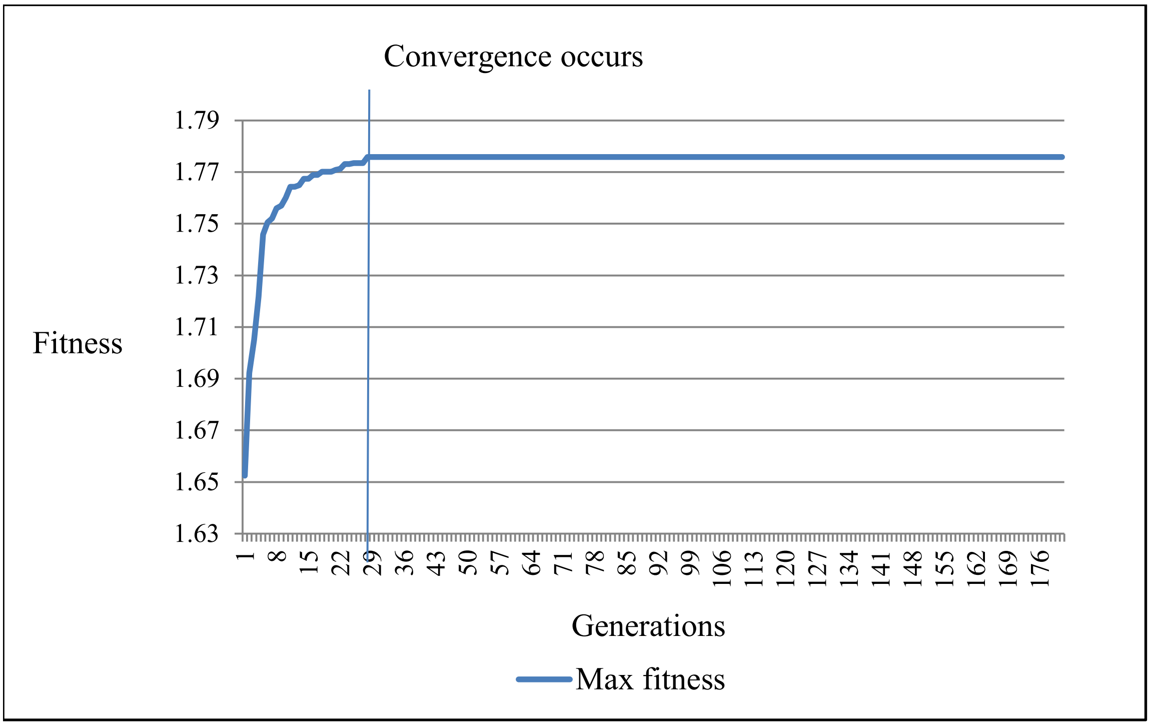


Figure 5-19 A typical convergence of fitness plot for the training set of AD

As seen in Figure 5-19, convergence occurred after 28 generations with the maximum fitness of 1.775. Although convergence was achieved in 28 generations, the optimal solution actually involved a total of 8400 evaluations which is not a small number given that this dataset with only 120 variables is considered to be of “low dimensionality” relative to most other biological datasets. The length of chromosomes is 10, representing 10 shrinkage threshold values. With a population size of 30, the evaluation in each generation involved 30* 10 shrinkage threshold values. Convergence after 28 generations would mean that a total of 8400 evaluations.

The optimal chromosome obtained for each of the runs had the same maximum fitness of 1.775, which resulted in a set of 11 features. This set of 11 features (proteins) is a subset of the 18 biomarkers (proteins) found in Ray *et al*. (2007)’s experiment and is shown in Table 5-6 .

Table 5-6 List of 11 proteins selected using NSC-GA

Features (Proteins)					
PDGF-BB_1	RANTES_1	IL-1a_1	TNF-a_1	EGF_1	M-CSF_1
ICAM-1_1	IL-11_1	IL-3_1	GCSF_1	ANG-2_1	

The same set of 11 proteins was obtained from each of the 15 independent runs. The average classification accuracy from 15 runs for the unseen AD test set was 89.49% and for the unseen MCI test set was 79% using the set of 11 proteins found in this study. These are similar to the result of 89% for unseen AD test set and 81% for the unseen MCI test dataset using 18 proteins obtained in Ray *et al.* (2007)'s study. The remaining 7 proteins excluded here from the original 18 protein signatures (Ray *et al.*, 2007) were also not included in the 6 and 5 protein signatures found in Ravetti and Moscato (2008)'s study. According to Ray and Wyss-coray (2010), TRAIL-R4 and IGFBP-6 proteins from the 7 excluded proteins are optional in the list of biomarkers for a diagnostic analysis of AD.

5.4.2. Alon *et al.* Colon cancer data

The Colon dataset consists of 2000 attributes, 40 Tumour (T) and 22 Normal (N) samples. The training set consists of 30 T and 16 N samples, and the test set consists of 10 T and 6 N samples. More details about this dataset can be found in Section 3.1.

Using the same procedure, 15 independent runs of NSC-GA was executed with 10 fold CV using the Colon dataset with the GA parameter setting listed in Table 5-4. The optimal shrinkage threshold value from the fittest chromosome upon convergence was used to evaluate on the training dataset first, and then applied to the unseen test dataset using the classifier in the NSC. A convergence of fitness plot from one of the typical runs is shown in Figure 5-20 and classification results using the optimal sets of features are in Table 5-7.

Table 5-7 Classification results, for the Colon data using NSC-GA approach, and from Klassen and Kim (2009)

Approach	Colon	
	No of attributes	Average classification accuracy on unseen test dataset (%)
Proposed approach NSC-GA	28 6	100 93.75
NSC (Klassen & Kim, 2009)	16	75

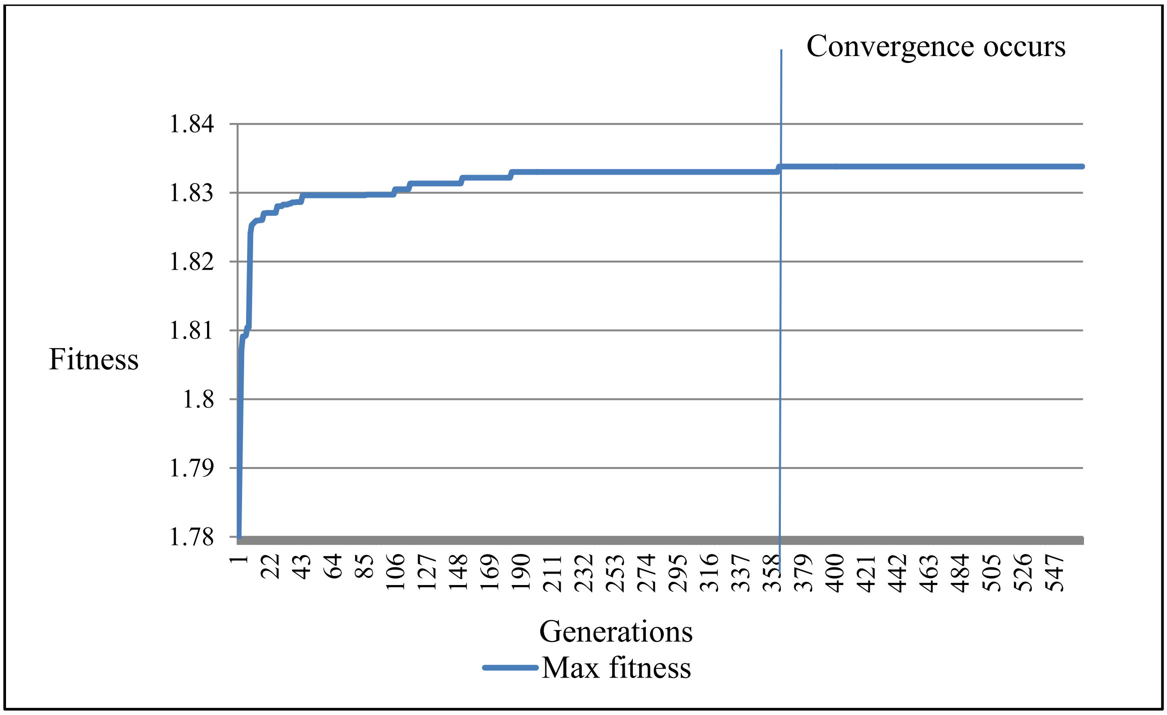


Figure 5-20 A typical convergence plot for the training set of Colon cancer data

As seen in Figure 5-20, the convergence of fitness occurred after 362 generations with the maximum fitness of 1.833. Nine runs had the maximum fitness of 1.833 which gave the same set of 28 features (genes) for the Colon dataset. Six runs had the maximum fitness of 1.823 which gave the same set of 6 genes which is a subset of the 28 gene set. The average classification accuracy from 15 independent runs was 97.5% on the unseen test set (93.75% for 6 gene set and 100% for 28 gene set) in comparison to 75% classification accuracy using 16 genes reported in Klassen and Kim (2009)'s experiments. It is not possible to check the set of 28 and 6 genes found by the proposed approach against the set of 16 genes from Klassen and Kim (2009) as these were not listed in their study. The set of 28 genes from this study are listed by their accession number in Table 5-8 (the highlighted genes belong to the set of 6 genes).

Table 5-8 Twenty eight genes selected for Colon cancer data using NSC-GA

Gene accession numbers						
T95018	X55715	M63391	H40560	T92451	T57619	R78934
T58861	M26697	M76378	R87126	H43887	H64489	M22382
T71025	Z24727	Z50753	X12671	T47377	L05144	H55758
M64110	M76378	T60155	M76378	J02854	X86693	T60778

5.4.3. Leukemia cancer data

The Leukemia dataset consists of 7129 attributes, 47 ALL and 25 AML samples. The training set consisting of 27 ALL and 11 AML samples, and the test set consisting of 20 ALL and 14 AML samples. More details about this dataset can be found in Section 3.1.

Similar to the experiments above, 15 independent runs with 10 fold CV were carried out using the Leukemia dataset. A convergence of fitness plot from one of the typical runs is shown in Figure 5-21 and classification results on unseen test dataset are in Table 5-9.

Table 5-9 Classification results for the Leukemia data using NSC-GA approach and from Tibshirani *et al*. (2002), Klassen and Kim (2009), S. Wang and Zhu (2007) and J. Fan and Fan (2008)

Approach	Leukemia	
	No of attributes	Average classification accuracy on unseen test dataset (%)
Proposed approach NSC-GA	9	97.05
NSC (Tibshirani *et al*., 2002)	21	94.12
NSC (Klassen & Kim, 2009)	21	94.12
ALP-NSC, AHP-NSC (S. Wang & Zhu, 2007)	16	94.12
FAIR (Fan & Fan, 2008)	11	97.05

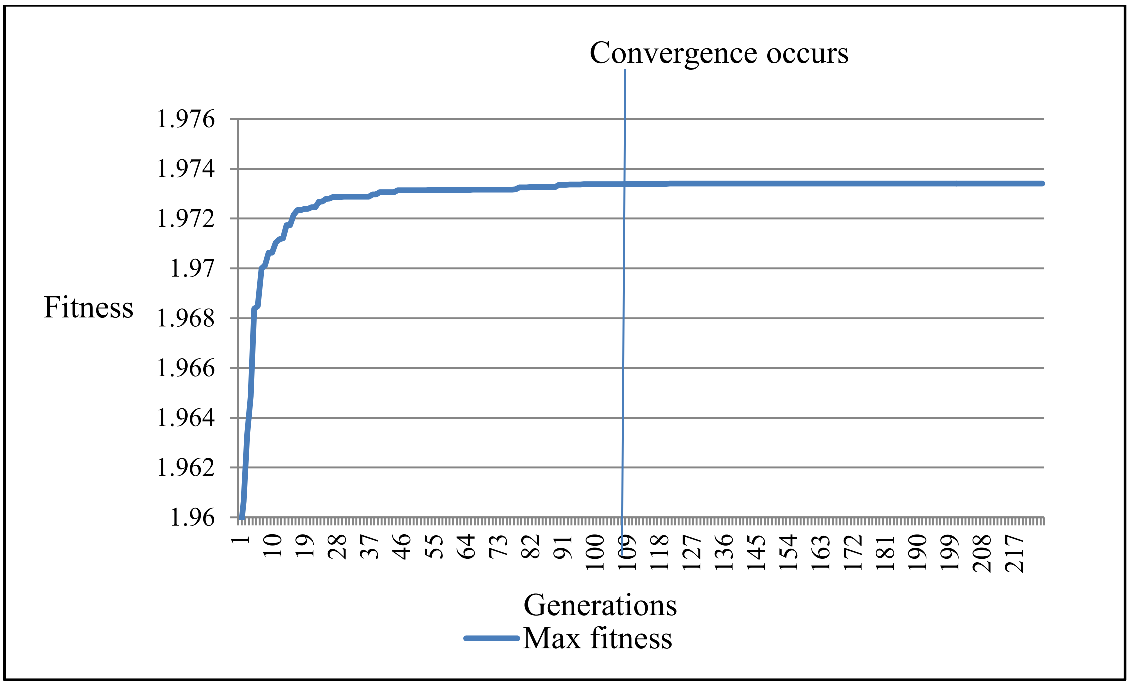


Figure 5-21 A typical convergence plot for the training set of Leukemia cancer data.

As seen in Figure 5-21, the convergence occurred after 109 generations with the maximum fitness of 1.973. The optimal shrinkage thresholds obtained from each of the 15 independent runs had the same maximum fitness of 1.973 which produced the same set of 9 features (genes) for the Leukemia cancer dataset. The set of nine features gave the classification accuracy of 97.05% on the unseen test dataset. Seven out of the nine genes listed in Table 5-10 (highlighted genes) are a subset of the 16 genes reported in S. Wang and Zhu's study (2007). Two genes having accession numbers M96326 and M28310 are not present in that set of 16 genes. It is not possible to check the set of 9 genes found using NSC-GA against the set of 11 genes in J. Fan and Fan (2008) as these were not listed in their study.

The proposed approach achieved a higher classification accuracy 97.05% using a smaller number of genes, 9, as compared to 94.12% classification accuracy with 21 genes reported in Tibshirani *et al*. (2002) and Klassen and Kim (2009), and 94.12% using 16 genes reported in S. Wang and Zhu (2007), and achieved the same classification accuracy of 97.05 but using the smaller set of 9 features compared to J. Fan and Fan (2008). The set of 9 genes from this study are listed by their accession number in Table 5-10. An interesting point here is, when comparing the results obtained via NSC-GA with those reported by Tibshirani *et al*. (2002), Klassen and Kim (2009), and S. Wang and Zhu (2007); all involved the same method, NSC, with S. Wang and Zhu (2007) having made attempts to improve the the original NSC approach via adaptive L1-norm penalized NSC (ALP-NSC) and adaptive hierarchically penalized NSC (AHP-NSC). The results reported for NSC-GA is an average of 15 runs and is obtained by trying to automatically find the optimal value for the shrinkage threshold for the original NSC method. The NSC-GA results lends support to the hypothesis, " an automatic approach that can effectively explore the search space to find a more precise shrinkage threshold value for NSC may result in an optimal value leading to a better classification result", as it produced a shrinkage threshold value that leads to the selection of 9 genes with a classification accuracy of 97.05%.

Table 5-10 Nine genes selected by the proposed NSC-GA for Leukemia cancer data

Gene accession number	Gene definition
M27891	*CST3 Cystatin C (amyloid angiopathy and cerebral hemorrhage)*
M84526	*Human adipsin/complement factor D mRNA, complete cds*
M96326	*Human azurocidin gene, complete cds*
U46751	*Phosphotyrosine independent ligand p62 for the Lck SH2 domain mRNA*
U50136	*Leukotriene C4 synthase (LTC4S) gene*
X17042	*Human mRNA for hematopoetic proteoglycan core protein*
X95735	*Homo sapiens mRNA for zyxin*
M28310	*Mus musculus 3/10 metalloproteinase inhibitor gene, exon 3*
Y00787	*Human mRNA for MDNCF (monocyte-derived neutrophil chemotactic factor)*

5.4.4. Ovarian cancer data

The Ovarian dataset consists of 15154 attributes, 162 Disease (D) and 91Normal (N) samples. The training set consists of 81 D and 45 N samples, and the test set consists of 81 D and 46 N samples. More details about this dataset can be found in Section 3.1. Similar to the experiments above, 15 independent runs with 10 fold CV were carried out using the Ovarian dataset. A convergence of fitness plot from one typical run is shown in Figure 5-22 and classification results on the unseen test dataset are in Table 5-11.

Table 5-11 Classification results for the Ovarian data using NSC-GA approach and from Foss (2010)

Approach	Ovarian	
	No of attributes	Average classification accuracy on unseen test dataset (%)
Proposed approach NSC-GA	7	96.06
GCLUS and SERA (Foss, 2010)	47	97.63

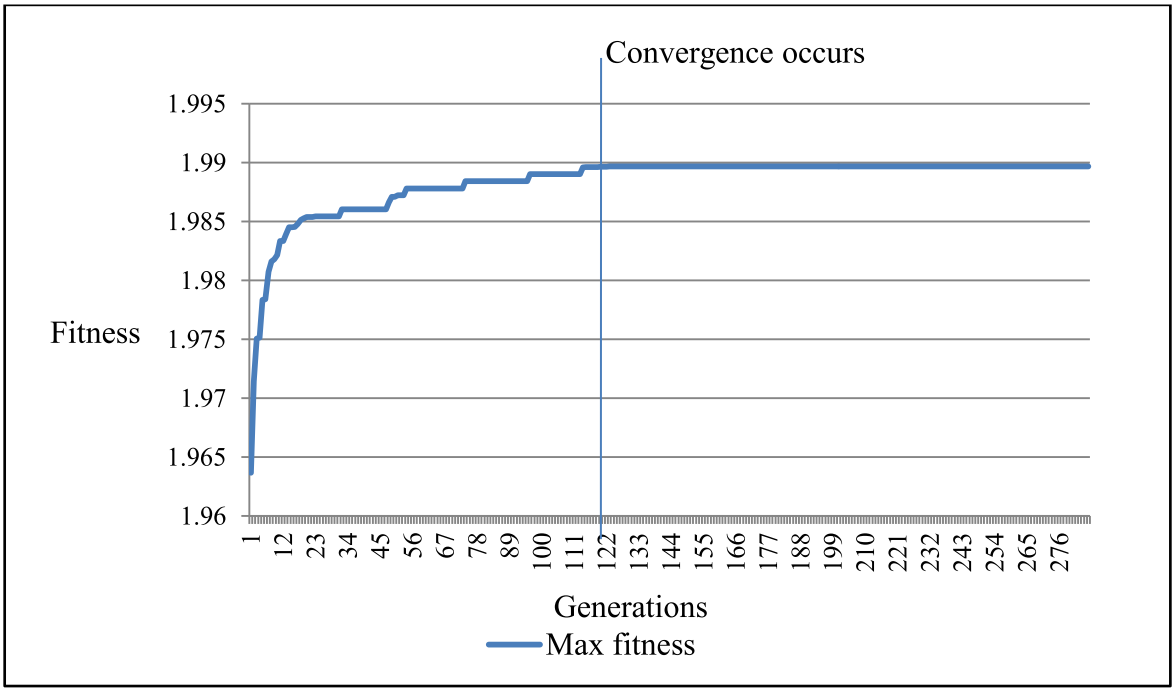


Figure 5-22 A typical convergence plot for the training set of Ovarian cancer data

As seen in Figure 5-22, the convergence occurred after 115 generations at the maximum fitness of 1.989. The optimal shrinkage thresholds obtained for each of the 15 independent runs had the same maximum fitness of 1.989 which produced the same set of 7 features (peptides), MZ244.36855, **MZ244.66041, MZ244.95245, Z245.24466, MZ245.8296, MZ245.53704** and **MZ246.12233**. These 7 peptides are a subset of 47 peptides reported in Foss (2010). Six peptides (in bold) are among the top 10 peptides reported in Yap, Tan and Pang (2013). The average classification accuracy from 15 independent runs on the Ovarian unseen test dataset using this set of 7 selected peptides was 96.06%, compared to 97.63% using the set of 47 peptides using the Implementation of the MAXCLUS framework (GCLUS) and Statistical Error Rate estimation Algorithm (SERA) in Foss (2010).

5.4.5. Lymphoma cancer data

The Lymphoma dataset consists of 4026 attributes, 24 GGL and 23 ACL samples. The training set consisting of 17 GCL and 17 ACL samples, and the test set consisting of seven GGL and six ACL samples. More details about this dataset can be found in Section 3.1.

Fifteen independent runs with 10 fold CV were carried out using the Lymphoma dataset. A convergence of fitness plot from one of the typical runs is shown in Figure 5-23 and classification results on unseen test dataset are in Table 5-12.

Table 5-12 Classification results for the Lymphoma data using NSC-GA approach and from Klassen and Kim (2009)

Approach	Lymphoma	
	No of attributes	Average classification accuracy on unseen test dataset (%)
Proposed approach NSC-GA	7 12 128 129 132	95.45 95.45 100 100 100
NSC (Klassen & Kim, 2009)	25	86.6

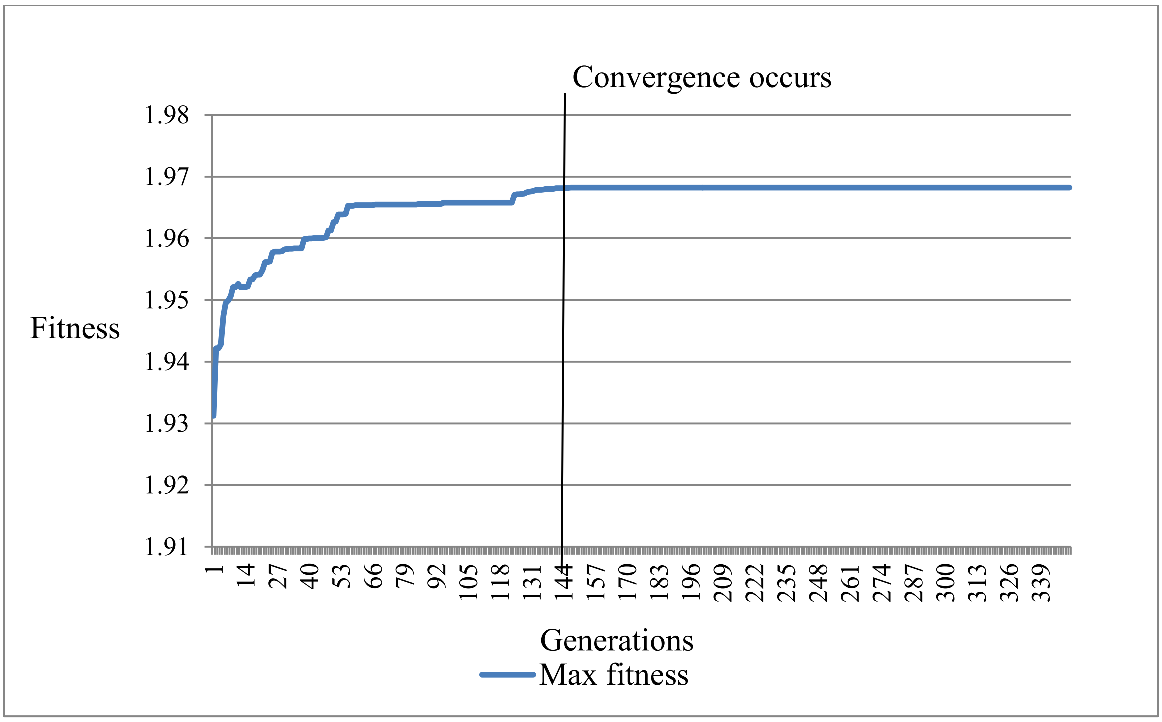


Figure 5-23 A typical convergence plot for the training set of Lymphoma cancer data

As seen in Figure 5-23, the convergence occurred after 145 generations with the maximum fitness of 1.968. From the 15 independent runs, 10 runs resulted in a shrinkage threshold that mapped to the same set of 128 features, one run resulted in a set of 129 features, one run resulted in a set of 132 features, one run gave a set of 7 features, and one run gave a set of 12 features. The set of 128, 129 and 132 features leads to the same classification accuracy of 100% and the set of 7 and 12 features resulted in the same classification accuracy of 95.45 on the unseen test set. The average classification accuracy from 15 runs was 99.39% on the unseen test set. The smaller set of features is a subset of the larger set, e.g., set of 7 features is a subset of the set of 12 features and both are subsets of the set of 128 features. Biomedical domain knowledge can be used to examine these sets further to make better informed decision for subsequent diagnostic test development. The set of 7 and 12 genes are listed by their accession number in Table 5-13.

Table 5-13 The sets of 7 and 12 genes selected by NSC-GA for Lymphoma cancer data

Gene accession number	Set	
	12 genes	7 genes
GENE3327X	√	√
GENE3329X	√	√
GENE3330X	√	√
GENE3332X	√	√
GENE3361X	√	√
GENE3258X	√	√
GENE3256X	√	√
GENE3328X	√	
GENE3314X	√	
GENE3260X	√	
GENE1252X	√	
GENE3967X	√	

5.4.6. Lung cancer data

The Lung dataset consists of 12533 attributes, 150 ADCA and 31 MPM samples. The training set consisting of 134 ADCA and 15MPM samples, and the test set consisting of 16 ADCA and 16 MPM samples. More details about this dataset can be found in Section 3.1.

Fifteen independent runs with 10 fold CV were carried out using the Lymphoma dataset. A convergence of fitness plot from one of the typical runs is shown in Figure 5-24 and classification results on unseen test dataset are in Table 5-14.

Table 5-14 Classification results for the Lung data from NSC-GA approach and from Klassen and Kim (2009), Tai and Pan (2007) and J. Fan and Fan (2008)

Approach	Lung	
	No of attributes	Average classification accuracy on unseen test dataset (%)
Proposed approach NSC-GA	8 9 10 11	100
NSC (Klassen & Kim, 2009)	5	93.7
Weighted NSC (Tai & Pan, 2007)	6	99.99
FAIR (Fan & Fan, 2008)	31	95.3

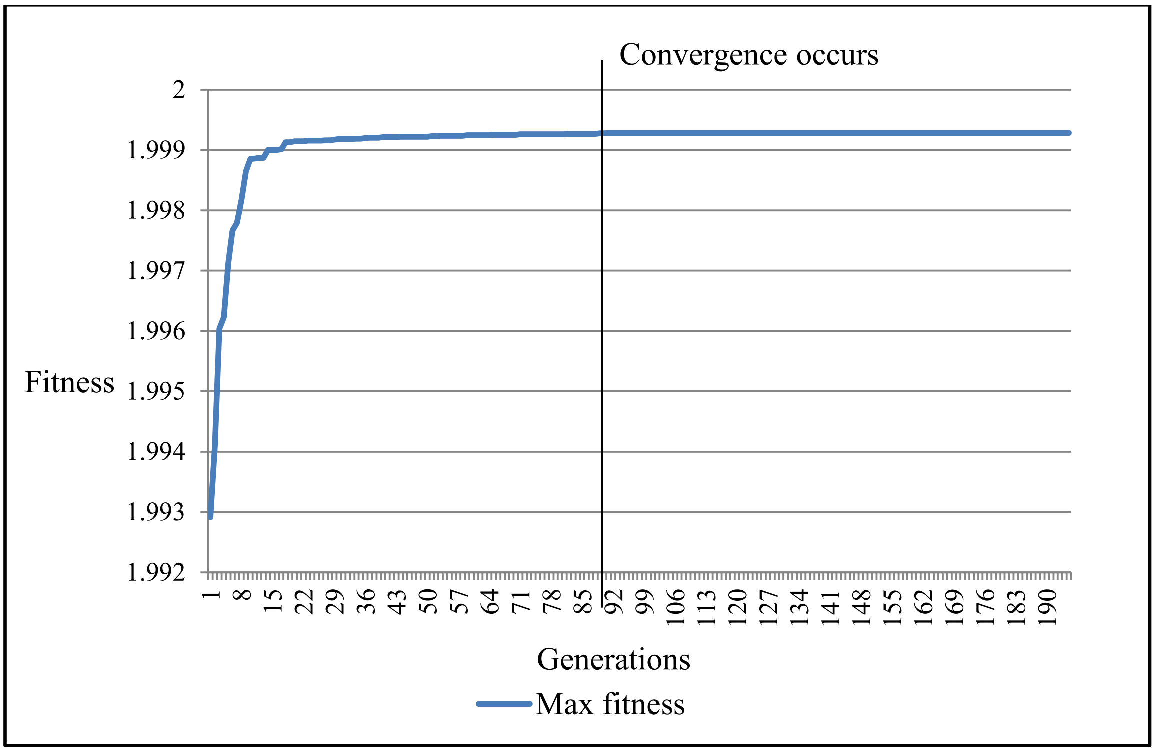


Figure 5-24 A typical convergence plot for the training set of Lung cancer data

As seen in Figure 5-24, the convergence occurred after 88 generations with the maximum fitness of 1.999. Three runs resulted in the same set of 8 features for the Lung cancer dataset, 6 runs resulted in a set of 9 features, 2 runs resulted in a set of 10 features and 4 runs resulted in a set of 11 features. The average classification accuracy of the sets of 8, 9, 10 and 11 features is 100% on the unseen test set. The set of 8, 9, 10 and 11 listed by their accession genes are number in Table 5-15.

Table 5-15 The sets of 8, 9, 10 and 11 genes selected by NSC-GA, for Lung cancer data

Gene accession number	Set			
	11 genes	10 genes	9 genes	8 genes
32551_at	√	√	√	√
33328_at	√	√	√	√
34320_at	√	√	√	√
36533_at	√	√	√	√
37157_at	√	√	√	√
37716_at	√	√	√	√
37954_at	√	√	√	√
40936_at	√	√	√	√
33833_at	√	√	√	
33327_at	√	√		
35823_at	√			

5.4.7. Prostate cancer data

The Prostate dataset consists of 12600 attributes, 77 Tumour (T) and 59 Normal (N) samples. The training set consisting of 52 T and 50N samples, and the test set consisting of 25 T and 9 N samples. More details about this dataset can be found in Section 3.1.

Fifteen independent runs with 10 fold CV were carried out using the Prostate dataset. A convergence of fitness plot from one of the typical runs is shown in Figure 5-25 and classification results on the unseen test dataset are in Table 5-16.

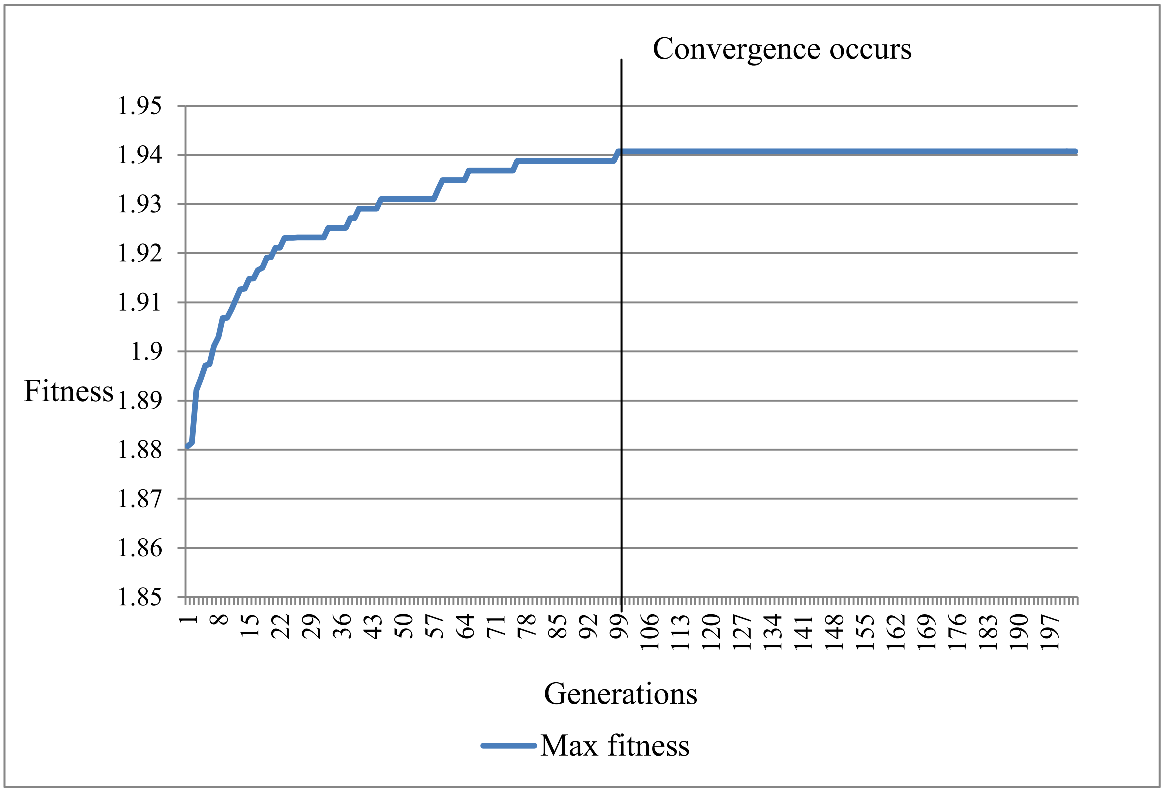


Figure 5-25 A typical convergence plot for the training set of Prostate cancer data

As seen in Figure 5-25, the convergence occurred after 99 generations with the maximum fitness of 1.94. The optimal shrinkage thresholds obtained for each of the 15 independent runs had the same maximum fitness of 1.94 which produced the same set of 6 genes, 31444_s_at, 41468_at, 37639_at, 38406_f_at, 769_s_at and 556_s_at. The average classification accuracy using the 6 gene set from 15 runs on the unseen test set was 90.2%, as shown in Table 5-16.

Table 5-16 Classification results for the Prostate data from NSC-GA approach and from Klassen and Kim (2009), Tai and Pan (2007) and J. Fan and Fan (2008)

Approach	Prostate			
	No of attributes	C1 (Tumour)	C2 (Normal)	Average classification accuracy on unseen test dataset (%)
Proposed approach NSC-GA	6	80	100	90.2
NSC (Klassen & Kim, 2009)	6			90.91
Weighted NSC (Tai & Pan, 2007)	10			60.51
FAIR (Fan & Fan, 2008)	2			73.52

The column headings C1 and C2 in the table stand for average classification accuracy (%) on the Prostate unseen test dataset for the Tumour class and Normal class, respectively. The column heading "*Average Test*" stands for the overall average classification accuracy (%) on the Prostate unseen test dataset for the 15 independent run. "*Average Test*" is calculated using Equation (5.3).

A summary of the results for the AD, Colon, Leukemia, Ovarian, Lymphoma, Lung and Prostate cancer datasets are shown in Table 5-17.

Table 5-17 Classification results for AD, Colon, Leukemia, Ovarian, Lymphoma, Lung and Prostate cancer data using NSC-GA

Approach	Alzheimer			Colon		Leukemia		Ovarian		Lymphoma		Lung		Prostate	
	AD		MCI												
	No attr	Test (%)	Test (%)	No attr	Test (%)	No attr	Test (%)	No attr	Test (%)	No attr	Test (%)	No attr	Test (%)	No attr	Test (%)
Proposed approach NSC-GA	11	89.49	79	28 6	100 93.75	9	97.05	7	96.06	7 12 128 129 132	95.45 100	8 9 10 11	100	6	90.2
NSC (Ray *et al*., 2007)	18	89	81												
NSC (Tibshirani et al., 2002)						21	94.12								
NSC (Klassen & Kim, 2009)				16	75	21	94.12			25	86.6	5	93.7	6	90.91
ALP-NSC, AHP-NSC (S. Wang & Zhu, 2007)						16	94.12								
Weighted NSC (Tai & Pan, 2007)												6	99.55	10	60.51
FAIR (Fan & Fan, 2008)						11	97.05					31	95.3	2	73.52
GCLUS & SERA (Foss, 2010)								47	97.63						

5.5. Discussion

Table 5-17 shows a summary of experimental results achieved by the proposed approach, NSC-GA in comparison to existing work that used the same datasets. The classification accuracy rate reported for each dataset for NSC-GA is based on the average classification accuracy of 15 independent runs. NSC-GA achieved similar classification results as in Ray *et al*. (2007) on AD and MCI independent test datasets, and also improved FS and/or classification accuracy on the other 6 datasets in terms of obtaining smaller sets of features and higher or similar classification accuracy on unseen test sets compared to the existing results (as reported in Klassen and Kim (2009), S. Wang and Zhu (2007), J. Fan and Fan (2008), Foss (2010), and Tai and Pan (2007)). For example, for the Leukemia cancer dataset, NSC-GA obtained a smaller set of 9 features and higher classification accuracy of 97.05% and for the Ovarian cancer dataset, a smaller set of 7 features was obtained with the similar classification accuracy of 96.06%. In terms of the Lung cancer dataset, Tai and Pan (2007) achieved 99.55 using a set of 6 features whereas in NSC-GA using 8 features to obtain 100% classification accuracy. However, as the actual features used have not been listed in Tai and Pan's paper. It is not possible to compare the results in terms of the actual features.

When comparing results obtained via NSC-GA with other NSC-based approaches (Tibshirani *et al*. (2002), Klassen & Kim (2009), S. Wang & Zhu (2007)), it can be seen that NSC-GA generally found optimal feature sets that have a smaller number of features and better classification results. This outcome is achieved by using GA to automatically explore the search space to find a more precise shrinkage threshold value for NSC, thus overcoming limitations typically associated with "trial and error" approaches. Unlike approaches (e.g. S. Wang & Zhu (2007)) that attempts to improve the performance of NSC by modifying it, this result is obtained using the original NSC algorithm, thus potentially the proposed approach can also be incorporated into modified NSC for further improvements.

Having information as shown in Table 5-17, with the Colon, Lymphoma and Lung cancer datasets, each are associated with multiple sets of features which are subsets of each other. This allows the domain expert to make informed decision in terms of sets of features that could be selected for further investigations. For example, in the case of the

Colon cancer dataset, one can make decisions based on the tradeoffs between classification accuracy and size of feature set. It can be seen that in the case of the Lymphoma cancer dataset, the set of 6 features resulted in the same classification accuracy as the set of 12 features (i.e. 95.45%). The domain expert can examine the 6 additional features in the set of 12 and use domain knowledge to decide on their potential relevance and make decision on subsequent analysis. Equally it is interesting to further analyse the Lung cancer dataset where sets with 8, 9, 10 and 11 features respectively resulted in classifiers producing the same classification accuracy on the unseen test dataset (100%). It appears that a major contributing factor relates to 8 features and thus may warrant further investigations into the relevance of the remaining features. This sort of information for analysis in bioinformatics is important as reducing the number of features to a smaller promising set for further investigations would reduce costs associated with future experiments and analysis. The set of selected features from a biological perspective implies that the level of expressions associated with the selected biomarkers differ significantly between disease and non-disease.

From Table 5-16, the NSC classification results associated with the set of 6 features mostly showed high specificity but low sensitivity, e.g., sensitivity (C1) is 80% but specificity (C2) is 100%, implying the majority of the truly *not-at-risk* cases will be correctly identified, but some of the truly *at-risk* cases will also be incorrectly identified as *not-at-risk*. Continuing the investigation about classifier bias that was initiated in Chapter 4, further analysis is carried out using the Prostate cancer data and the corresponding set of 6 genes identified via NSC-GA. This set of features are used to construct 22 different classifiers from WEKA software (Hall *et al.*, 2009) for classifying the unseen test dataset. The aim here is to further examine the trend observed in Chapter 4 in terms sensitivity and specificity being associated with specific classifiers (in this case NSC). The classification results from 22 different classifiers are shown in Table 5-18.

Table 5-18 Results of classification for the 6 selected genes with 22 WEKA classifiers on the Prostate unseen test set

Classifier	Set of 6 genes		
	C1 (%)	C2 (%)	Average (%)
SMO	80	96.2	88.2
Simple Logistic	88	100	94.1
Logistic	80	92.3	86.3
Multilayer Perceptron*	88.8	98.8	93.9
Bayes Net	80	88.5	84.3
Naïve Bayes	**92**	**92.3**	**92.2**
Naïve Bayes Simple	92	88.5	90.2
Naïve Bayes Up	**92**	**92.3**	**92.2**
IB1	84	92.3	88.2
KStar	68	96.2	82.4
LWL	84	92.3	88.2
AdaBoost	**92**	**92.3**	**92.2**
ClassVia Regression	**92**	**92.3**	**92.2**
Decorate*	82	94.35	88.2
Multiclass Classifier	80	92.3	86.3
Random Committee*	92	88.8	90.4
j48	92	88.5	90.2
LMT	96	88.5	92.2
NBTree	72	92.3	82.4
Part	92	88.5	90.2
Random Forest*	**92.8**	**91.55**	**92.2**
Ordinal Classifier	92	88.5	90.2

Mixed results, with regards to the classifiers used and their corresponding sensitivity and specificity, were obtained. Naïve Bayes, Naïve Bayes Updateable, AdaBoost and ClassVia Regression (in bold) produced results showing high sensitivity (92%) and high specificity (92.3%). However, there are also other classifiers showing behaviour similar to that of the NSC classifiers (i.e. lower sensitivity and higher specificity). Also interestingly, there are a number of classifiers demonstrating higher sensitivity (shaded cells) than specificity. These results demonstrated that the use of specific classifiers may have an impact on the sensitivity and specificity obtained using a set of features in classification. Thus in a DM analysis for finding suitable sets of biological markers, a number of classifiers should be used instead of just using one. This will avoid missing out on sets of features with high discriminatory capabilities that should be further investigated in early diagnostic test developments.

5.6. Summary

This chapter describes the proposed approach of incorporating NSC and a single objective algorithm, GA, to overcome the limitations of previous approaches such as empirical methods with NSC, by 1) searching automatically for the optimal shrinkage threshold value for NSC, and 2) obtaining the optimal set of minimal number of features for higher classification results. This advantage here is due to the fact that the proposed approach employs GA as a search algorithm to find optimal shrinkage thresholds based on the fitness evaluation from the NSC. An additional advantage of the proposed approach is the use of computers to run the algorithm for finding the optimal shrinkage threshold values automatically. This is unlike the traditional NSC approach involving manual shrinkage threshold value selection where the user spends a lot of time and effort to choose the optimal shrinkage threshold value via trial and error.

A further analysis was also carried out on the Prostate cancer data using the same set of 6 genes to construct 22 different classifiers from WEKA software (Hall *et al.*, 2009) to investigate the impact of using different classifiers on sensitivity and specificity.

To continue the exploration of evolutionary approaches for FS in biological data, the following chapter describes an approach incorporating MA for automatically finding optimal shrinkage thresholds for NSC, an attempt to further improve upon the NCS-GA approach described in this chapter.

6. Incorporating the NSC algorithm into MA

Chapter 5 described an approach incorporating NSC into GA (NSC-GA) to automatically search for optimal shrinkage thresholds in NSC leading to the selection of optimal sets of features with better classification accuracy. According to Elbeltagi, Hegazy, & Grierson (Elbeltagi *et al.*, 2005), computation time associated with GA processing is intensive. One of the factors that contribute to the quality of optimal solutions in EA is the evaluation of fitness of individuals in the population. That is, the better the fitness evaluation the better the quality of the optimal solution. One of the approaches to improve GAs both in processing time and quality of optimal solutions is the use of a MA (Elbeltagi *et al.*, 2005). MA (Albrechtsson et al., 2001) is a hybrid algorithm that incorporates an EA and a local search (LS) to search for a local optimum to further improve its fitness (Elbeltagi et al., 2005; Krasnogor & Smith, 2005; Wu, 2001b). As a result of increased exploitations, fitness of each chromosome is improved significantly in each generation, leading to a faster convergence in population fitness, and subsequently, computation time for the evolutionary process is reduced. The quality of the optimal solution could also be improved owing to the fact that chromosomes have already been evaluated locally by LS before being subjected to a global search for the optimal solution.

In this chapter, an approach of incorporating the NSC algorithm into a MA, namely NSC-MA, for automatically searching for an optimal range of shrinkage threshold values is proposed. The aim here is to explore how to improve the NSC-GA approach.

MA has been described in Section 2.3.5. The following section describes the proposed approach of incorporating the NSC algorithm and MA, the results are reported in Section 6.2, the discussion is in Section 6.3 and the summary is in Section 6.4.

6.1. The proposed approach, NSC-MA.

The combination of EA and LS makes MA more efficient and effective in terms of processing time for convergence to optimal solutions, finding smaller sets of features, and improving classification accuracy in comparison to other traditional EAs such as GA (Elbeltagi *et al.*, 2005; Zhu *et al.*, 2007). Different LS strategies such as pair-wise LS (Merz & Freisleben, 1999), Adding Subtracting LS, I*mprovement First Strategy* LS and greedy LS (Zhu *et al.*, 2007) have been incorporated into GA in different approaches. For example, a LS strategy can be applied to elite chromosomes only or to the entire population or only to chromosomes that have been modified by crossover and/or mutation operation, etc.

Adding Subtracting LS strategy is carried out to search for a better chromosome in terms of fitness by adding or subtracting a small random value generated by a RNG to a meme (gene) value in the chromosome to create a new chromosome. The fitness of the new chromosome is then evaluated, if improved (i.e. better fitness) the new chromosome is retained, otherwise discarded. The process continues for the rest of the memes in the chromosome (Elbeltagi *et al.*, 2005). This strategy has been known as a *greedy search strategy* (Zhu *et al.*, 2007) or a *hill climbing search strategy* where the search progresses from the current best chromosome to the one that has a better fitness (Kohavi & John, 1997; H. Wang *et al.*, 2009). MA with the *greedy search strategy* LS outperformed GA in terms of achieving better classification accuracy and processing time (Elbeltagi *et al.*, 2005). However, according to Zhu, *et al.* (2007)'s study, the LS with *Improvement First Strategy* outperformed the *greedy search* strategy LS. Their study also found that the *Improvement First Strategy* LS when applied to a few of the elite chromosomes resulted in better solutions when compared to the approach that applied the *Improvement First Strategy* LS to all chromosomes in the population.

Improvement First Strategy LS was also employed as a LS to find a local optimum for offspring generated from crossover and mutation operation. That is, after the application of crossover and mutation operators on chromosomes, *Improvement First Strategy* LS is then applied to offspring for searching for a local optimum. According to Krasnogor and Smith (2005)'s experiments, MA with *Improvement First Strategy* LS outperformed other LS strategies such as *Multi Start Local Search* (MSLS) (Marchiori, 2002), *Genetic Local Search* (Aarts, 1997) and the *general procedure of MA* (Elbeltagi *et al*., 2005).

Motivated by the performance of the MA that incorporated the *adding and subtracting Improvement First Strategy* LS (Krasnogor & Smith, 2005), it is combined with the NSC algorithm in the development of a hybrid approach for finding optimal threshold values in NSC automatically. The basic concepts of NSC (Tibshirani *et al*., 2002) and GA have been reviewed in Section 2.4.3 and Section 2.4.4.1, respectively. The following sections describe the NSC-MA approach.

6.1.1. NSC-MA proposed approach

Similar to the NSC-GA approach proposed in Chapter 5, the proposed approach, NSC-MA, consists of 2 major steps:

Step 1: This step involved the automatic calculation of Th_{max}. This procedure is performed once only at the beginning of the proposed approach, NSC-MA, to obtain $Th_{max.}$

Step 2: MA (Albrechtsson *et al*., 2001) is employed in this step as an optimization method to search for optimal sets of shrinkage thresholds for NSC algorithm that lead to the selection of optimal sets of features. Also in this step, NSC algorithm is employed as a fitness evaluator to evaluate the fitness of each chromosome in terms of the number of selected features and its corresponding training classification accuracy.

The framework of the proposed approach, NSC-MA, is illustrated in Figure 6-1 and is described in the following section.

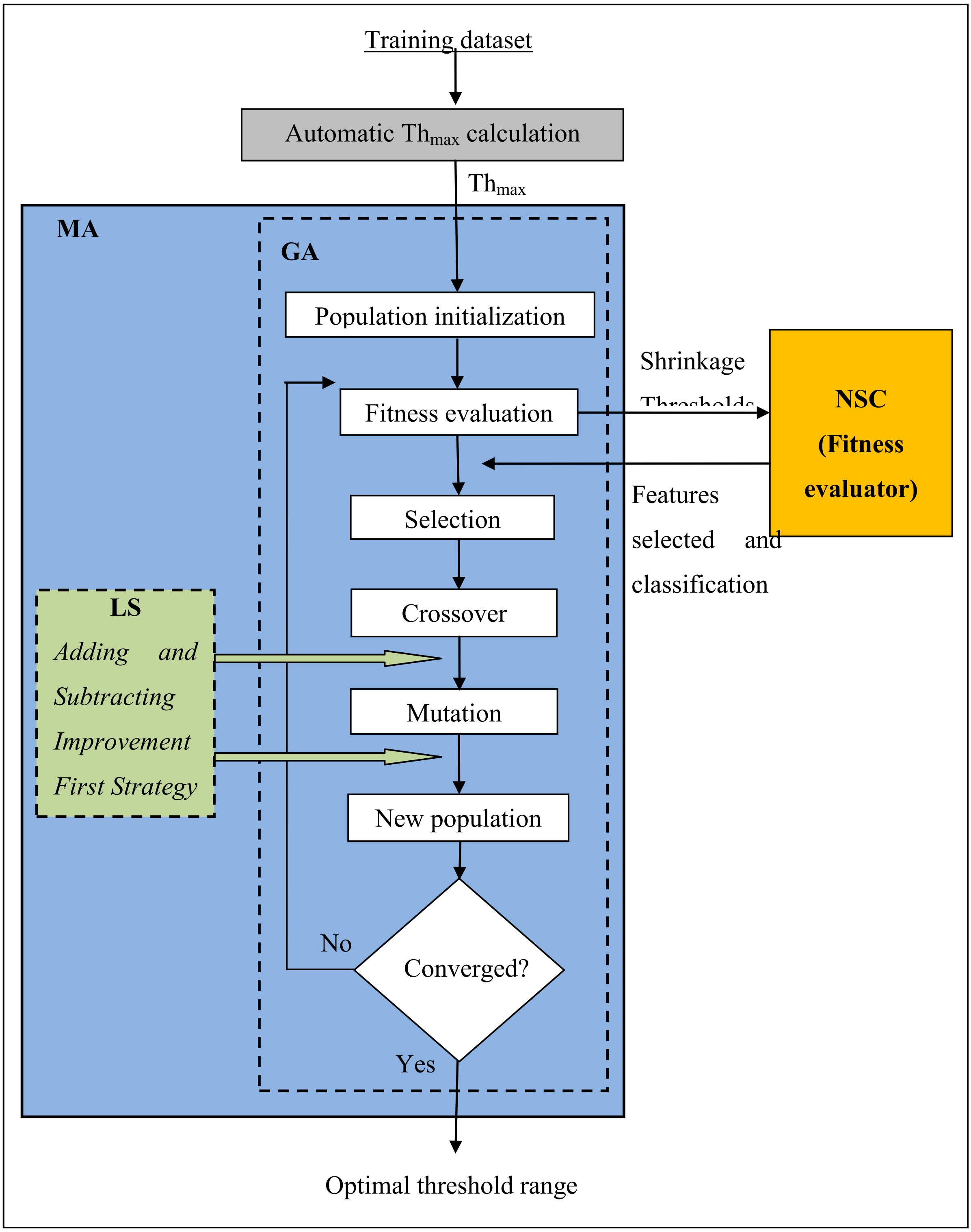


Figure 6-1 Framework of the proposed approach, NSC-MA, using MA with adding and subtracting *Improvement First Strategy* LS

The same concepts associated with chromosome encoding, estimation of the initial range of values for the shrinkage threshold and fitness evaluation as previously discussed in NSC-GA in Chapter 5 also applies in NSC-MA.

6.1.2. Steps of the proposed approach, NSC-MA

In examining Figure 6-1 and Figure 5-2, it can be seen that the only difference between NSC-GA and NSC-MA is an additional component for MA, that is, the incorporation of LS into the GA thus converting the GA into a MA. Since the core components of the algorithm are essentially steps associated with the GA, many of these have been discussed in Chapter 5 and are applicable here. These include the calculation of Th_{max} and some of the steps associated with the GA (i.e. population initialization, fitness evaluation, selection, crossover, mutation). The step "new population generation" used in NSC-GA are also used in NSC-MA, but has an addition, the incorporation of the "*adding and subtracting LS with Improvement First Strategy*". This additional step is applied to offspring chromosomes after crossover and mutation to further improve the quality of chromosome. Figure 6-2 describes the procedure of *adding and subtracting LS* with *Improvement First Strategy*.

```
Input:
    Chromosome (chrom)
    Chromosome length (len)

Output:
    An improved local search chromosome ($chrom_{ls}$)
Steps:
    1. Generate a real random number ($R_n$) in the range [0, 1] using RNG
    2. Evaluate fitness of chrom
    3. For counter from 1 to len
        a. Add $R_n$ to chrom[counter] to create a new chromosome ($chrom_{ls}$)
        b. Evaluate the fitness of $chrom_{ls}$
        c. If fitness of $chrom_{ls}$ > chrom
            • Retain $chrom_{ls}$ as an improved local search chromosome
            • Exit the loop
        d. Else
            • subtract $R_n$ to chrom[counter] create a new chromosome
              ($chrom_{ls}$)
            • evaluate the fitness of $chrom_{ls}$
            • If fitness of $chrom_{ls}$ > chrom
                o retain $chrom_{ls}$ as an improved local search
                  chromosome
                o exit the loop
            • else
                o discard $chrom_{ls}$
```

Figure 6-2 Procedure of *adding and subtracting LS with Improvement First Strategy*

The procedure for generating a new population in NSC-MA is described in Figure 6-3. A single elitist strategy is also employed in this study. The best candidate solution (*elite*) from the previous generation is retained and placed into the new generation to improve the search in evolutionary algorithms (Ahn & Ramakrishna, 2010). Also in the step of generating a new population, two best offspring chromosomes produced from the previous steps via selection, crossover, mutation and LS strategy are placed into the new population. These steps are repeated until the generation of the new population is completed.

```
Input:
        Chromosome population ($p$)
        Fitness population ($F_p$)
        Crossover probability ($P_c$)
        Mutation probability ($P_m$)
        Elite chromosome (Elite)
        Chromosome length ($len_C$)
Output:
        New population ($N_p$)
Steps:
  1. Set Size = size of population, $p$
  2. Set new population ($N_p$) = {Ø}
  3. Store Elite into $N_p$
  4. For counter from 1 to ½ Size
        a. Select 2 parent chromosomes using binary tournament selection
              i. Select 2 chromosomes randomly from $p$
                 • Select the best fit chromosome as 1st parent ($parent_1$)
              ii. Select 2 chromosomes randomly from $p$
                 • Select the best fit chromosome as 2nd parent ($parent_2$)
        b. Create 2 offspring chromosomes using $parent_1$ and $parent_2$
              i. Generate a random number ($R_n$) in the range [0, 1] using RNG
              ii. If  $R_n \leq P_c$
                 • Perform one point crossover on 2 parents to produce
                   $offspring_1$ and $offspring_2$
                 • Perform adding and subtracting LS with Improvement First
                   Strategy on $offspring_1$ and $offspring_2$ to produce 2 new
                   offspring ($offspring_1ls_{cross}$ and $offspring_2ls_{cross}$)
              iii. If  $R_n \leq P_m$
                   For counter from 1 to $len_C$
                      • Generate a random number ($R_n$) in the range [0, 1] using
                        RNG
                      If $R_n \leq P_m$
                      • Perform uniform mutation on each bit of $offspring_1$ to
                        generate $offspring_{1mut}$
                      • Perform uniform mutation on each bit of $offspring_2$ to
                        generate $offspring_{2mut}$
                      • Perform adding and subtracting LS with Improvement
                        First Strategy on $offspring_{1mut}$ and $offspring_{2mut}$ to produce
                        2 new offspring ($offspring_1ls_{mut}$ and $offspring_2ls_{mut}$)
              iv. Evaluate fitness of $offspring_1ls_{cross},$ $offspring_2ls_{cross},$
                  $offspring_1ls_{mut}$ and $offspring_2ls_{mut}$ chromosomes
        c. Store the best 2 chromosomes into $N_p$
```

Figure 6-3 Algorithm for generating a new population using MA incorporated *adding and subtracting LS with Improvement First Strategy*

6.1.3. Parameter settings

The parameter settings for running NSC-MA are shown in Table 6-1. The parameters used here are the same as those used in NSC-GA (described in Chapter 5), except for an additional parameter "Local Search".

Table 6-1 Parameter settings used in the proposed approach, NSC-MA

Parameters	Values/Algorithm
Population size	30
Chromosome length - Real encoding	10
Crossover rate	0.6
Mutation rate	0.033
Maximum generation	1000
Selection	Tournament
Crossover	Single point
Mutation	Uniform
Elitist	Single
Local search	*Adding and subtracting with First Improvement Strategy*

6.2. Experiment results

Similar to the experiments for the NSC-GA approach described in Chapter 5, NSC-MA was evaluated using seven datasets: AD, Colon, Leukemia, Ovarian, Lymphoma, Lung and Prostate cancer datasets as described in Section 3.1. For each dataset, 15 independent runs of NSC-MA were executed using the respective training data and parameter values shown in Table 6-1. For each run, 10 fold CV strategy described in Section 3.2 was employed to evaluate the selected feature sets. The optimal set of features was then used to construct the NSC classifier to classify the unseen test data associated with the dataset. The classification results for classifying the unseen test data were recorded and the average classification result from 15 independent runs was calculated. The following sections detail the results obtained from applying the approach on each of the seven datasets. Where appropriate, the comparison of the performance of the proposed algorithm with existing work is based on classification accuracy and the selected feature sets.

6.2.1. Ray *et al.* Alzheimer's Disease data

As mentioned previously in Section 3.1.1, this dataset consists of 120 attributes. The training set consists of 43 AD and 40 NDC samples and 2 test sets: the AD test set consists of 42 AD, 50 NAD samples and the MCI test set consists of 22 AD and 25 NAD samples.

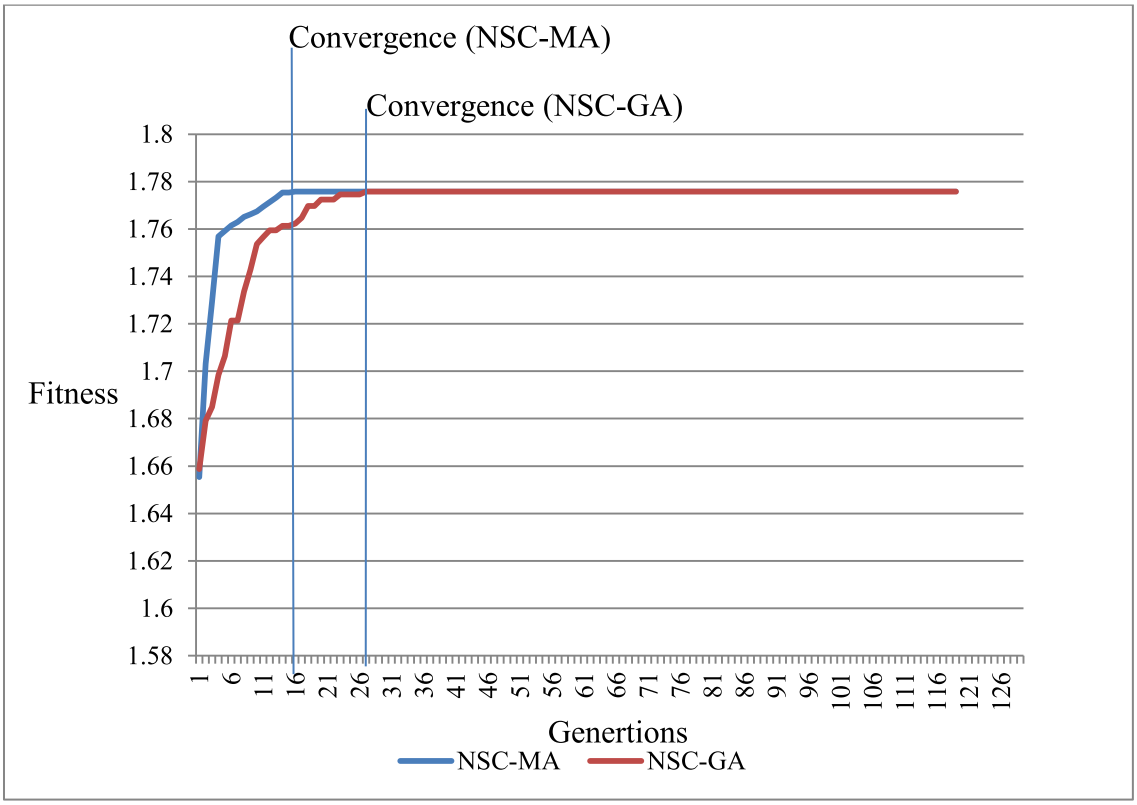


Figure 6-4 A comparison convergence of fitness plot for AD training dataset using NSC-MA and NSC-GA associated with one typical run

Figure 6-4 shows a plot of convergence of fitness associated with one typical run for NSC-MA and NSC-GA. Both algorithms converged to the global optimum with the same maximum fitness of 1.775. However, the number of generations that the algorithms have to run to achieve convergence of fitness is different. With the proposed approach, NSC-MA, convergence occurred after 14 generations, and with NSC-GA, convergence occurred after 28 generations in this case. From Table 6-2, it can be seen that on average NSC-MA takes 18 +/- 2.97 runs to converge versus NSC-GA requiring 28 +/- 4.68 runs for convergence of fitness. Therefore the NSC-MA takes less computational time to obtain the same global optimum as NSC-GA. This is due to the fact that, with NSC-MA, chromosomes in the population have been subjected to the local search to further improve the fitness in each generation and subsequently, the optimal fitness is obtained in a shorter time.

The same set of 11 features is obtained from 15 independent runs using NSC-MA. Classifier constructed from this set of features gave an average classification accuracy of 89.34% for the unseen AD test dataset and 76.59% for the unseen MCI test dataset, compared to 89.49% and 79%, respectively, using NSC-GA. Although the same set of 11 features was obtained using the proposed approach, the resulting classification accuracy is slightly different from the value obtained using NSC-GA. This is due to the fact that the optimal shrinkage threshold values obtained from NSC-MA are only slightly different from those using NSC-GA. The nature of shrinkage thresholds associated with NSC is that rather than an exact value, a narrow range of values maps to the same set of features. Since the optimal threshold value from NSC-GA and NSC-MA is only slightly different, both mapped to the same set of 11 features but still produced slight differences in classification accuracy. The classification results of NSC-MA in comparison to NSC-GA are shown in Table 6-2.

Table 6-2 Classification results and time to converge for NSC-GA and NSC-MA using AD data

Approach	Alzheimer			Average number of generations for convergence of fitness over 15 independent runs	Standard deviation (Stdev)
	AD		MCI		
	No of attributes	Unseen test data (%)	Unseen test data (%)		
Proposed approach NSC-MA	11	89.34	76.59	18	2.97
NSC-GA	11	89.49	79	28	4.68

6.2.2. Alon *et al.* Colon cancer data

Details about the Colon dataset can be found in Section 3.1.

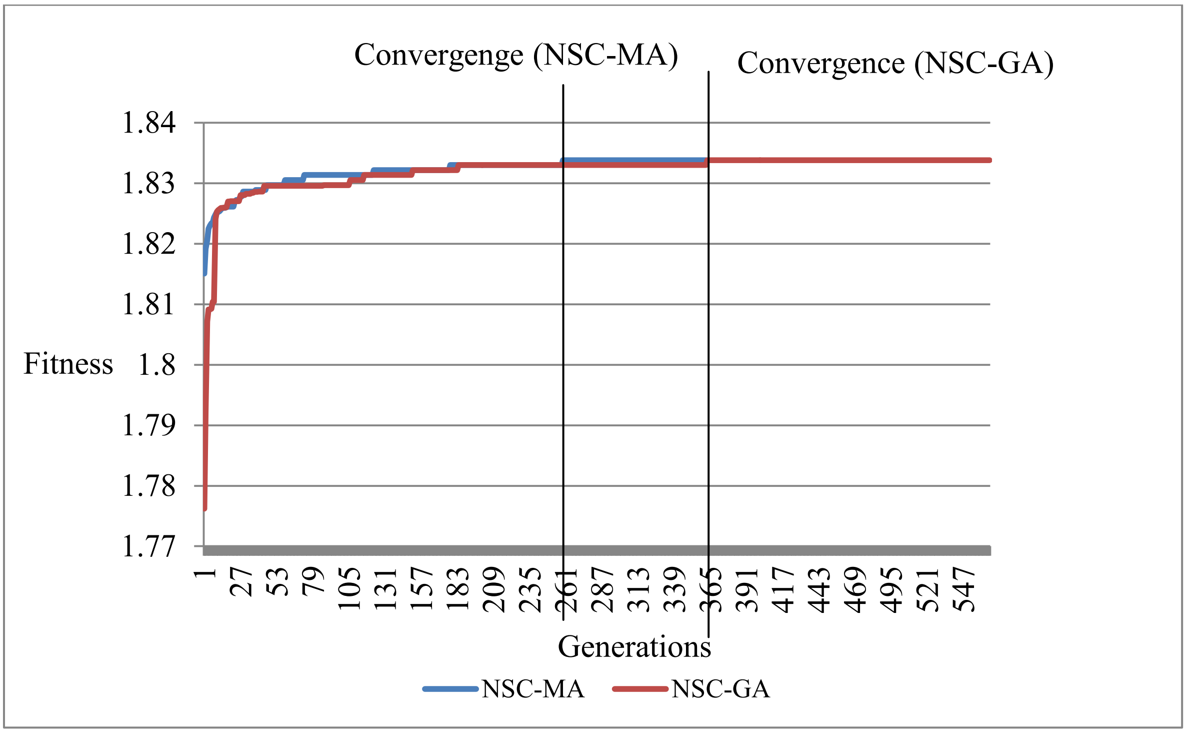


Figure 6-5 Plots for convergence of fitness from a typical run for Colon training dataset using NSC-MA and NSC-GA

As seen in Figure 6-5, both algorithms converged to the global optimum with the same maximum fitness of 1.883. However, the number of generations that the algorithms have to run to achieve convergence of fitness is different. With NSC-MA convergence occurred after 259 generations, and with NSC-GA, convergence occurred after 363 generations in this sample run. From Table 6-3, it can be seen that on average NSC-MA takes 274 +/- 178.84 runs to converge versus NSC-GA requiring 309 +/- 194. 98 runs for convergence of fitness. The same set of 28 features is obtained from each of 15 independent runs using NSC-MA and resulted in an average classification accuracy of 100% for the unseen test cancer dataset. In comparison, sets of 6 and 28 features were obtained using NSC-GA with 93.75% and 100% for average classification accuracy on the same unseen test dataset, respectively. This shows that the proposed approach NSC-MA selects sets of features consistently for all 15 independent runs (i.e. the same set of 28 features is obtained for every run) compared to NSC-GA where 2 sets, one of 6 and one of 28 features were obtained from 15 runs. The classification results of NSC-MA in comparison to those associated with NSC-GA are shown in Table 6-3.

Table 6-3 Classification results and time to converge for NSC-GA and NSC-MA using the Colon cancer data

Approach	Colon			
	No of attributes	Unseen test data (%)	Average number of generations for convergence of fitness over 15 independent runs	Standard deviation (Stdev)
Proposed approach NSC-MA	28	100	271	178.84
NSC-GA	28 6	100 93.75	309	194.89

6.2.3. Leukemia cancer data

Details about the Leukemia dataset can be found in Section 3.1.

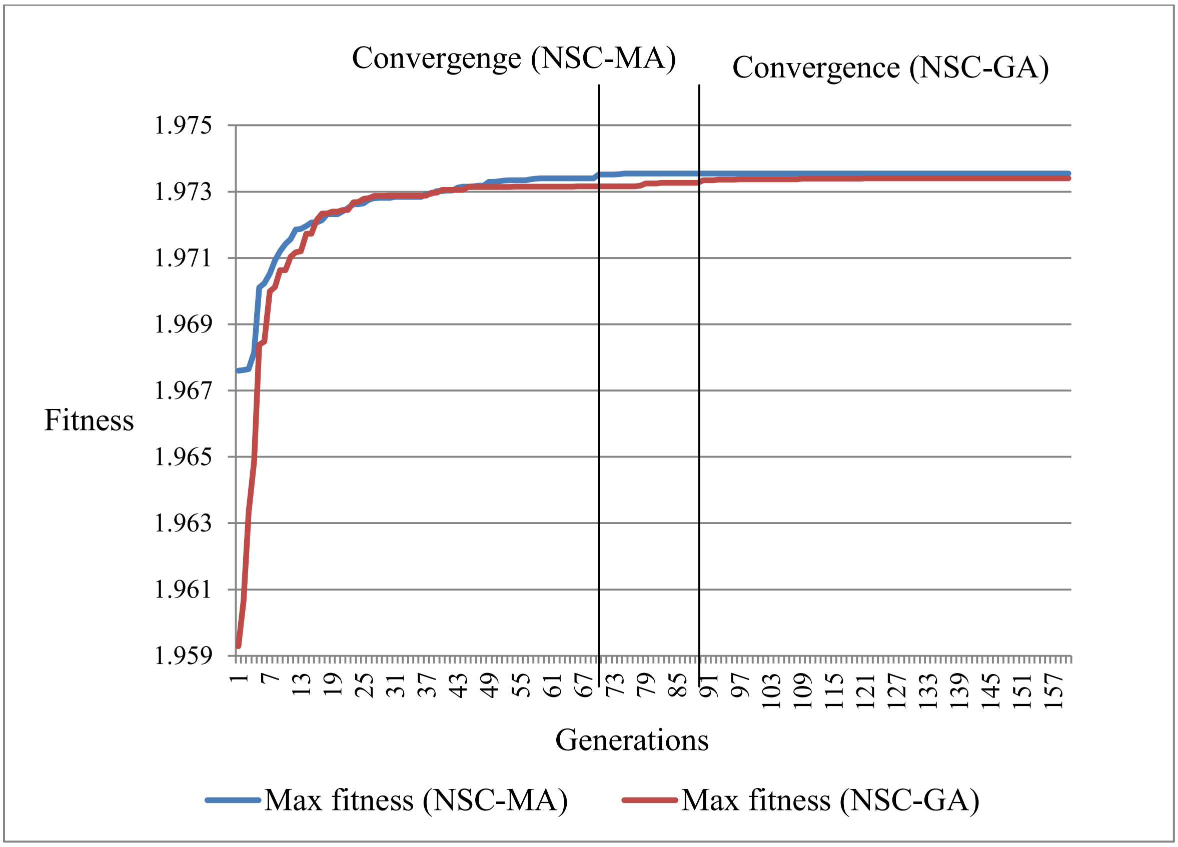


Figure 6-6 Plots for convergence of fitness from a typical run for Leukemia training dataset using NSC-MA and NSC-GA

As seen in Figure 6-6, both algorithms converged to the global optimum with the same maximum fitness of 1.973. In terms of convergence of fitness, NSC-MA took 73 generations, and NSC-GA took 91 generations for this sample run. From Table 6-4, it can be seen that on average NSC-MA takes 54 +/- 38.73 runs to converge versus NSC-GA requiring 82 +/- 43.57 runs for convergence of fitness. The same set of 9 features is obtained from each of the 15 independent runs using NSC-MA, resulting in an average classification accuracy of 97.05% on the Leukemia unseen test dataset. The classification results of NSC-MA in comparison to those of NSC-GA are shown in Table 6-4.

Table 6-4 Classification results and time to converge for NSC-GA and NSC-MA using the Leukemia cancer data

Approach	Leukemia			
	No of attributes	Unseen test data (%)	Average number of generations for convergence of fitness over 15 independent runs	Standard deviation (Stdev)
Proposed approach NSC-MA	9	97.05	54	38.73
NSC-GA	9	97.05	82	43.57

6.2.4. Ovarian cancer data

Details about this dataset can be found in Section 3.1.

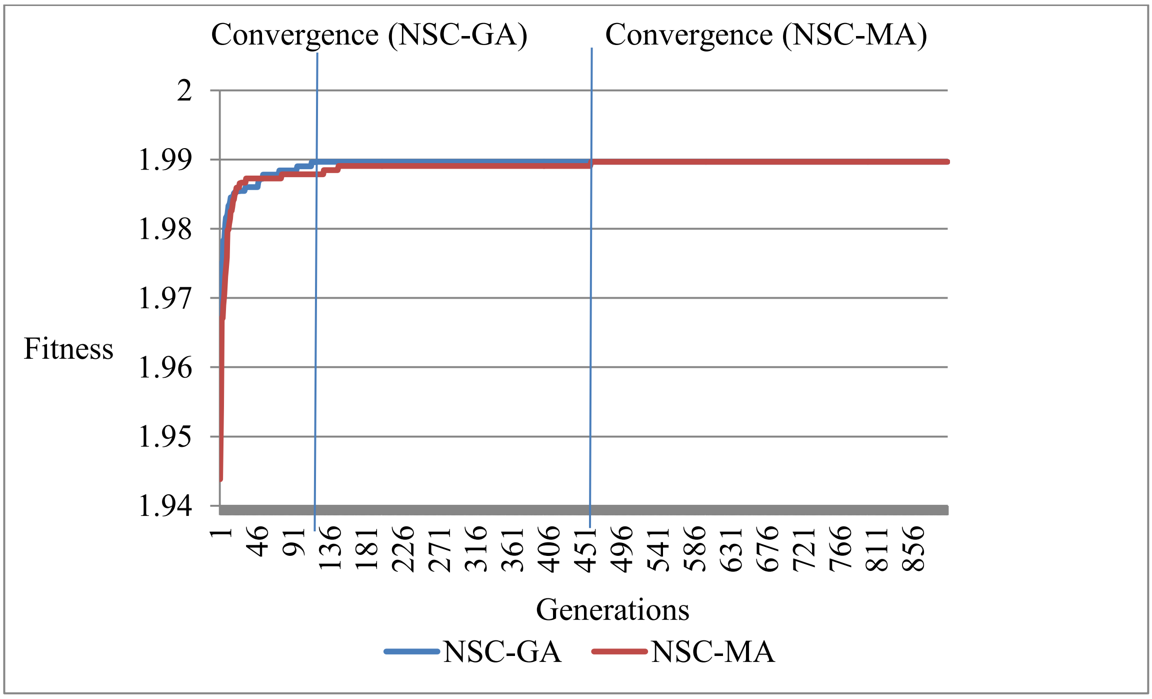


Figure 6-7 Plots for convergence of fitness from a typical run for Ovarian training dataset using NSC-MA and NSC-GA

As seen in Figure 6-7, both algorithms converged to the global optimum with the same maximum fitness of 1.99. With NSC-MA, convergence occurred after 452 generations, and with NSC-GA, convergence occurred after 124 generations in this sample run. From Table 6-5, it can be seen that on average NSC-MA takes 177 +/- 160.78 runs to converge versus NSC-GA requiring 86.88 +/- 24.5 obvious from runs for convergence of fitness. NSC-MA found the same set of 7 peptides resulting in the same average classification accuracy of 96.06% on the Ovarian cancer unseen test dataset as that obtained via NSC-GA. The classification results of NSC-MA in comparison to those of NSC-GA are shown in Table 6-5.

Table 6-5 Classification results and time to converge for NSC-GA and NSC-MA using the Ovarian cancer data

Approach	Ovarian			
	No of attributes	Unseen test data (%)	Average number of generations for convergence of fitness over 15 independent runs	Standard deviation (Stdev)
Proposed approach NSC-MA	7	96.06	177	160.78
NSC-GA	7	96.06	86.88	24.5

6.2.5. Lymphoma cancer data

Details about this dataset can be found in Section 3.1.

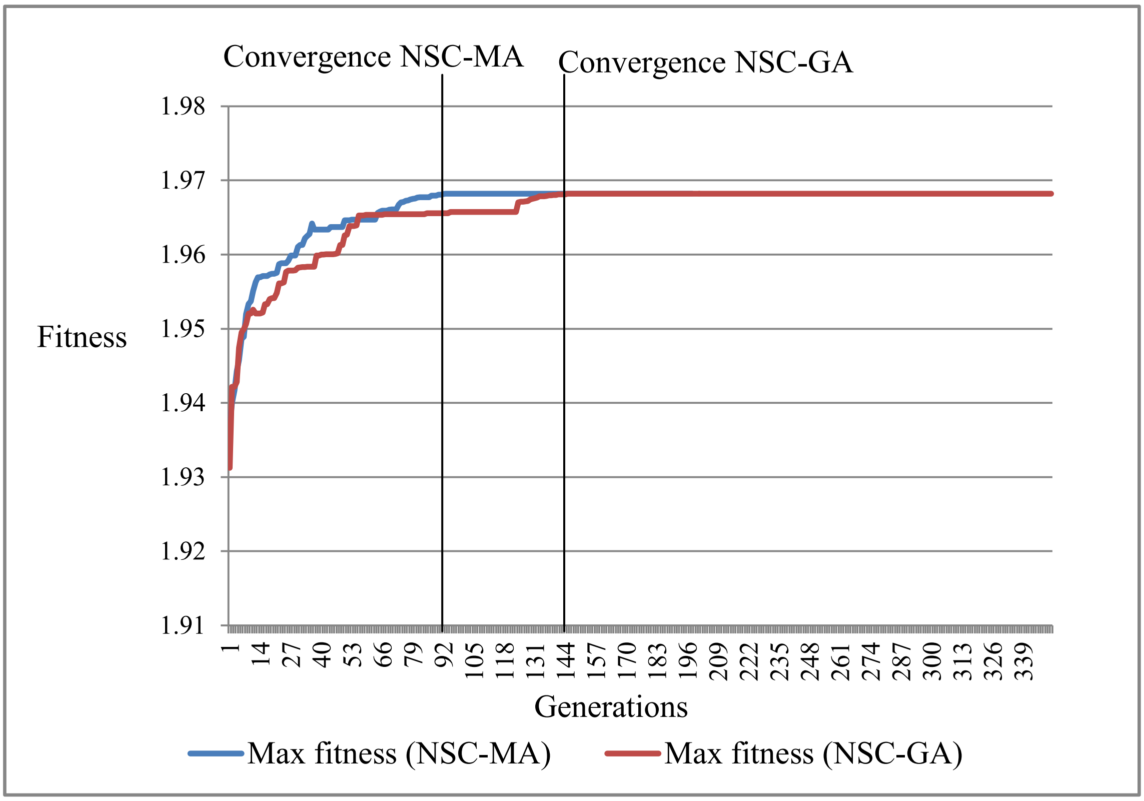


Figure 6-8 Plots for convergence of fitness from a typical run for Lymphoma training dataset using NSC-MA and NSC-GA

As seen in Figure 6-8, both algorithms converged to the global optimum with the same maximum fitness of 1.968. With NSC-MA, convergence occurred after 92 generations, and with NSC-GA, convergence occurred after 144 generations in this typical run. From Table 6-6, it can be seen that on average NSC-MA takes 88 +/- 8.91 runs to converge versus NSC-GA requiring 100 +/- 62.42 runs for convergence of fitness. The same set of 128 features obtained for each of the 15 independent runs using NSC-MA gave the same average classification accuracy of 100% on the Lymphoma unseen test dataset. This shows that the proposed approach NSC-MA selects features consistently for all 15 independent runs (i.e. the same set of 128 features is obtained for every run) compared to the approach NSC-GA where 5 different sets of 128, 129, 132, 7 and 12 features were selected from the independent 15 runs. The classification results of NSC-MA in comparison to NSC-GA are shown in Table 6-6.

Table 6-6 Classification results and time to converge for NSC-GA and NSC-MA using the Lymphoma cancer data

Approach	Lymphoma			
	No of attributes	Unseen test data (%)	Average number of generations for convergence of fitness over 15 independent runs	Standard deviation (Stdev)
Proposed approach NSC-MA	128	100	88	8.91
NSC-GA	7 12 128 129 132	95.45 95.45 100 100 100	100	62.42

6.2.6. Lung cancer data

Details about this dataset can be found in Section 3.1.

As seen in Figure 6-9, both algorithms converged to the global optimum with the same maximum fitness of 1.999. With NSC-MA, convergence occurred after 27 generations, and with NSC-GA, convergence occurred after 88 generations in this sample run. From Table 6-7, it can be seen that on average NSC-MA takes 41 +/- 9.5 runs to converge versus NSC-GA requiring 68 +/- 53.08 runs for convergence of fitness. The same set of 8 features obtained from each of the 15 independent runs using NSC-MA gave the same average classification accuracy of 100% on the Lung unseen test dataset. In comparison, NSC-GA produced 4 different sets consisting of 8, 9, 10, and 11 features from 15 runs. The classification results of NSC-MA in comparison to NSC-GA are shown in Table 6-7.

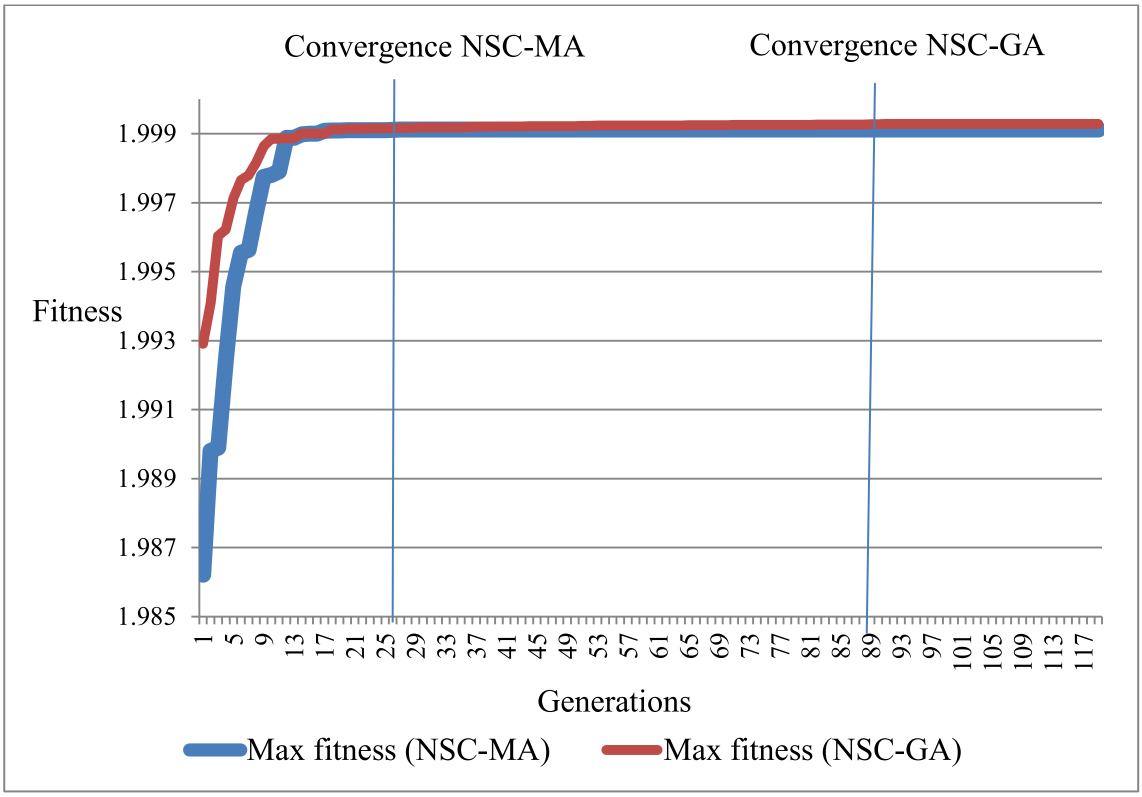


Figure 6-9 Plots for convergence of fitness from a typical run for Lung training dataset using NSC-MA and NSC-GA

Table 6-7 Classification results and time to converge for NSC-GA and NSC-MA using the Lung cancer data

Approach	Lung			
	No of attributes	Unseen test data (%)	Average number of generations for convergence of fitness over 15 independent runs	Standard deviation (Stdev)
Proposed approach NSC-MA	8	100	41	9.5
NSC-GA	8 9 10 11	100	68	53.08

6.2.7. Prostate cancer data

Details about this dataset can be found in Section 3.1.

As seen in Figure 6-10, both algorithms converged to the global optimum with the same maximum fitness of 1.973. With NSC-MA, convergence occurred after 62 generations, and with NSC-GA, convergence occurred after 99 generations in this sample run. From Table 6-8, it can be seen that on average NSC-MA takes 65 +/- 23.62 runs to converge versus NSC-GA requiring 82 +/- 32.26 runs for convergence of fitness. The same set of 6 features obtained from each of the 15 independent runs using NSC-MA gave the same average classification accuracy of 90.2% on the Prostate unseen test dataset. The classification results of NSC-MA in comparison to NSC-GA are shown in Table 6-8.

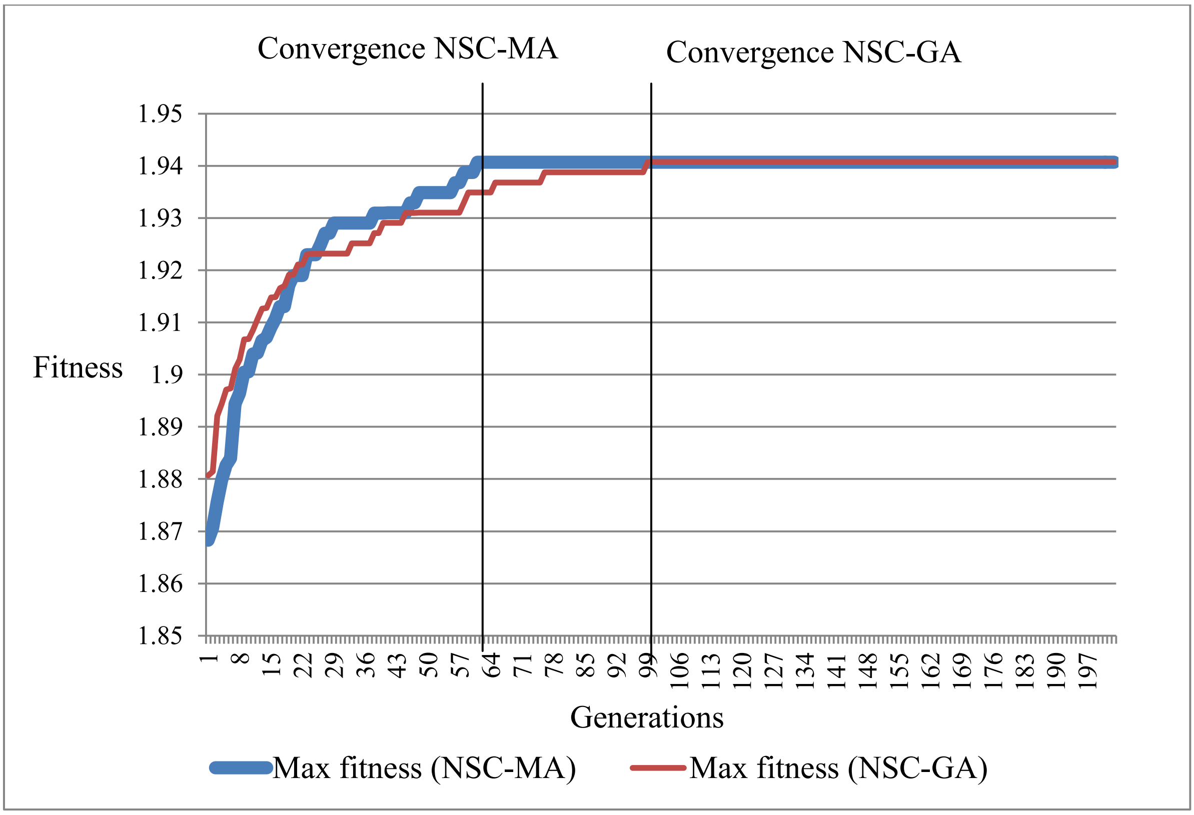


Figure 6-10 A comparison convergence of fitness plot for Prostate training dataset using NSC-MA and NSC-GA associated with one typical run

Table 6-8 Classification results and time to converge for NSC-GA and NSC-MA using the Prostate cancer data

Approach	Prostate			
	No of attributes	Unseen test data (%)	Average number of generations for convergence of fitness over 15 independent runs	Standard deviation (Stdev)
Proposed approach NSC-MA	6	90.2	65	23.62
NSC-GA	6	90.2	80	32.26

Table 6-9 Summary of results obtained from the NSC-GA and NSC-MA approach

Dataset	NSC-GA				NSC-MA			
	No of attributes	Unseen test data (%)	Average number of generations for convergence of fitness over 15 independent runs	Standard deviation (Stdev)	No of attributes	Unseen test data (%)	Average number of generations for convergence of fitness over 15 independent runs	Standard deviation (Stdev)
AD	11	89.49	28	2.97	11	89.34	18	4.68
Colon	28 6	100 93.75	309	194.9	28	100	271	178.84
Leukemia	9	97.05	82	43.57	9	97.05	54	38.73
Lymphoma	7 12 128 129 132	95.45 95.45 100 100 100	100	62.42	128	100	88	8.91
Ovarian	7	96.06	86	24.5	7	96.06	177	160.78
Lung	8 9 10 11	100	68	53.08	8	100	41	9.5
Prostate	6	90.2	80	32.26	6	90.2	65	23.62

6.3. *Discussion*

Table 6-9 shows a summary of experimental results obtained using NSC-MA, in comparison to results from NSC-GA on the same datasets. The proposed approach achieved the same classification results in terms of the sets of selected features and classification accuracy on unseen test sets for the seven datasets. However, the NSC-MA approach has generally improved performance in terms of computational time for all datasets except for the Ovarian and Lymphoma datasets. That is, the NSC-MA approach, on average over 15 independent runs, required a smaller number of generations for convergence of fitness.

Another difference from NSC-MA when compared to NSC-GA is that, NSC-MA consistently obtained the same set of features for each of the 15 independent runs. This highlights the advantage of incorporating LS into the previous approach NSC-GA to further improve the fitness of candidate solutions and subsequently, that leads to only one constant optimal solution being obtained for all the independent runs for the respective dataset. For example, for the Lung cancer dataset, the proposed approach NSC-MA obtained the same set of 8 genes for each of the 15 independent runs, whilst the NSC-GA approach obtained sets with 8, 9, 10, and 11 features from the 15 independent runs.

Overall, the impact of incorporating MA with NSC for finding shrinkage threshold values automatically are (1) reduced computational time and (2) obtaining the same feature set over different runs of NSC-MA.

6.4. *Summary*

This chapter has described the proposed approach of incorporating the NSC and MA to automatically search for optimal threshold values for the NSC, and subsequently to be used for FS and classification. The approach incorporated the adding and subtracting LS with *Improvement First Strategy* in a MA to optimize the optimal threshold value automatically for NSC, and to obtain the sets of relevant features. The results obtained shows that with NSC-MA, convergence of fitness is quicker while obtaining the same feature set and similar classification accuracy, compared to those obtained via NSC-GA.

In Chapter 7, the investigation of incorporating different similarity distance measures into the NSC algorithm in NSC-GA will be described.

7. Incorporating different similarity distance measures into the NSC algorithm in the NSC-GA approach

The NSC classifier uses Euclidean distance as a measure to assign data points to different classes. Each data point is assigned to the class that it has the shortest Euclidean distance. The classification accuracy is calculated based on correct class assignment. The higher the number of data points correctly assigned to its class, the higher the resulting classification accuracy. According to Bandyopadhyay and Saha (2013, p. 60), "*similarity measurement is essential for performing classification*", therefore employing a different similarity distance measure in the NSC classifier would impact its class prediction of data points and consequently the classification accuracy.

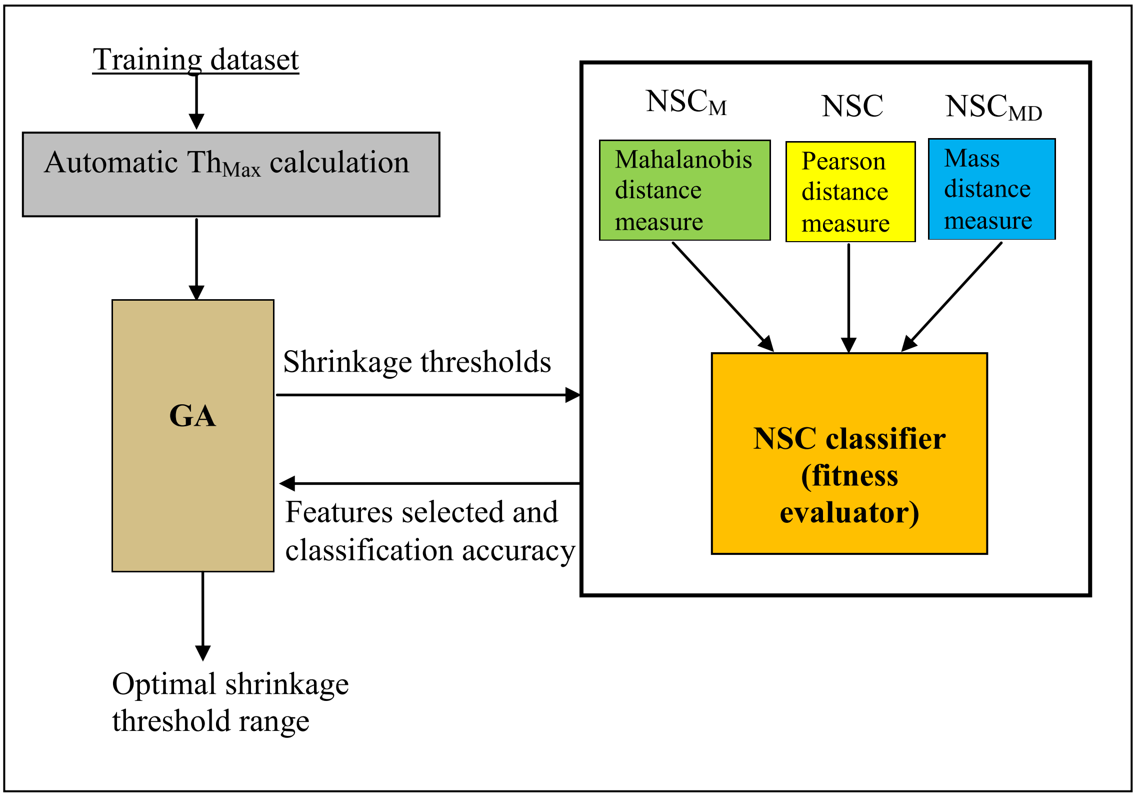
The aim of this chapter is to investigate the impact of employing different similarity distance measures (Mahalanobis, Pearson and Mass distance (MD) measure) in the NSC classifier on FS and classification using the same NSC-GA approach, which has been proposed and implemented in Chapter 5. Subsequently the impact of incorporating different distance measures in NSC$_*$-GA (with * representing M or P or MD) is evaluated using the seven biomedical datasets described in Chapter 3. Section 7.1.1 describes the proposed approaches, NSC$_M$-GA, NSC$_P$-GA and NSC$_{MD}$-GA, the results are reported in Section 7.2, discussion is in Section 7.3 and summary is in Section 7.4.

7.1. NSC*-GA proposed approach

Similar to the NSC-GA approach where Euclidean distance is employed in the NSC, the proposed approach NSC$_*$-GA consists of the same 2 major steps that have been described in Section 5.2.

Considerations for chromosomes encoding, estimation of the initial range of values for the shrinkage threshold and fitness evaluation in NSC*-GA are also the same as NSC-GA and have been discussed in Section 5.2.1. The basic concepts of NSC algorithm (Tibshirani *et al.*, 2002) have been reviewed in Section 2.3.3 and different distance measures, Euclidean, Mahalanobis (Mahalanobis, 1936), Pearson (Pearson, 1895) and MD (Yona *et al.*, 2006) have been reviewed in Section 2.3.6. The following sections describe the proposed approach which incorporates NSC with different similarity distance measures, Mahalanobis, Pearson and MD, into GA, denoted as NSC_M-GA, NSC_P-GA and NSC_{MD}-GA, respectively.

7.1.1. NSC-GA with Mahalanobis (NSC_M-GA), Pearson (NSC_P-GA) and Mass distance measure (NSC_{MD}-GA)



Training dataset
Automatic Th_{Max} calculation
GA
Shrinkage thresholds
Features selected and classification accuracy
Optimal shrinkage threshold range
NSC_M
NSC
NSC_{MD}
Mahalanobis distance measure
Pearson distance measure
Mass distance measure
NSC classifier (fitness evaluator)

Figure 7-1 Framework of NSC*-GA

As seen in Figure 7-1, the core of the framework of NSC*-GA is the same as of the framework of NSC-GA. The only difference between the 2 approaches is the different distance measure method employed in the NSC classifier. That is, instead of using the Euclidean distance as a distance measure in the original implementation of NSC, Mahalanobis distance (green box), Pearson (yellow box) and Mass distance (blue box) are employed in the NSC classifier to measure the distance between data points and classes when performing classification. The modified NSC* classifiers (NSC with a different distance measure) are NSC_M, NSC_P and NSC_{MD} with M, P and MD denoting Mahalanobis distance, Pearson and Mass distance respectively. These modified classifiers are detailed in the following sections.

A number of calculations in the original NSC algorithm still apply in NSC$_M$, NSC$_P$ and NSC$_{MD.}$ These include the calculations of class centroid, overall centroid, relative difference, shrunken relative difference, updated class centroid and classification. Equations for these calculations are shown in Equation (7.1), (7.2), (7.3), (7.4), (7.5), and (7.6), respectively. More details about these equations can be found in Section 2.4.3.

- Class centroid : $\bar{x}_{ik} = \sum_{j\in C_k} x_{ij} \,/\, n_k$ (7.1)
- Overall centroid: $\bar{x}_i = \sum_{j=1}^{n} x_{ij} \,/n$ (7.2)
- Relative difference: $d_{ik} = \frac{\bar{x}_{ik}\text{-}\bar{x}_i}{m_k(s_i+s_o)}$ (7.3)
- Shrunken relative difference: $d'_{ik} = sign\ (d_{ik})(|d_{ik}| - \Delta)$ if $|d_{ik}| > \Delta$ Otherwise 0 (7.4)
- Updated class centroid: $\bar{x}'_{ik} = \bar{x}_i + m_k\ (s_i + s_o)\ d'_{ik}$ (7.5)
- Classification: $C(x^*) = \ell$ if $\delta_\ell(x^*) = \min_k\ \delta_k(x^*)$ (7.6)

The following sections describe the specific distance measure employed in NSC$_M$, NSC$_P$ and NSC$_{MD}$ classifiers respectively.

➢ NSC$_M$ classifier

The difference between the NSC and NSC$_M$ classifier is the calculation of discriminant scores for data points, where each data point is assigned to the class that it has the closest distance, i.e., the distance is based on a minimal discriminant score.

In NSC$_M$, the calculation of the discriminant score is now obtained using the calculation of Mahalanobis distance ($Dist_M$) as defined by Equation (7.8) and (7.9). Descriptions of Mahalanobis distance and associated equations are found in Section 2.3.6.2.

$$Dist_M = \sqrt{(\boldsymbol{x} - \mu)^{\mathrm{T}}\ \Sigma^{-1}(\boldsymbol{x} - \mu)} \quad (7.7)$$

where x is a data point

$\boldsymbol{\mu}$ is a class centroid

superfix T is a matrix transpose

Σ^{-1} is an inverse covariance matrix

Hence the calculation of discriminant scores using Mahalanobis distance measure is defined by Equation (7.8) as follows.

$$\delta_k(x^*) = Dist_M \tag{7.8}$$

- NSC$_P$ classifier

In NSC$_P$, the calculation of the discriminant score is now obtained using Pearson Correlation. These calculations are defined by Equation (7.9), (7.10) and (7.11). Descriptions of Pearson Correlation and associated equations are found in Section 2.3.6.3.

- Correlation coefficient $$r = \frac{\Sigma(x-\bar{X})(y-\bar{Y})}{\sqrt{(x-\bar{X})^2}\sqrt{(y-\bar{Y})^2}} \tag{7.9}$$

where x is variable value

$\bar{X}$ *and* $\bar{Y}$ are class centroids

- Pearson correlation measure $$P_D = 1 - |r| \tag{7.10}$$

Hence the calculation of discriminant scores using Pearson correlation is defined by Equation (7.11) as follows.

$$\delta_k(x^*) = P_D \tag{7.11}$$

- NSC$_{MD}$ classifier

In NSC$_{MD,}$ the calculation of the discriminant score is now obtained using MD. These calculation are defined by Equation (7.12), (7.13), 7.14), (7.15) and (7.16) in the calculations for Mass Distance. These equations have been described in Section 2.3.6.4.

$$MASS(_{a_ib_i}) = \int_{\min(a_i,b_i)}^{\max(a_i,b_i)} Prob_i(x)\,dx \tag{7.12}$$

$$Prob_i(x) = \left(\frac{1}{\sigma} * \sqrt{2\pi}\right) * e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{7.13}$$

$$dx\ is\ \Delta\text{x} = \frac{max-min}{n} \tag{7.14}$$

$$MASS(_{a_ib_i}) = \left(\int_{\min(a_i,b_i)}^{\max(a_i,b_i)} \left(\frac{1}{\sigma} * \sqrt{2\pi}\right) * e^{-\frac{(x-\mu)^2}{2\sigma^2}}\right) * \left(\frac{max-min}{n}\right) \tag{7.15}$$

Hence the calculation of discriminant scores using mass distance is defined by Equation (7.16) as follows.

$$\delta_k(x^*) = MASS(_{a_ib_i}) \tag{7.16}$$

As mentioned previously, NSC∗-GA comprised of the same 2 major steps as NSC-GA. These steps include the automatic Th_{max} calculation, GA search optimization including population initialization, fitness evaluation, GA operators and new population generation steps which have been described in Section 5.2.1 and will not be repeated here. The same parameter settings used in NSC-GA (Table 5-4) are also used to run NSC_{M}-GA, NSC_{P}-GA and NSC_{MD}-GA.

7.2. Experiment results

Similar to the experiments for NSC-GA described in Chapter 5, NSC_{M}-GA, NSC_{P}-GA and NSC_{MD}-GA, were evaluated using the same seven datasets. For each dataset, 15 independent runs of NSC∗-GA (i.e. 15 runs for each of NSC_{M}-GA, NSC_{P}-GA and NSC_{MD}-GA) were executed using the respective training dataset and parameter settings shown in Table 5-4. For each independent run, 10 fold CV strategy described in Section 3.2 was employed to evaluate the proposed approaches. The optimal set of features obtained for each dataset was then used to construct the respective NSC∗ classifier to evaluate the corresponding unseen test dataset. The classification results for the unseen test data were recorded and the average classification result from 15 independent runs was calculated. The following sections report the results of experiments for each approach on each of the seven datasets. Where appropriate, the comparison of the performance of the proposed algorithm with existing work is based on classification accuracy and the selected feature sets.

7.2.1. NSC$_M$-GA

7.2.1.1. Ray *et al.* AD data

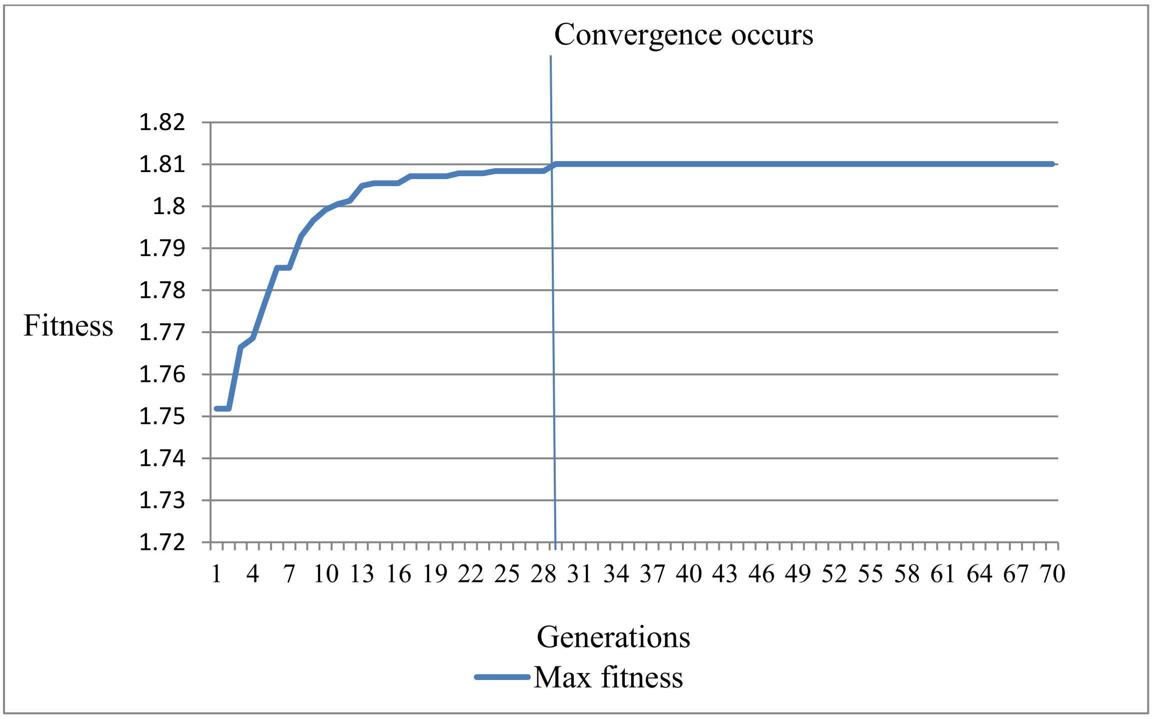


Figure 7-2 A typical plot for convergence of fitness for the training data of AD using NSC$_M$-GA

As seen in Figure 7-2, convergence in this sample run occurred after 28 generations with the maximum fitness of 1.81. The optimal shrinkage threshold obtained for each of the runs had the same maximum fitness of 1.81 which produced the set of 18 proteins with resulting 97.82% classification accuracy on the unseen test data (Figure 7-1). The 18 proteins selected by NSC$_M$-GA, for the AD dataset are the same set of 18 proteins found by Ray *et al.* (2007).

Table 7-1 Comparison of classification results between NSC-GA and NSC$_M$-GA for AD data

NSC-GA (Euclidean dist.)		Proposed NSC$_M$-GA (Mahalanobis dist.)	
Number of proteins	Unseen Test (%)	Number of proteins	Unseen Test (%)
11	90.21	18	97.82

7.2.1.2. Alon *et al.* Colon cancer data

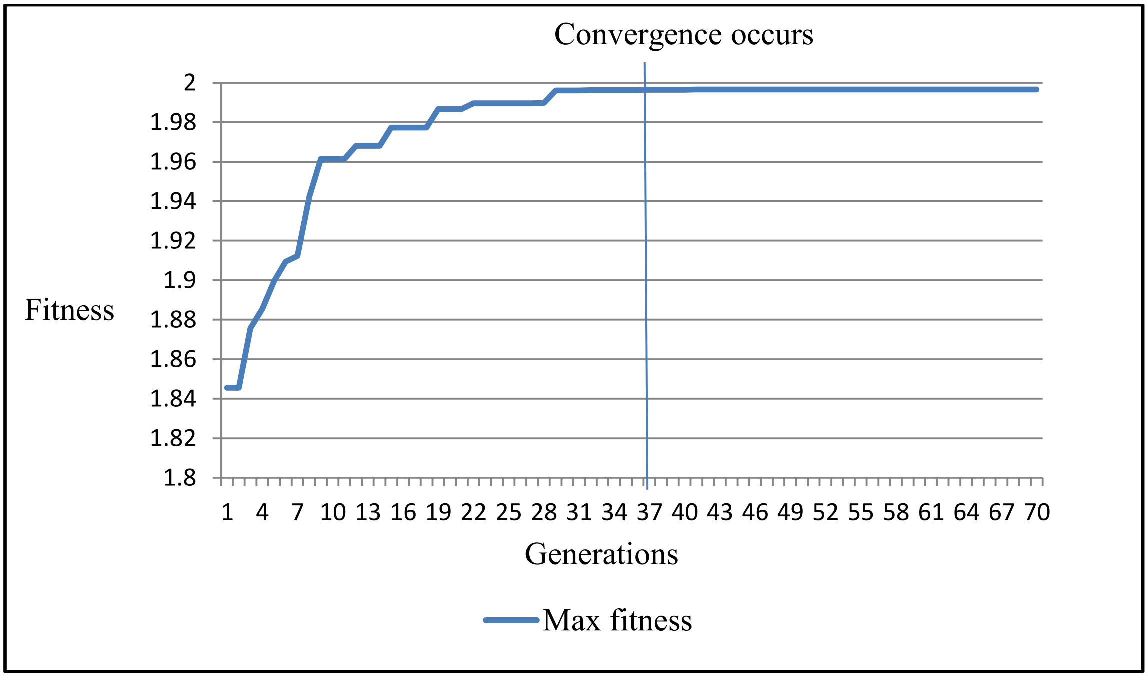


Figure 7-3 A typical plot for convergence of fitness for the training data of Colon using NSC$_M$-GA

As seen in Figure 7-3, convergence in this sample run occurred after 37 generations with the maximum fitness of 1.99. The optimal shrinkage threshold obtained for each of the runs had the same maximum fitness of 1.99 which produced the set of 7 genes with 93.54% classification accuracy on the unseen test set. The gene accession numbers of 7 genes selected by NSC$_M$-GA for Colon dataset are T71025, M76378, M63391, T92451, H64489, M76378 and J02854. This set of 7 genes is a subset of the set of 28 genes and the superset of the 6 genes which have been found by NSC-GA. As seen in Table 7-2, the classification accuracy using the set of 6 and 7 genes are very similar.

Table 7-2 Comparison of classification results between NSC-GA and NSC$_M$-GA for Colon cancer data

NSC-GA (Euclidean dist.)		Proposed NSC$_M$-GA (Mahalanobis dist.)	
Number of genes	Unseen Test (%)	Number of genes	Unseen Test (%)
28 6	100 93.75	7	93.54

7.2.1.3. Leukemia cancer data

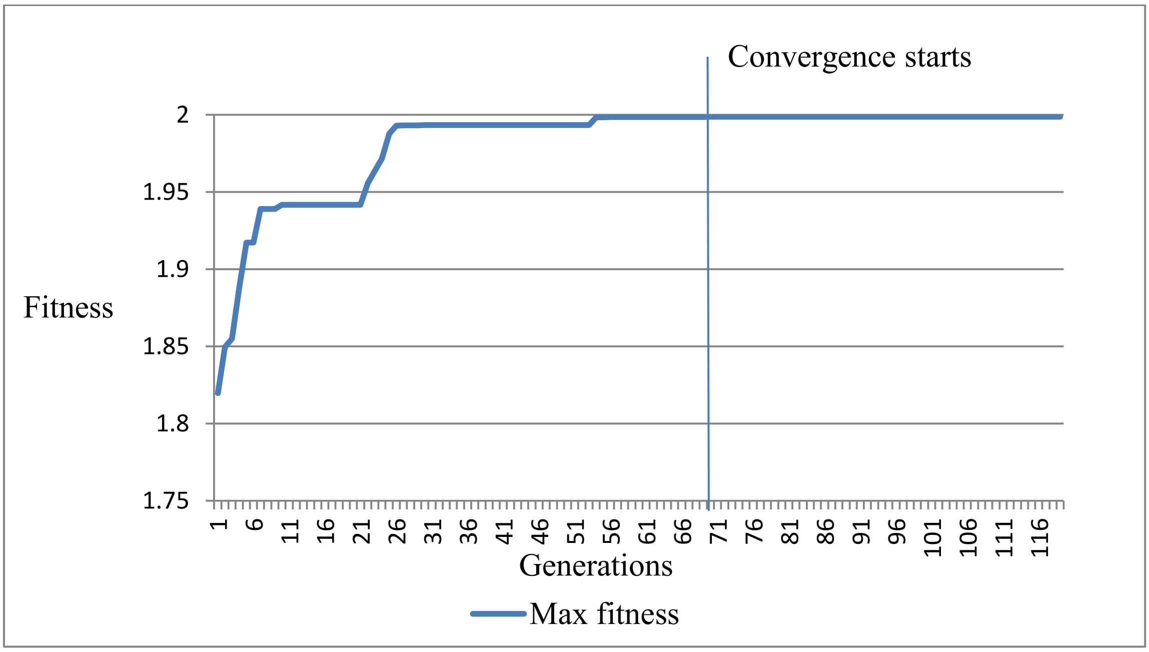


Figure 7-4 A typical plot for convergence of fitness for the training data of Leukemia using NSC$_M$-GA

As seen in Figure 7-4, convergence in this sample run occurred after 69 generations with the maximum fitness of 1.99. The optimal shrinkage threshold obtained for each of the 15 independent runs had the same maximum fitness of 1.99 which produced the same set of 9 genes for the Leukemia cancer dataset. As shown in Table 7-3, this set of 9 genes produced the average classification accuracy of 94.12% on the unseen test set. The classification accuracy is slightly different from those obtained using NSC-GA. Owing the fact that the optimal shrinkage threshold value obtained from NSC$_M$-GA is slightly different from that of NSC-GA. These two slightly different optimal thresholds still mapped to the same set of 9 genes but produced slightly different classification accuracy on the same unseen test dataset.

Table 7-3 Comparison of classification results between NSC-GA and NSC_M-GA for Leukemia data

NSC-GA (Euclidean dist.)		Proposed NSC_M-GA (Mahalanobis dist.)	
Number of genes	Unseen Test (%)	Number of genes	Unseen Test (%)
9	97.05	9	94.12

7.2.1.4. Lymphoma cancer data

As seen in Figure 7-5, convergence in this sample run occurred after 31 generations with the maximum fitness of 1.99. The optimal shrinkage threshold obtained for each of the 15 independent runs had the same maximum fitness of 1.99 which produced the same set of 3 genes for the Lymphoma cancer dataset. This set of 3 genes produced the average classification accuracy of 100% on the unseen test set. The 3 gene set with gene accession numbers, GENE3327X, GENE3329X and GENE3361X, selected by NSC_M-GA for Lymphoma dataset, is a subset of the following sets consisting of 7, 12, 128, 129 and 132 genes found using NSC-GA. For this dataset, the proposed approach selected a smaller set of genes resulting in a higher classification accuracy on the same unseen test dataset.

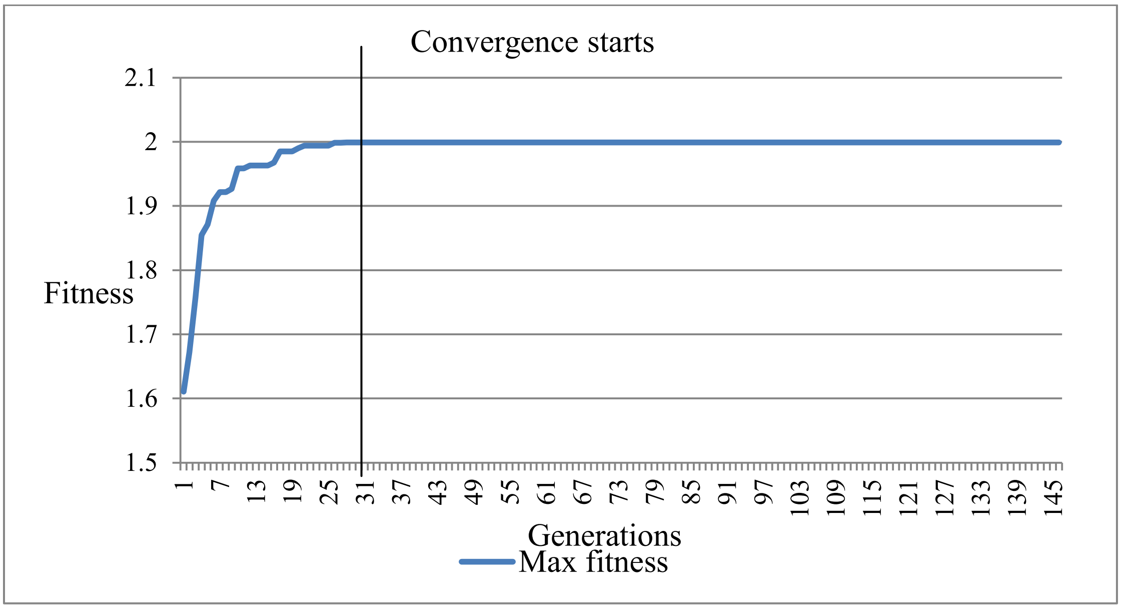


Figure 7-5 A typical plot for convergence of fitness for the training data of Lymphoma using NSC$_M$-GA

Table 7-4 Comparison of classification results between NSC-GA and NSC$_M$-GA for Lymphoma data

NSC-GA (Euclidean dist.)		Proposed NSC$_M$-GA (Mahalanobis dist.)	
Number of genes	Unseen Test (%)	Number of genes	Unseen Test (%)
7 12 128 129 132	95.45 95.45 100 100 100	3	100

7.2.1.5. Lung cancer data

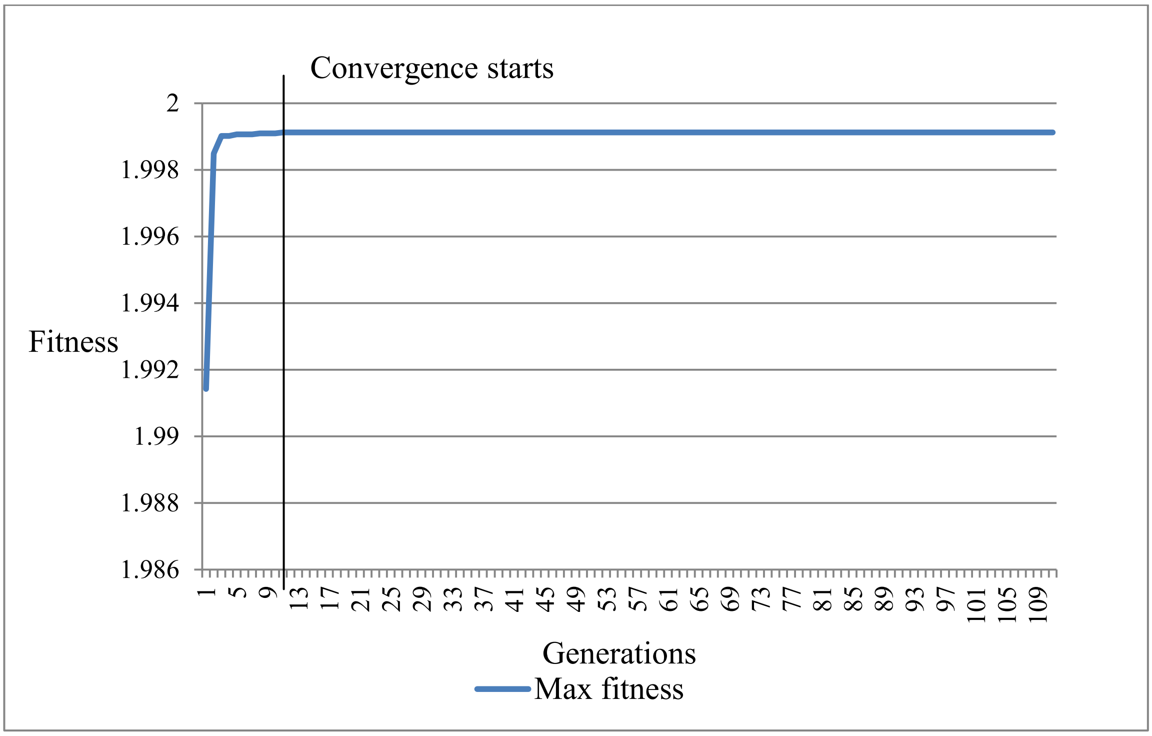


Figure 7-6 A typical plot for convergence of fitness for the training data of Lung using NSC$_M$-GA

As seen in Figure 7-6, convergence in this sample run occurred after 10 generations with the maximum fitness of 1.998. The optimal shrinkage threshold obtained for each of the 15 independent runs produced the set of 9 and 11 genes for the Lung cancer dataset. As shown in Table 7-5, the set of 9 and 11 genes resulted in the average classification accuracy of 98.88% on the unseen test set. These sets of 9 and 11 genes selected by NSC$_M$-GA are the same set of 9 and 11 genes found by using NSC-GA with the classification accuracy of 100%.

Table 7-5 Comparison of classification results between NSC-GA and NSC$_M$-GA for Lung data

NSC-GA (Euclidean dist.)		Proposed NSC$_M$-GA (Mahalanobis dist.)	
Number of genes	Unseen Test (%)	Number of genes	Unseen Test (%)
7, 8, 9, 10	100	9, 11	98.88

7.2.1.6. Ovarian cancer data

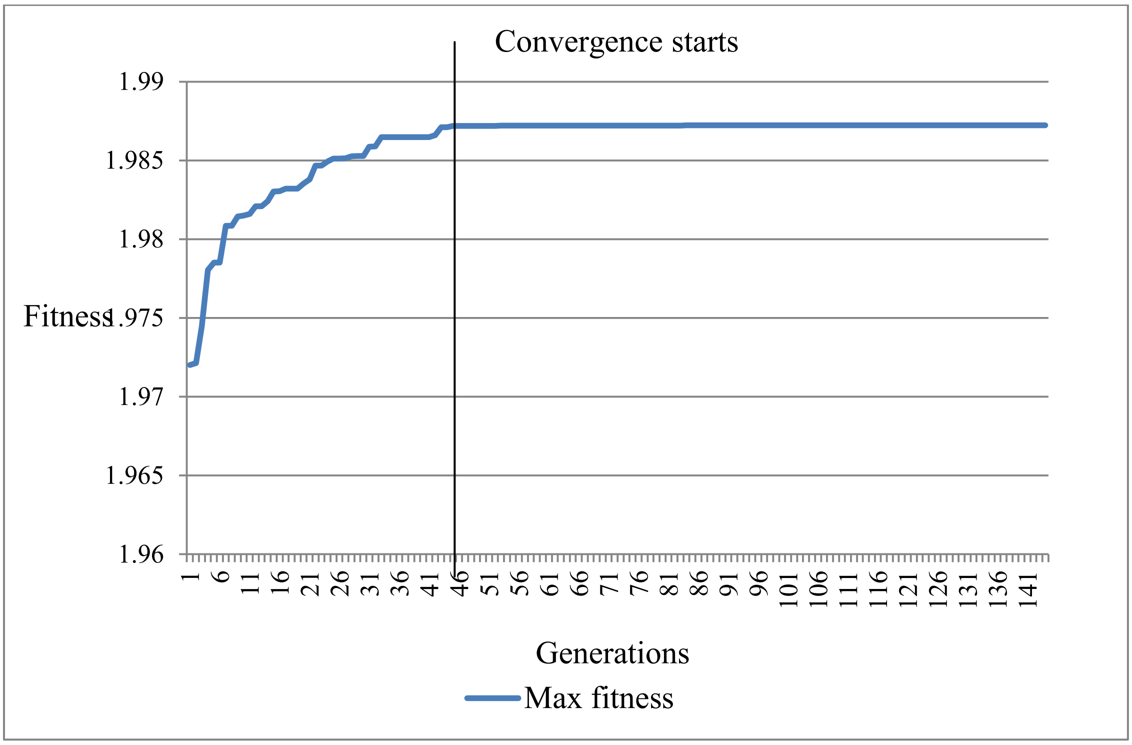


Figure 7-7 A typical plot for convergence of fitness for the training data of Ovarian using NSC$_M$-GA

As seen in Figure 7-7, convergence in this sample run occurred after 46 generations with the maximum fitness of 1.98. The optimal shrinkage threshold obtained for each of the 15 independent runs had the same maximum fitness of 1.98 which resulted in the set of 1 peptide for the Ovarian cancer dataset. As shown in Table 7-6, the set of 1 gene (MZ244.36855) gave the average classification accuracy of 96.06% on the unseen test set. This set of 1 peptide selected by NSC$_M$-GA is a subset of 7 peptides found using the NSC-GA approach but gave the same classification accuracy of 96.06%. It appears that a major contributing factor relates to 1 peptide and thus may warrant further investigations into the relevance of the remaining features. This sort of information for analysis in bioinformatics is important as reducing the number of features to a smaller promising set for further investigations would reduce costs associated with future experiments and analysis.

Table 7-6 Comparison of classification results between NSC-GA and NSC$_M$-GA for Ovarian data

NSC-GA (Euclidean dist.)		Proposed NSC$_M$-GA (Mahalanobis dist.)	
Number of genes	Unseen Test (%)	Number of genes	Unseen Test (%)
7	96.06	1	96.06

7.2.1.7. Prostate cancer data

As seen in Figure 7-8, convergence in this sample run occurred after 9 generations with the maximum fitness of 1.998. The optimal shrinkage threshold obtained for each of the 15 independent runs had the same maximum fitness of 1.998 which produced the sets of 17 genes for the Prostate cancer dataset. As shown in Table 7-7, this set of 17 genes resulted in the average classification accuracy of 100% on the unseen test set. The gene accession numbers of 17 genes selected by NSC$_M$-GA for Prostate cancer dataset are: 31444_s_at, 31527_at, 33614_at, 41468_at, 37639_at, 39756_g_at, 40435_at, 40436_g_at, 36587_at, 36666_at, 37720_at, 38406_f_at, 38429_at, 40282_s_at, 769_s_at, 556_s_at and 216_at. The gene accession number, 31444_s_at and 769_s_at are listed in the prognosis gene patent that indicates high risk for TTD (*time to death*) (Liu & Iba, 2002). In comparison, this set of 17 genes is a superset of the 6 genes obtained using NSC-GA which produced a resulting classification accuracy of 90.2% on the same unseen test dataset.

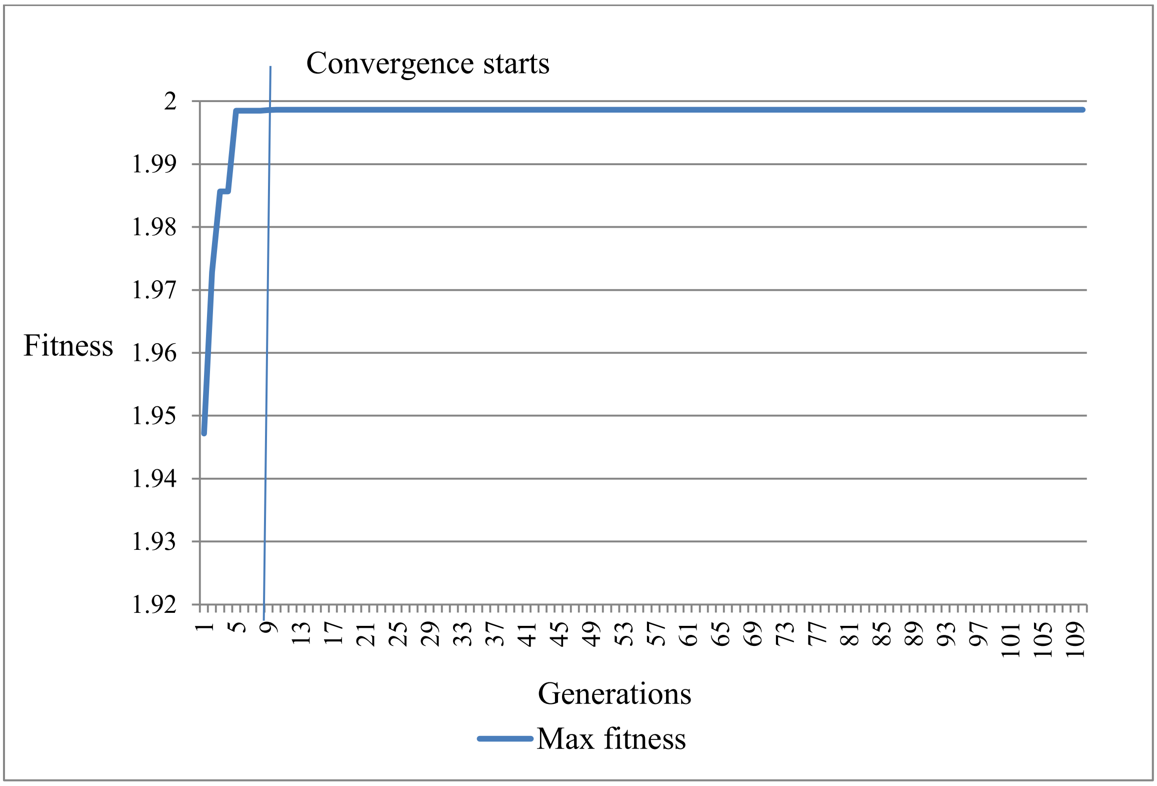


Figure 7-8 A typical plot for convergence of fitness for the training data of Prostate using NSC$_M$-GA

Table 7-7 Comparison of classification results between NSC-GA and NSC$_M$-GA for Prostate data

NSC-GA (Euclidean dist.)		Proposed NSC$_M$-GA (Mahalanobis dist.)	
Number of genes	Unseen Test (%)	Number of genes	Unseen Test (%)
6	90.2	17	100

7.2.2. NSC$_P$-GA approach

Similar to the experiments for NSC$_M$-GA approach, the same parameter settings, experimental conditions and evaluation strategy were also used here for evaluating NSC$_P$-GA. The results from these experiments are reported as follows.

7.2.2.1. Ray *et al.* AD data

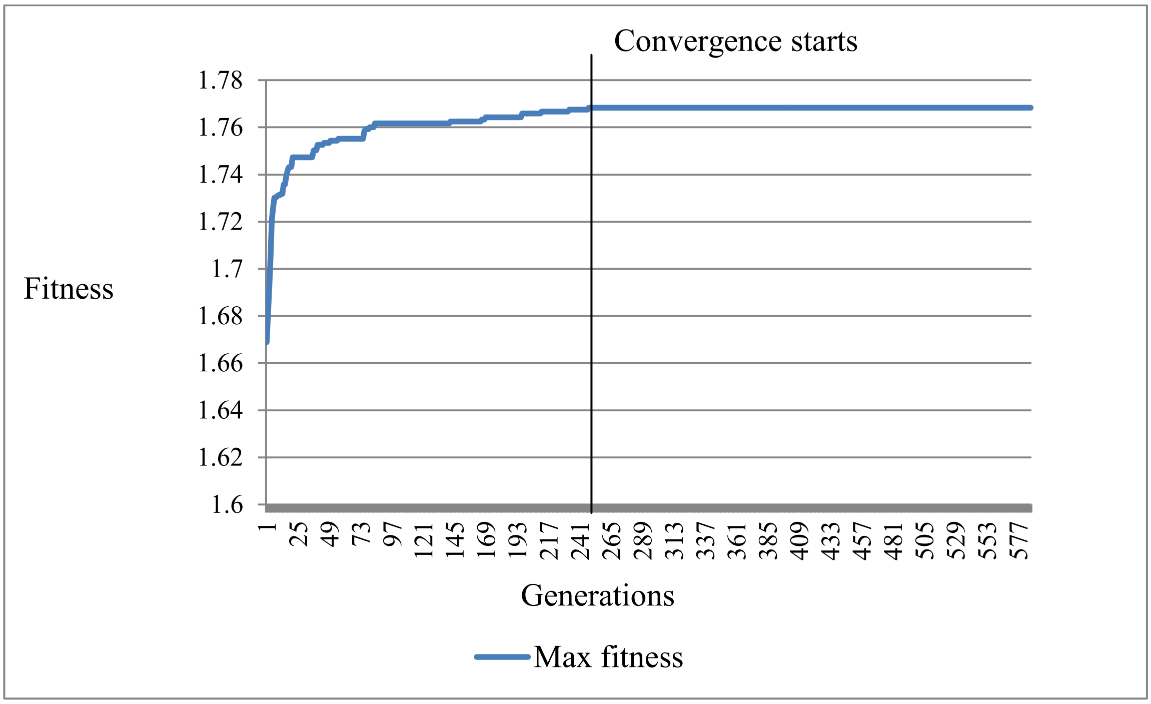


Figure 7-9 A typical plot for convergence of fitness for the training data of AD using NSC$_P$-GA

As seen in Figure 7-9, convergence in this sample run occurred after 251 generations with the maximum fitness of 1.768 in this sample run. The optimal shrinkage threshold obtained for each of the 15 independent runs had the same maximum fitness of 1.768 which produced the same set of 9 proteins for the AD dataset. As shown in Table 7-8, the set of 9 proteins resulted in the average classification accuracy of 92.39% on unseen test dataset, an improvement over the results from the set of 11 proteins obtained via NSC-GA. The selected 9 proteins, PDGF-BB_1, RANTES_1, IL-1a_1, TNF-a_1, EGF_1, M-CSF_1, ICAM-1_1, IL-3-1 and GCSF_1, are a subset of the sets of 11 and 18 proteins obtained using NSC-GA and NSC$_M$-GA, respectively, as well as being a subset of the 18 proteins in Ray *et al.* (2007).

Table 7-8 Comparison of classification results between NSC-GA and NSC$_P$-GA for AD data

NSC-GA (Euclidean dist.)		Proposed NSC$_P$-GA (Pearson dist.)	
Number of genes	Unseen Test (%)	Number of genes	Unseen Test (%)
11	89.49	9	92.39

7.2.2.2. Alon *et al.* Colon cancer data

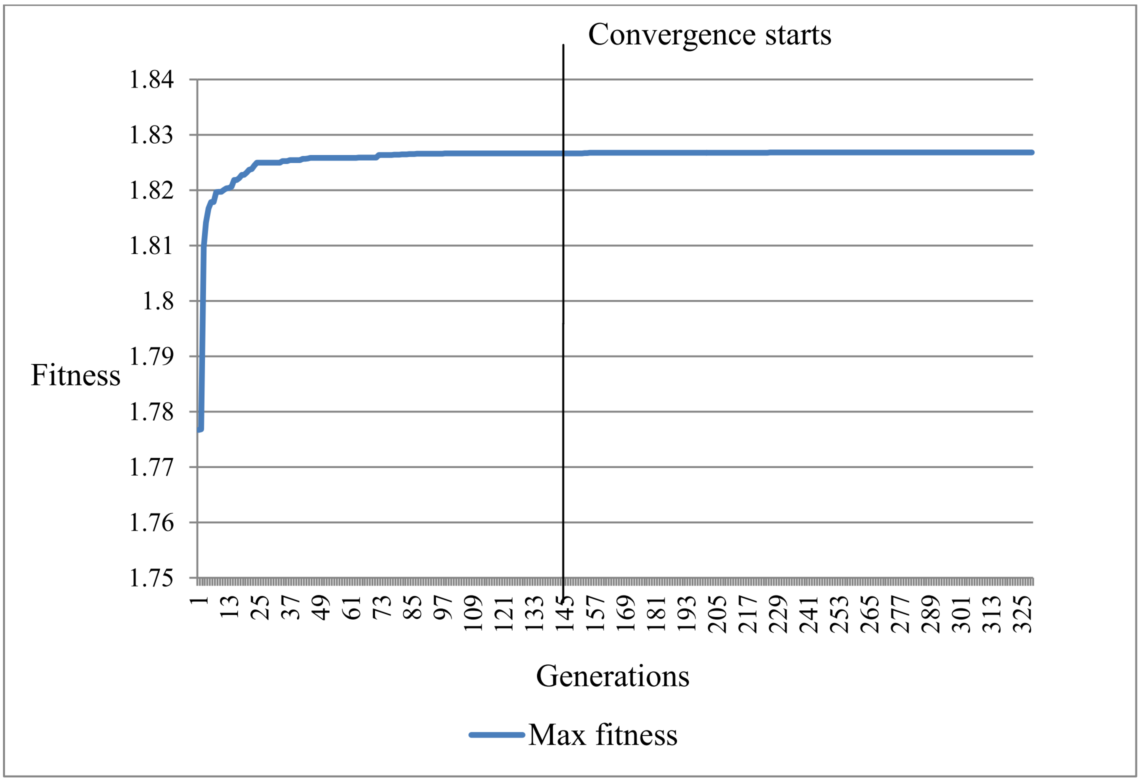


Figure 7-10 A typical plot for convergence of fitness for the training data of Colon using NSC$_P$-GA

As seen in Figure 7-10, convergence in this sample run occurred after 144 generations with the maximum fitness of 1.826. Seven runs produced a maximum fitness of 1.826 which gave the same set of 42 genes and other 8 runs produced a maximum fitness of 1.823 which mapped to the same set of 6 genes which is a subset of the 42 gene set. As shown in Table 7-9, these sets of 42 and 6 genes, each resulted in an average classification accuracy of 100% on the unseen test dataset. The set of six genes selected by the proposed approach is the same set of 6 genes found by the NSC-GA approach and is also a subset of the sets of 28 genes (from NSC-GA) and 42 genes.

Table 7-9 Comparison of classification results between NSC-GA and NSC$_{P}$-GA for Colon data

NSC-GA (Euclidean dist.)		Proposed NSC$_{P}$-GA (Pearson dist.)	
Number of genes	Unseen Test (%)	Number of genes	Unseen Test (%)
28 6	100 93.75	42 6	100

7.2.2.3. Leukemia cancer data

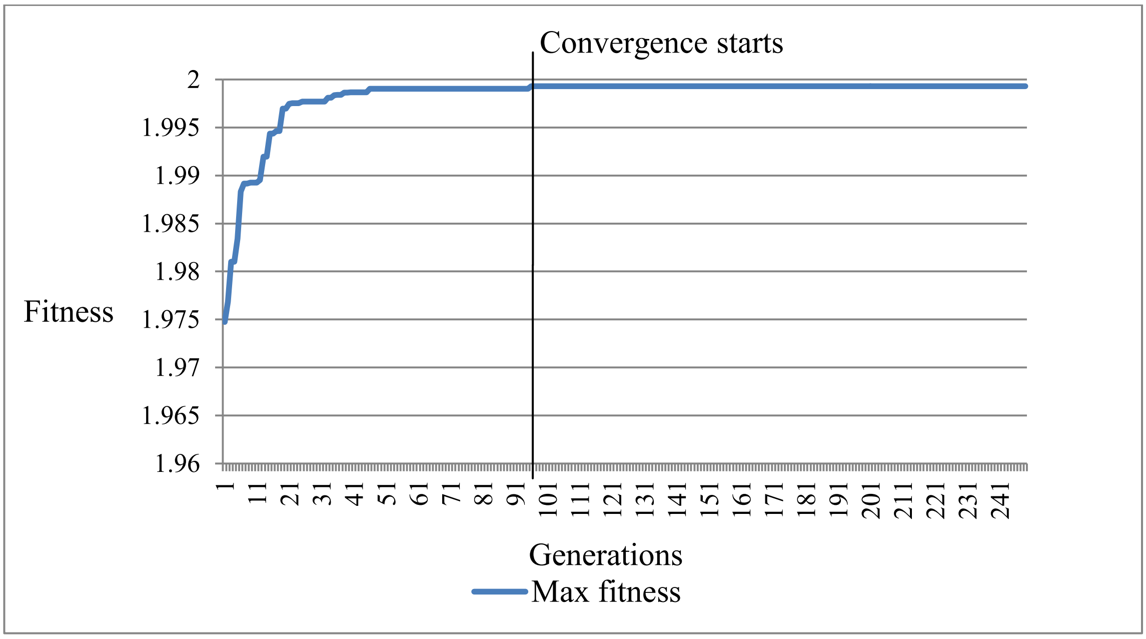


Figure 7-11 A typical convergence of fitness plot for the training data of Leukemia data using NSC$_{P}$-GA

As seen in Figure 7-11, convergence in this sample run occurred after 91 generations with the maximum fitness of 1.99. Three different optimal shrinkage thresholds were obtained from the 15 independent runs, each are associated with 7 sets of 4 genes, 5 sets of 5 genes and 3 sets of 24 genes respectively. As shown in Table 7-10, these sets of 4, 5 and 24 genes, each resulted in an average classification accuracy of 100% on unseen test data. The sets of 4 and 5 genes, each is a subset of the 9 gene set produced from using NSC-GA.

Table 7-10 Comparison of classification results between NSC-GA and NSC$_P$-GA for Leukemia data

NSC-GA (Euclidean dist.)		Proposed NSC$_P$-GA (Pearson dist.)	
Number of genes	Unseen Test (%)	Number of genes	Unseen Test (%)
9	97.05	4 5 24	100

7.2.2.4. Lymphoma cancer data

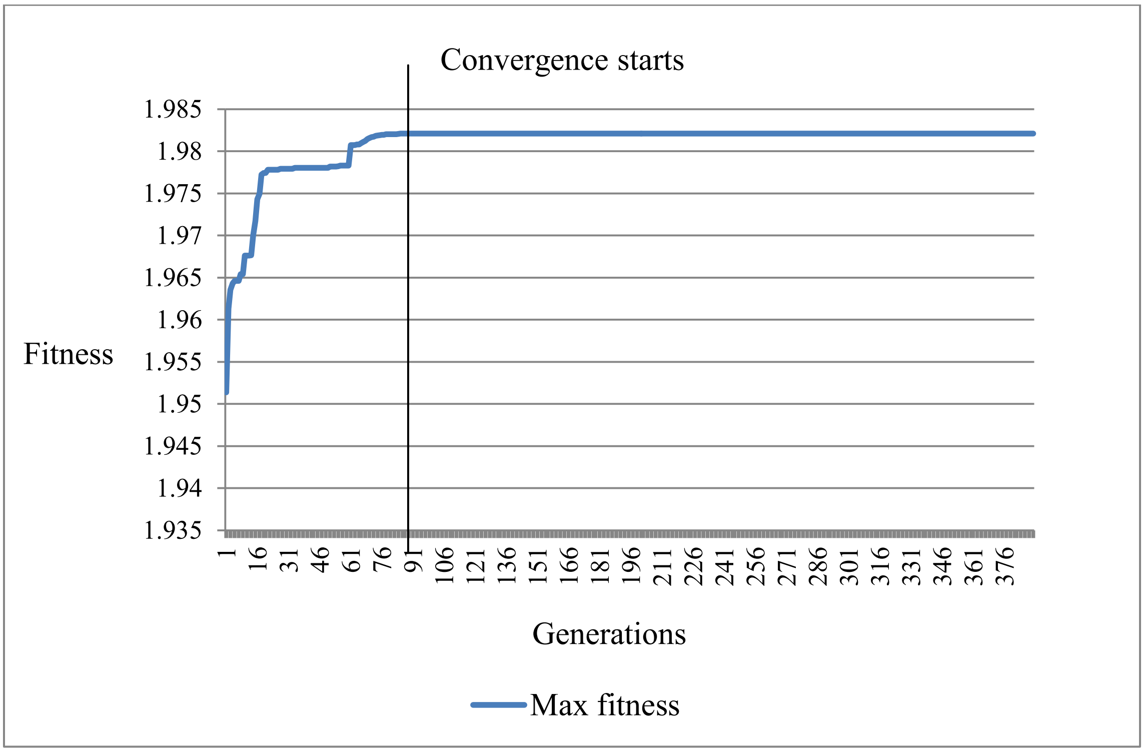


Figure 7-12 A typical plot for convergence of fitness for the training data of Lymphoma using NSC$_P$-GA

As seen in Figure 7-12, convergence in this sample run occurred after 91 generations with the maximum fitness of 1.983. Five different optimal shrinkage thresholds obtained for the 15 independent runs, each are associated with 5 sets of 72 genes, 3 sets of 73 genes, 5 sets of 75 genes, 1 set of 77 and 80 genes respectively. As shown in Table 7-11, these sets of 72, 73, 75, 77 and 80 genes, each produced an average classification accuracy of 100% on unseen test data. The sets of 72, 73, 75, 77 and 80 genes are subset of the 128 gene set produced using NSC-GA.

Table 7-11 Comparison of classification results between NSC-GA and NSC$_P$-GA for Lymphoma data

NSC-GA (Euclidean dist.)		Proposed NSC$_P$-GA (Pearson dist.)	
Number of genes	Unseen Test (%)	Number of genes	Unseen Test (%)
7 12 128 129 132	95.45 95.45 100 100 100	72 73 75 77 80	100

7.2.2.5. Lung cancer data

As seen in Figure 7-13, convergence in the sample run occurred after 36 generations with the maximum fitness of 1.9996. Three different optimal shrinkage thresholds were obtained from the 15 independent runs. Each is associated with 8 sets of 4 genes, 5 sets of 5 genes and 2 sets of 7 genes, respectively. The smaller set is a subset of the larger set. As shown in Table 7-12, the set of 4, 5 and 7 genes resulted in an average classification accuracy of 100% on unseen test data. The set of 7 genes, with gene accession numbers 32551_at, 33328_at, 34320_at, 36533_at, 37157_at, 37716_at and 37954_at. Sets of 4, 5 and 7 genes are each a subset of the sets consisting of 8, 9, 10 and 11 genes produced using NSC-GA.

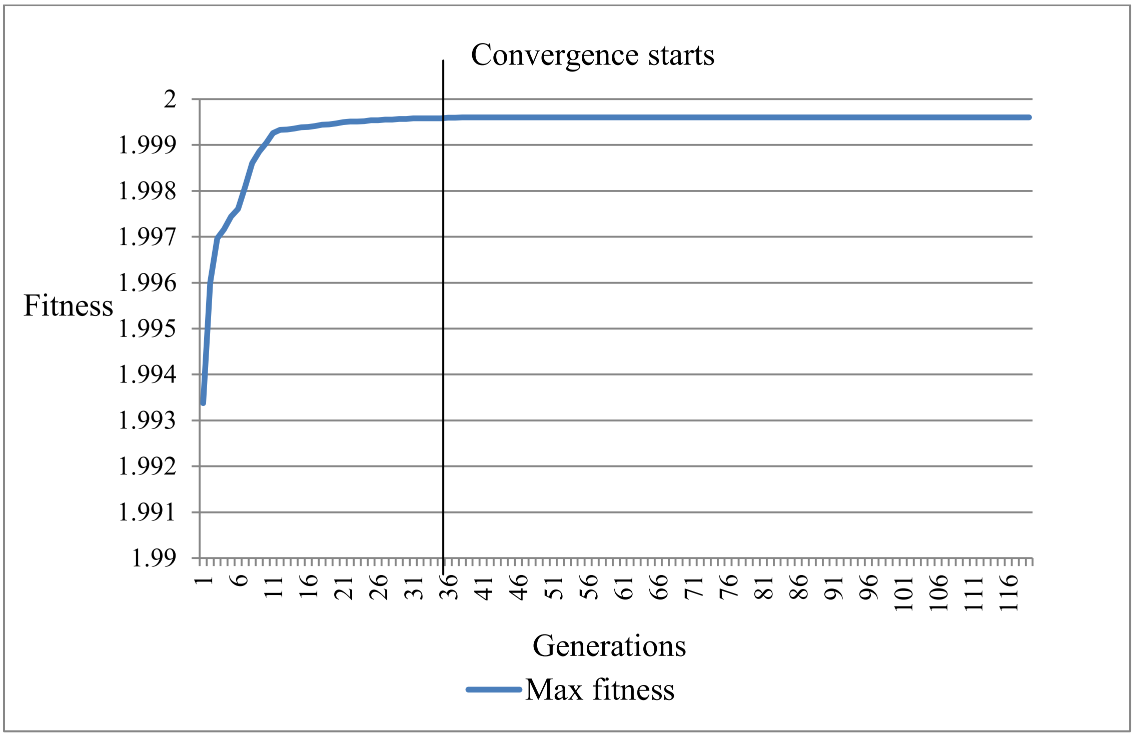


Figure 7-13 A typical plot for convergence of fitness for the training data of Lung using NSC$_P$-GA

Table 7-12 Comparison of classification results between NSC-GA and NSC$_P$-GA for Lung data

NSC-GA (Euclidean dist.)		Proposed NSC$_P$-GA (Pearson dist.)	
Number of genes	Unseen Test (%)	Number of genes	Unseen Test (%)
8 9 10 11	100	4 5 7	100

7.2.2.6. Ovarian cancer data

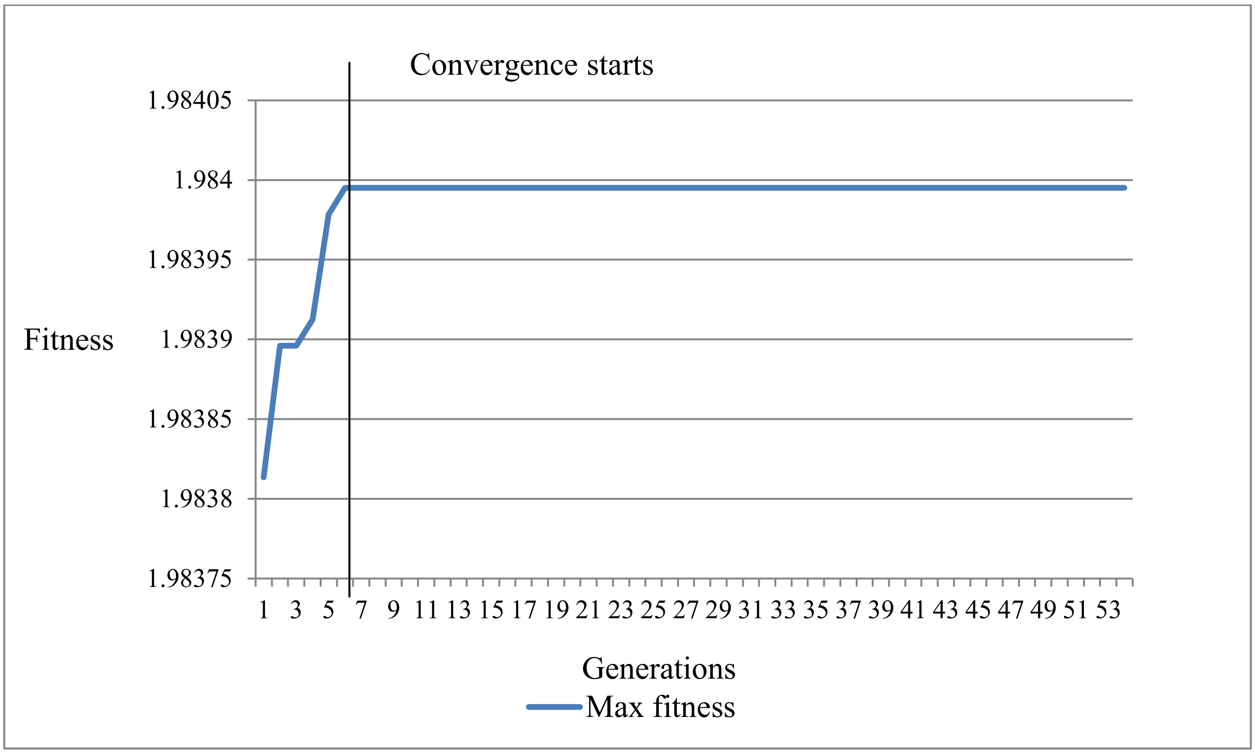


Figure 7-14 A typical plot for convergence of fitness for the training data of Ovarian using NSC$_P$-GA

As seen in Figure 7-14, convergence in this sample run occurred after 7 generations with the maximum fitness of 1.984. Four different optimal shrinkage thresholds were obtained from the 15 independent runs, each are associated with 2 sets of 2 peptides, 8 sets of 8 peptides, 3 sets of 9 peptides and 2 set of 10 peptides respectively. As shown in Table 7-13, the set of 2 peptides produced an average classification accuracy of 96.85% and the set of 8, 9 and 10 genes produced an average classification accuracy of 96.06% on unseen test data. The set of the 2 peptides, MZ244.95245 and MZ245.24466, is a subset of 7 peptides found by the approach NSC-GA.

Table 7-13 Comparison of classification results between NSC-GA and NSC$_P$-GA for Ovarian data

NSC-GA (Euclidean dist.)		Proposed NSC$_P$-GA (Pearson dist.)	
Number of peptides	Unseen Test (%)	Number of peptides	Unseen Test (%)
7	96.06	2 8 9 10	96.85 96.06 96.06 96.06

7.2.2.7. Prostate cancer data

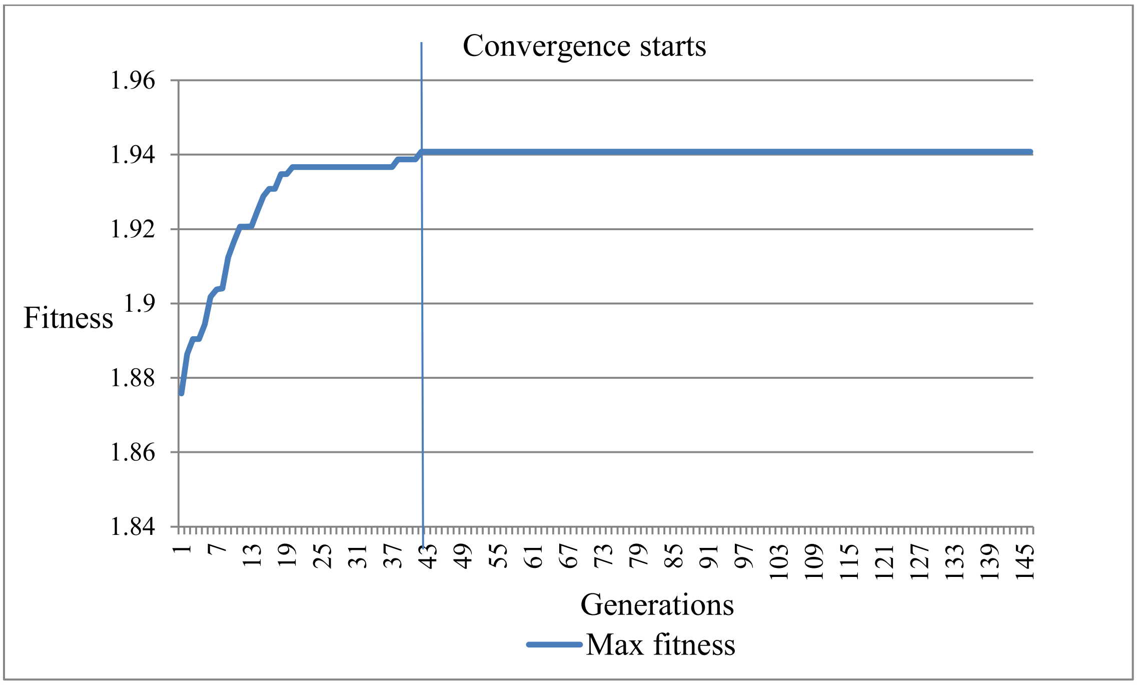


Figure 7-15 A typical plot for convergence of fitness for the training data of Prostate using NSC$_P$-GA

As seen in Figure 7-15, convergence in this sample run occurred after 43 generations with the maximum fitness of 1.942. The optimal shrinkage threshold obtained for each of the 15 independent runs had the same maximum fitness of 1.942 which mapped to the same set of five genes. As shown in Table 7-14, this set of five genes produced an average classification accuracy of 90.2% on unseen test data. This five gene set, with gene accession numbers 41468_at, 37639_at, 38406_f_at, 769_s_at and 556_s_at , is a subset of six genes found using NSC-GA.

Table 7-14 Comparison of classification results between NSC-GA and NSC$_P$-GA for Prostate data

NSC-GA (Euclidean dist.)		Proposed NSC$_P$-GA (Pearson dist.)	
Number of genes	Unseen Test (%)	Number of genes	Unseen Test (%)
6	90.2	5	90.2

7.2.3. NSC$_{MD}$-GA approach

Similar to the experiments for NSC$_M$-GA and NSC$_P$-GA, the same parameter settings, experimental conditions and evaluation strategy were also used here for evaluating NSC$_{MD}$-GA. The results from these experiments are reported as follows.

7.2.3.1. Ray *et al.* AD data

As seen in Figure 7-16, convergence in this sample run occurred after 51 generations with the maximum fitness of 1.84. The optimal shrinkage threshold obtained for each of the 15 independent runs had the same maximum fitness of 1.84 which mapped to the same set of 4 proteins. As shown in Table 7-15, this set of 4 proteins resulted in an average classification accuracy of 91.3% on unseen test data. This 4 protein set, PDGF-BB_1, RANTES_1, TNF-a_1 and IL-1a_1, is a subset of the following sets with each comprising of 11, 18 and 9 features selected using NSC-GA, NSC$_M$-GA and NSC$_P$-GA, respectively, and is also a subset of 18 features in Ray *et al*. (2007).

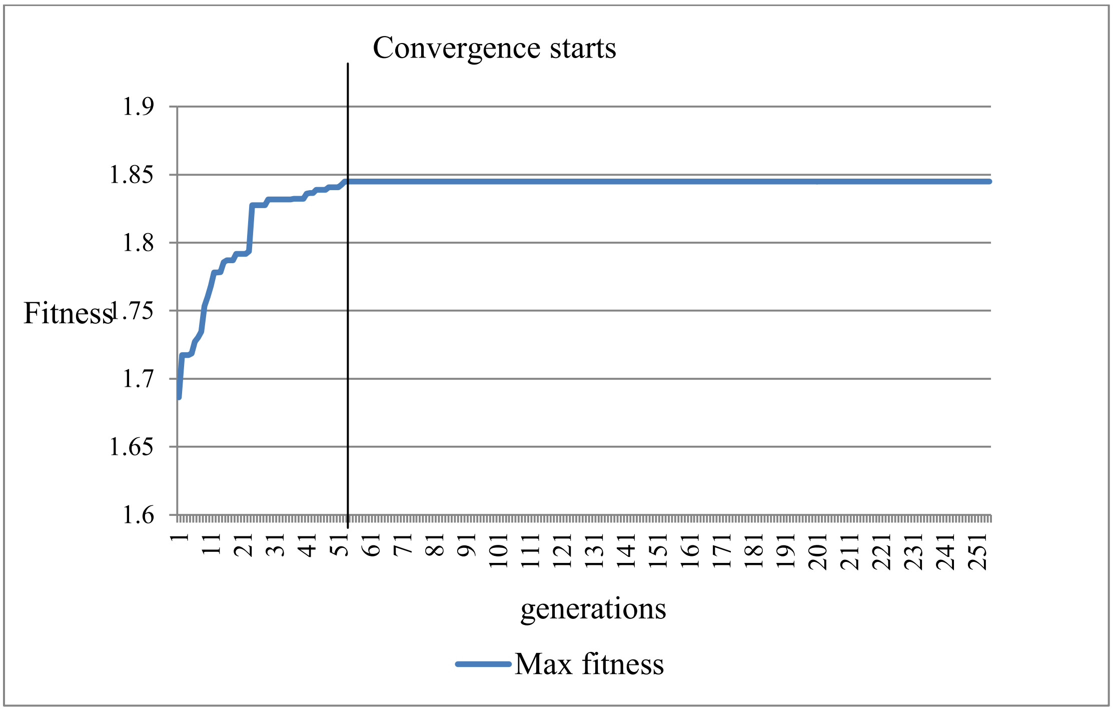


Figure 7-16 A typical plot for convergence of fitness for the training data of AD using $\mathrm{NSC}_{\mathrm{MD}}$-GA

Table 7-15 Comparison of classification results between NSC-GA and $\mathrm{NSC}_{\mathrm{MD}}$-GA for AD data

NSC-GA (Euclidean dist.)		Proposed $\mathrm{NSC}_{\mathrm{MD}}$-GA (Mass dist.)	
Number of proteins	Unseen Test (%)	Number of proteins	Unseen Test (%)
11	90.21	4	91.3

7.2.3.2. Alon *et al.* Colon cancer data

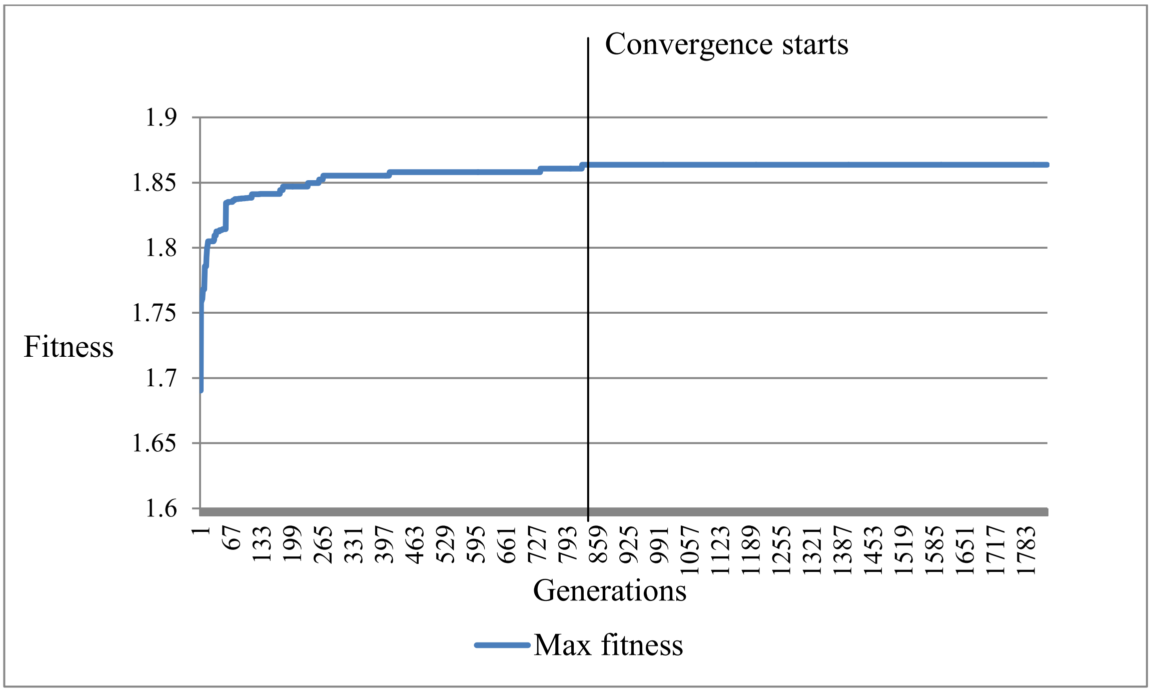


Figure 7-17 A typical plot for convergence of fitness for the training data of Colon using NSC_{MD}-GA

As seen in Figure 7-17, convergence in this sample run occurred after 853 generations with the maximum fitness of 1.86. The optimal shrinkage threshold obtained from each of the 15 independent runs had the same maximum fitness of 1.86 which mapped to the same set of 12 genes. As shown in Table 7-16, this set of 12 genes resulted in an average classification accuracy of 100% on the unseen test data. The set of 12 genes has gene accession numbers: T71025, Z24727, M76378, M63391, M76378, R87126, X12671, M76378, T92451, H43887, T47377 and J02854. This set is also a subset of the set of 28 genes found using NSC-GA. The interesting point from the perspective of early diagnostic test developments is a small set with high discriminatory potentials and here both sets (set of 12 genes and the set of 28 genes) produced the same classification accuracy on the unseen test dataset.

Table 7-16 Comparison of classification results between NSC-GA and NSC$_{MD}$-GA for Colon data

NSC-GA (Euclidean dist.)		Proposed NSC$_{MD}$-GA (Mass dist.)	
Number of genes	Unseen Test (%)	Number of genes	Unseen Test (%)
28 6	100 93.75	12	100

7.2.3.3. Leukemia cancer data

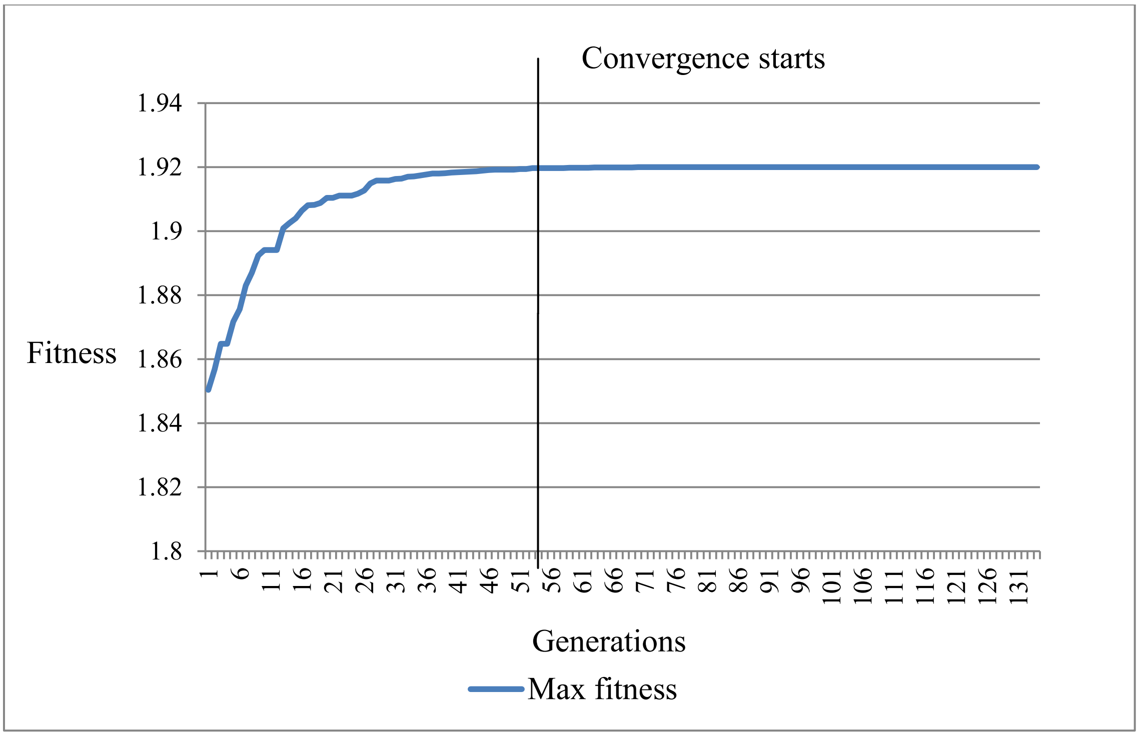


Figure 7-18 A typical plot for convergence of fitness for the training data of Leukemia using NSC$_{MD}$-GA

As seen in Figure 7-18, convergence in this sample run occurred after 56 generations with the maximum fitness of 1.92. The optimal shrinkage threshold obtained for each of the 15 independent runs had the same maximum fitness of 1.92 which mapped to the same set of 3 genes. As shown in Table 7-17, this set of 3 genes gave the average classification accuracy of 94.12% on unseen test data. The set of 3 genes selected with gene accession numbers, M27891, M84526, and X17042, is a subset of 5 features found in NSC$_P$GA, and also a subset of 9 genes found in NSC-GA.

Table 7-17 Comparison of classification results between NSC-GA and NSC$_{MD}$-GA for Leukemia data

NSC-GA (Euclidean dist.)		Proposed NSC$_{MD}$-GA (Mass dist.)	
Number of genes	Unseen Test (%)	Number of genes	Unseen Test (%)
9	97.05	3	94.12

7.2.3.4. Lymphoma cancer data

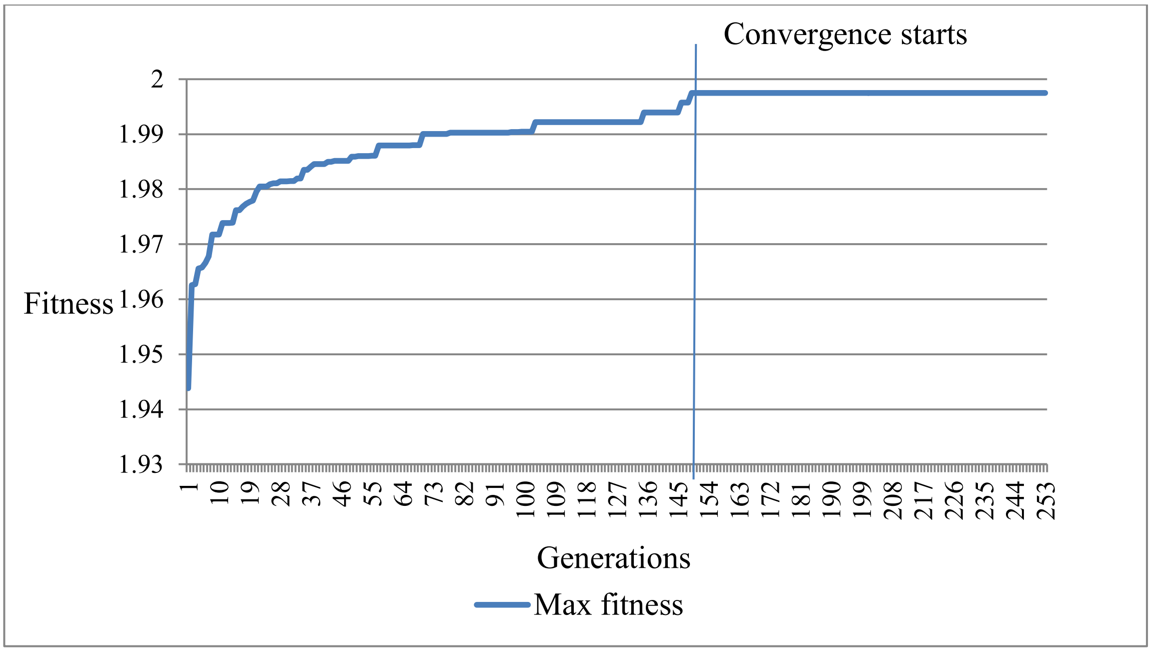


Figure 7-19 A typical plot for convergence of fitness for the training data of Lymphoma using NSC$_{MD}$-GA

As seen in Figure 7-19, the convergence in this sample run occurred after 151 generations with the maximum fitness of 1.997. The optimal shrinkage threshold obtained for each of the 15 independent runs had the same maximum fitness of 1.997 which produced the same set of 3 genes for the Lymphoma cancer dataset. As shown in Table 7-18, the set of 3 genes gave the average classification accuracy of 100% on unseen test data. The set of 3 genes selected with gene accession numbers, GENE3327X, GENE3329X and GENE3361X, is a subset of 7 features found in NSC-GA.

Table 7-18 Comparison of classification results between NSC-GA and NSC$_{MD}$-GA for Lymphoma data

NSC-GA (Euclidean dist.)		Proposed NSC$_{MD}$-GA (Mass dist.)	
Number of genes	Unseen Test (%)	Number of genes	Unseen Test (%)
7 12 128 129 132	95.45 95.45 100 100 100	3	100

7.2.3.5. Lung cancer data

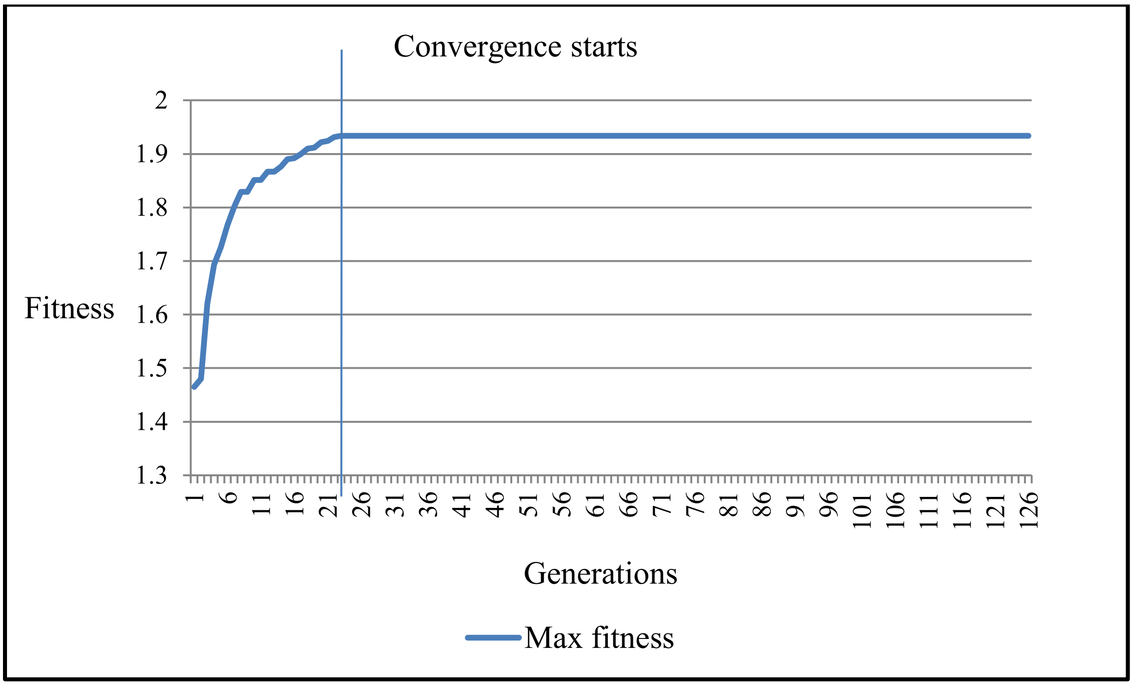


Figure 7-20 A typical plot for convergence of fitness for the training data of Lung using NSC$_{MD}$-GA

As seen in Figure 7-20, the convergence in this sample run occurred after 21 generations with the maximum fitness of 1.997. The optimal shrinkage threshold obtained for each of the 15 independent runs had the same maximum fitness of 1.997 which produced the same set of 2 genes for the Lung cancer dataset. As shown in Table 7-19, the set of 2 genes gave the average classification accuracy of 63.33% on unseen test data. The set of 2 genes selected with accession numbers, 33328_at and 40936_at, is a subset of 8 genes found in NSC-GA.

Table 7-19 Comparison of classification results between NSC-GA and NSC$_{MD}$-GA for Lung data

NSC-GA (Euclidean dist.)		Proposed NSC$_M$-GA (Mass dist.)	
Number of genes	Unseen Test (%)	Number of genes	Unseen Test (%)
8 9 10 11	100	2	63.33

7.2.3.6. Ovarian cancer data

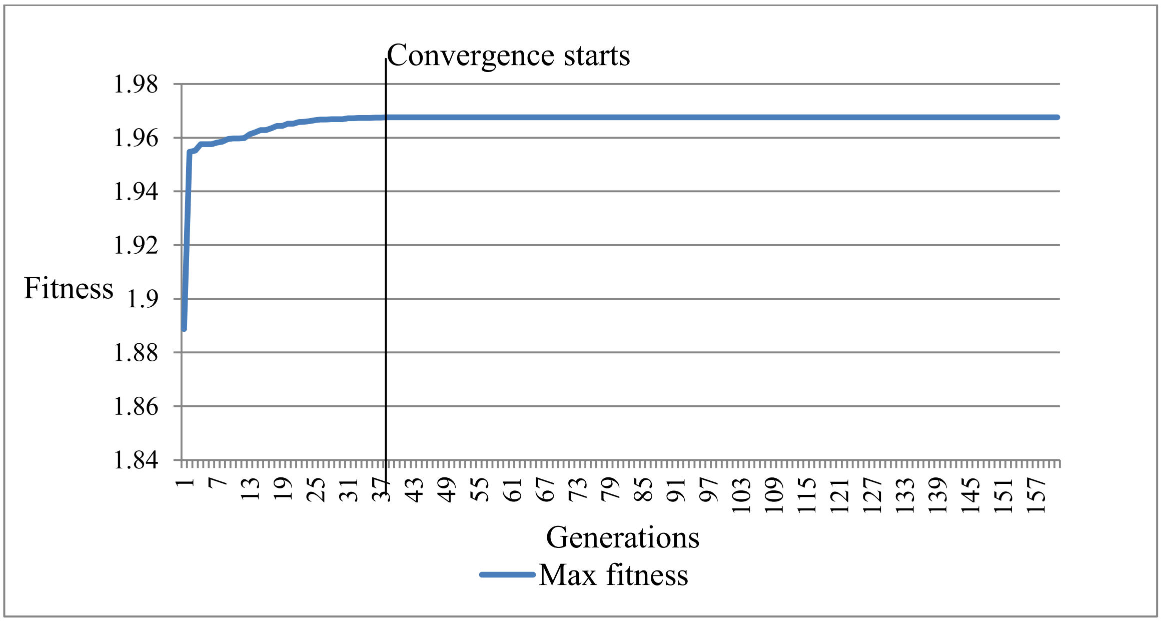


Figure 7-21 A typical plot for convergence of fitness for the training data of Ovarian using NSC$_{MD}$-GA

As seen in Figure 7-21, the convergence in this sample run occurred after 37 generations with the maximum fitness of 1.964. The optimal shrinkage thresholds obtained for the 15 independent runs produced the 12 sets of 10 peptides, 2 sets of 11 peptides and 1 set of 20 peptides for the Ovarian cancer dataset. As shown in Table

7-20, the set of 10, 11 and 20 peptides gave the average classification accuracy of 63.33% on unseen test data. The smaller set is a subset of the larger sets.

Table 7-20 Comparison of classification results between NSC-GA and NSC$_{MD}$-GA for Ovarian data

NSC-GA (Euclidean dist.)		Proposed NSC$_{MD}$-GA (Mass dist.)	
Number of peptides	Unseen Test (%)	Number of peptides	Unseen Test (%)
7	96.06	10 11 20	92.12 92.12 88.97

7.2.3.7. Prostate cancer data

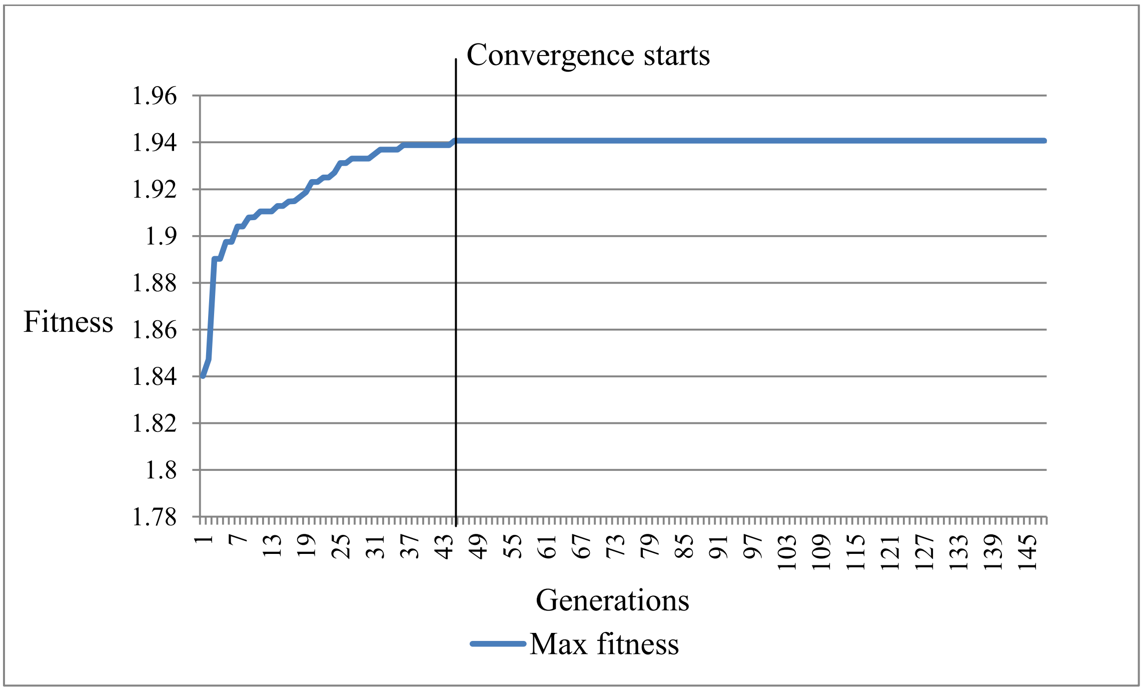


Figure 7-22 A typical plot for convergence of fitness for the training data of Prostate using NSC$_{MD}$-GA

As seen in Figure 7-22, the convergence occurred after 43 generations with the maximum fitness of 1.94. The optimal shrinkage threshold obtained for each of the 15 independent runs had the same maximum fitness of 1.94 which produced the same set of 5 genes for the Prostate cancer dataset. As shown in Table 7-21, the set of 5 genes gave the average classification accuracy of 94.12% on unseen test data. The set of 5 genes selected with gene accession numbers are 41468_at, 37639_at, 38406_f_at, 769_s_at and 556_s_at, which is a subset of 6 genes found in NSC-GA.

Table 7-21 Comparison of classification results between NSC-GA and NSC$_{MD}$-GA for Prostate data

NSC-GA (Euclidean dist.)		Proposed NSC$_{MD}$-GA (Mass dist.)	
Number of genes	Unseen Test (%)	Number of genes	Unseen Test (%)
6	90.2	5	94.12

7.2.4. Summary: selected feature subsets and corresponding classification results

Table 7-22 to Table 7-30 summarised all the features sets that were obtained using NSC$_M$-GA, NSC$_P$-GA, NSC$_{MD}$-GA and NSC-GA on the seven datasets.

Table 7-22 Summary of the sets of selected features obtained from NSC$_M$-GA, NSC$_P$-GA, NSC$_{MD}$-GA and NSC-GA for AD data

Proteins	NSC-GA (Euclidean dist.)	Proposed approaches		
		NSC$_M$-GA	NSC$_P$-GA	NSC$_{MD}$-GA
	Number of proteins	Number of proteins		
	11	18	9	4
PDGF-BB_1	√	√	√	√
RANTES_1	√	√	√	√
IL-1a_1	√	√	√	√
TNF-a_1	√	√	√	√
EGF_1	√	√	√	
M-CSF_1	√	√	√	
ICAM-1_1	√	√	√	
IL-3_1	√	√	√	
IL-11_1	√	√	√	
GCSF_1	√	√		
ANG-2_1	√	√		
PARC_1		√		
GDNF_1		√		
TRAIL R4_1		√		
IL-8_1		√		
MIP-1d_1		√		
IGFBP-6_1		√		
MCP-3_1		√		

Table 7-23 Summary of the sets of selected features obtained from NSC$_{M}$-GA, NSC$_{P}$-GA, NSC$_{MD}$-GA and NSC-GA for Colon data

Gene Accession number	NSC-GA (Euclidean dist.)	Proposed approaches		
		NSC$_{M}$-GA	NSC$_{P}$-GA	NSC$_{MD}$-GA
	Number of genes	Number of genes		
	28	7	6	12
T71025	√	√	√	√
M63391	√	√	√	√
R87126	√	√	√	√
M76378	√	√	√	√
T92451	√	√	√	√
J02854	√	√	√	√
M76378	√	√		√
R78934	√			
M26697	√			
Z24727	√			√
X55715	√			
T60778	√			
T57619	√			
M76378	√			√
H64489	√			
Z50753	√			
T60155	√			
M64110	√			
H40560	√			
T58861	√			
M22382	√			
X12671	√			√
T95018	√			
X86693	√			
H43887	√			√

T47377	√			√
L05144	√			
H55758	√			

Table 7-24 Summary of the sets of selected features obtained from NSC$_M$-GA, NSC$_P$-GA, NSC$_{MD}$-GA and NSC-GA for Leukemia data

Gene Accession number	NSC-GA (Euclidean dist.)	Proposed approaches				
		NSC$_M$-GA	NSC$_P$-GA			NSC$_{MD}$-GA
	Number of genes	Number of genes				
	9	9	4	5	24	3
M27891	√	√	√	√	√	√
M84526	√	√	√	√	√	√
M96326	√	√	√	√	√	√
X17042	√	√	√	√	√	
U50136	√	√		√	√	
U46751	√	√			√	
X95735	√	√			√	
M28130	√	√			√	
Y00787	√	√			√	
L08246					√	
L16896					√	
M11147					√	
M16038					√	
M19507					√	
M55150					√	
M57710					√	
M62762					√	
M63138					√	
M69043					√	
Y12670					√	
X85116					√	

J03801					√	
M19045					√	
X14008					√	

Table 7-25 Summary of the sets of selected features obtained from NSC$_M$-GA, NSC$_P$-GA, NSC$_{MD}$-GA and NSC-GA for Lymphoma data

Gene accession number	NSC-GA (Euclidean dist.)		Proposed approaches		
			NSC$_M$-GA	NSC$_P$-GA	NSC$_{MD}$-GA
	Number of genes		Number of genes		
	12	7	3	The list of 80 genes is shown in Table 7-26	3
GENE3327X	√	√	√		√
GENE3329X	√	√	√		√
GENE3361X	√	√	√		√
GENE3332X	√	√			
GENE3330X	√	√			
GENE3258X	√	√			
GENE3256X	√	√			
GENE3328X	√				
GENE3314X	√				
GENE3260X	√				
GENE1252X	√				
GENE3967X	√				

Table 7-26 List of 80 genes in the selected set obtained from NSC$_P$-GA for Lymphoma data

Gene accession numbers							
GENE3940X	GENE3554X	GENE3325X	GENE3338X	GENE3259X	GENE1212X	GENE3966X	GENE1693X
GENE3941X	GENE2496X	GENE3326X	GENE3341X	GENE3256X	GENE1213X	GENE3967X	GENE1694X
GENE3939X	GENE2326X	GENE3327X	GENE3314X	GENE3261X	GENE1251X	GENE3968X	GENE1697X
GENE3946X	GENE2106X	GENE3328X	GENE3312X	GENE3263X	GENE1252X	GENE947X	GENE1719X
GENE3945X	GENE2066X	GENE3329X	GENE3311X	GENE3264X	GENE1174X	GENE3932X	GENE1720X
GENE3947X	GENE2065X	GENE3330X	GENE3309X	GENE3265X	GENE1159X	GENE3617X	GENE3839X
GENE3699X	GENE3290X	GENE3331X	GENE3361X	GENE3246X	GENE3988X	GENE3815X	GENE1349X
GENE3755X	GENE3347X	GENE3332X	GENE3258X	GENE2760X	GENE3987X	GENE384X	GENE1171X
GENE3556X	GENE3346X	GENE3334X	GENE3257X	GENE3025X	GENE3986X	GENE1609X	GENE1080X
GENE3555X	GENE3315X	GENE3335X	GENE3260X	GENE1211X	GENE3965X	GENE1616X	GENE1556X

Table 7-27 Summary of the sets of selected features obtained from NSC$_M$-GA, NSC$_P$-GA, NSC$_{MD}$-GA and NSC-GA for Lung data

Gene accession number	NSC-GA (Euclidean dist.)	Proposed approaches					
		NSC$_M$-GA		NSC$_P$-GA			NSC$_{MD}$-GA
	Number of genes	Number of genes					
	11	9	11	4	5	7	2
33328_at	√	√	√	√	√	√	√
40936_at	√	√	√	√	√	√	√
34320_at	√	√	√	√	√	√	
32551_at	√	√	√	√	√	√	
37157_at	√	√	√		√	√	
36533_at	√	√	√			√	
37954_at	√	√	√			√	
37716_at	√	√	√				
33833_at	√	√	√				
33327_at	√		√				
35823_at	√		√				

Table 7-28 Summary of the sets of selected features obtained from NSC$_{M}$-GA, NSC$_{P}$-GA, NSC$_{MD}$-GA and NSC-GA for Ovarian data

Gene accession number	NSC-GA (Euclidean dist.)	Proposed approaches							
		NSC$_{M}$-GA	NSC$_{P}$-GA				NSC$_{MD}$-GA		
	Number of Peptides	Number of Peptides							
	7	1	2	8	9	10	10	11	20
MZ244.36855	√	√	√	√	√	√	√	√	√
MZ244.66041	√		√	√	√	√	√	√	√
MZ244.95245	√			√	√	√	√	√	√
Z245.24466	√			√	√	√	√	√	√
MZ245.8296	√			√	√	√	√	√	√
MZ245.53704	√			√	√	√	√	√	√
MZ246.12233	√			√	√	√	√	√	√
MZ246.41524				√	√	√	√	√	√
MZ25.589892					√	√	√	√	√
MZ25.49556						√	√	√	√
MZ25.684398								√	√
MZ28.600577									√
MZ220.47402									√
MZ28.700483									√
MZ220.75125									√
MZ29.001246									√
MZ246.70832									√
MZ463.55767									√
MZ463.95962									√
MZ464.36174									√

Table 7-29 Summary of the sets of selected features obtained from NSC$_M$-GA, NSC$_P$-GA, NSC$_{MD}$-GA and NSC-GA for Prostate data

Gene Accession number	NSC-GA (Euclidean dist.)	Proposed approaches		
		NSC$_M$-GA	NSC$_P$-GA	NSC$_{MD}$-GA
	Number of genes	Number of genes		
	6	17	5	5
41468_at	√	√	√	√
37639_at	√	√	√	√
38406_f_at	√	√	√	√
769_s_at	√	√	√	√
556_s_at	√	√	√	√
31444_s_at	√	√		
31527_at		√		
33614_at		√		
39756_g_at		√		
40435_at		√		
40436_g_at		√		
36587_at		√		
36666_at		√		
37720_at		√		
216_at		√		
38429_at		√		
40282_s_at		√		

Table 7-30 shows a summary of classification results associated with using the different sets of features, obtained from NSC-GA, NSC$_M$-GA, NSC$_P$-GA and NSC$_{MD}$-P on the AD, Colon, Leukemia, Lymphoma, Lung, Ovarian and Prostate cancer data, on the corresponding unseen test dataset.

Table 7-30 Summary of classification results for the respective unseen test datasets using the corresponding feature sets obtained using NSC-GA, NSC$_{M}$-GA, NSC$_{P}$-GA and NSC$_{MD}$-GA for each of the seven datasets

Dataset	NSC-GA (Euclidean dist.)		Proposed approaches					
			NSC$_{M}$-GA		NSC$_{P}$-GA		NSC$_{MD}$-GA	
	No of features	Test (%)	No of features	Test (%)	No of features	Test (%)	No of features	Test (%)
AD	11	90.21	18	97.82	9	92.39	4	91.3
Colon	28 6	100	7	93.54	6 42	100	12	100
Leukemia	9	97.05	9	94.12	4 5 24	94.12	3	94.12
lymphoma	7 12 128 129 132	95.45 95.45 100 100 100	3	100	72 73 75 77 80	100	3	100
Lung	8 9 10 11	100	9 11	98.88	4 5 7	100	2	63.33
Ovarian	7	96.06	1	96.06	2 8 9 10	96.85 96.06 96.06 96.06	10 11 20	92.12 92.12 88.97
Prostate	6	90.2	17	100	5	90.2	5	94.12

As seen in Table 7-30, the columns of “No of features” list the number of selected features using the different approaches, NSC-GA, NSC$_{M}$-GA, NSC$_{P}$-GA and NSC$_{MD}$-GA, with the smaller set being a subset of the larger set of features for the corresponding dataset. Also seen from the table, for AD data, NSC$_{M}$-GA approach was the best in terms of achieving the highest classification accuracy, 97.82%, on unseen test data compared with NSC-GA, NSC$_{P}$-GA and NSC$_{MD}$-GA. However, NSC$_{MD}$-GA approach selected a smallest set of features, 4, and classification accuracy of 91.3%, that was higher than the NSC-GA approach and compatible with the NSC$_{P}$-GA approach; for Colon data, NSC$_{P}$-GA outperformed the other approaches in terms of selecting a smallest set of relevant features, 6, with highest classification accuracy of 100%; for Leukemia data, NSC$_{MD}$-GA selected a smallest set of relevant features, 3, and still retained the same classification accuracy of 94.12% as of NSC$_{M}$-GA and NSC$_{P}$-GA using 13, 9 and 5 features respectively, and was compatible with NSC-GA of 97.05 using 9 features. It can be stated that NSC$_{MD}$-GA approach is able to select a smallest set of features and still retain compatible classification accuracy compared to the other approaches for AD and Leukemia data; for Lymphoma data, NSC$_{M}$-GA and NSC$_{MD}$-GA outperformed the other approaches for both in selecting a smallest feature set, 3, with the highest classification accuracy of 100%; for Lung and Ovarian data, NSC$_{P}$-GA showed its best in overall for selecting small set of 4 and 2 features with the highest classification accuracy of 100% and 96.85%, respectively; and for Prostate data, NSC$_{M}$-GA selected the larger set of 17 features with the highest classification accuracy of 100%, NSC$_{MD}$-GA outperformed NSC-GA and NSC$_{P}$-GA for selecting a smallest set of 5 features with higher classification accuracy of 94.12%.

Again, these results showed that the developed techniques support a comprehensive analysis, providing a number of multi-variate signatures for each dataset, each with a varying number of features. Biomedical researchers can make informed decision based on the tradeoffs between classification accuracy and size of feature sets as well as use domain knowledge to decide on the potential relevance of features in the different signatures. An important aspect of the smaller sets being subsets of the larger set also provides some information about the possible correlations/interactions amongst the features and the joint behaviour of these features.

7.3. Summary

This chapter has described the proposed approach of implementing different similarity distance measures in the NSC classifier and incorporating NSC and GA to automatically search for optimal shrinkage threshold values for NSC. The approach used the modified NSC classifier with different distance measure as an evaluator to evaluate the fitness of the candidate shrinkage threshold values, utilized the GA as a search algorithm to search for optimal shrinkage threshold values, and obtained the sets of relevant features. The results obtained shows that the new approaches, NSC$_{M}$-GA,

NSC_P-GA and NSC_{MD}-GA, are able to select smaller set of features and improve classification accuracy compared to the NSC classifier using Euclidean distance.

In the next chapter, the proposed approach of using a multi-objective algorithm to incorporate into the NSC algorithm for searching multiple optimal solutions will be described in details.

8. Incorporating Nearest shrunken centroid and multi-objective evolutionary algorithm for searching multiple shrinkage threshold solutions

8.1. Introduction

In Chapter 5, the approach of incorporating NSC into GA (NSC-GA) was proposed for finding an optimal shrinkage threshold for NSC automatically (Dang et al., 2013). In NSC-GA, the approach of aggregating 2 objective functions as a single objective was implemented for measuring the fitness of chromosomes. In order to optimize a multi-objective problem more effectively and to obtain multiple optimal solutions in a single run, an approach involving MOEA is developed in this study. The non-dominated sorting algorithm (NSGA2) algorithm (Deb *et al.*, 2002) is an example of an MOEA that has been used in bioinformatics. For example, Deb, *et al.* (2002), Deb and Reddy (2003), Mitra and Banka (2006), and Banerjee, Mitra and Banka (2007) employed the NSGA2 algorithm to produce multiple feature sets for Colon, Lymphoma and Leukemia cancer dataset in their studies. One of the advantages of using MOEAs is its ability to evaluate multiple objectives simultaneously in order to find optimal solutions showing good tradeoffs between all objective functions (Deb *et al.*, 2002). For example, Deb and Reddy (2003) employed NSGA2 to analyse the Leukemia cancer dataset and obtained 352 different three-gene sets that gave 100% classification accuracy.

Motivated by 1) the effectiveness of MOEA (NSGA2) in its potential to find multiple solutions, 2) the NSC algorithm in FS and classification, and 3) the automated shrinkage threshold optimization in NSC-GA, a hybrid approach incorporating NSGA2 (Deb *et al.*, 2002) and NSC algorithm (Tibshirani *et al.*, 2002) is proposed in this chapter to automatically find the Pareto front associated with optimal shrinkage threshold values for the NSC. These optimal shrinkage threshold values mapped to potential sets of relevant features for classification. The aim of this study is to see the impact of incorporating a MOEA with NSC with the use of multiple objective functions to evaluate the fitness of chromosomes in the task for obtaining multiple shrinkage threshold solutions. This chapter is an extended version of the paper “NSC-NSGA2: Optimal Search for Finding Multiple Thresholds for Nearest Shrunken Centroid” (Dang

& Lam, 2013). The proposed approach uses NSC as a fitness evaluator in NSGA2 to measure the goodness of feature sets and NSGA2 optimizes the search for multiple solutions. Unlike NSC-GA, where the shrinkage threshold value is selected on the basis of a single objective function which is an aggregation of 2 objective functions, the proposed approach, NSC-NSGA2, supports finding optimal shrinkage threshold values while considering different tradeoffs by simultaneously considering multiple objective functions. The proposed approach is evaluated using the evaluation strategy and the 7 biomedical datasets described in Chapter 3.

Section 8.2, describes the proposed approach, NSC-NSGA2, with evaluation results in Section 8.3, details and results for NSC-NSGA2 using 3 objective functions are described in Section 8.4. Section 8.5 describes the investigation of using Mahalanobis distance in NSC-NSGA2 and followed by the summary in Section 8.6.

8.2. The proposed approach, NSC-NSGA2

Figure 8-1 illustrates the framework of the proposed approach, NSC-NSGA2. There are 2 main steps consisting of:

Step 1: This step carries out the procedure for automatic calculation of Th_{max}. This procedure is performed once only at the beginning of NSC-NSGA2.

Step 2: NSGA2 is employed in this step to search for multiple optimal sets of shrinkage thresholds for NSC algorithm. The NSC algorithm is employed as a fitness evaluator to evaluate the fitness of each chromosome in terms of the number of features selected and its training classification accuracy.

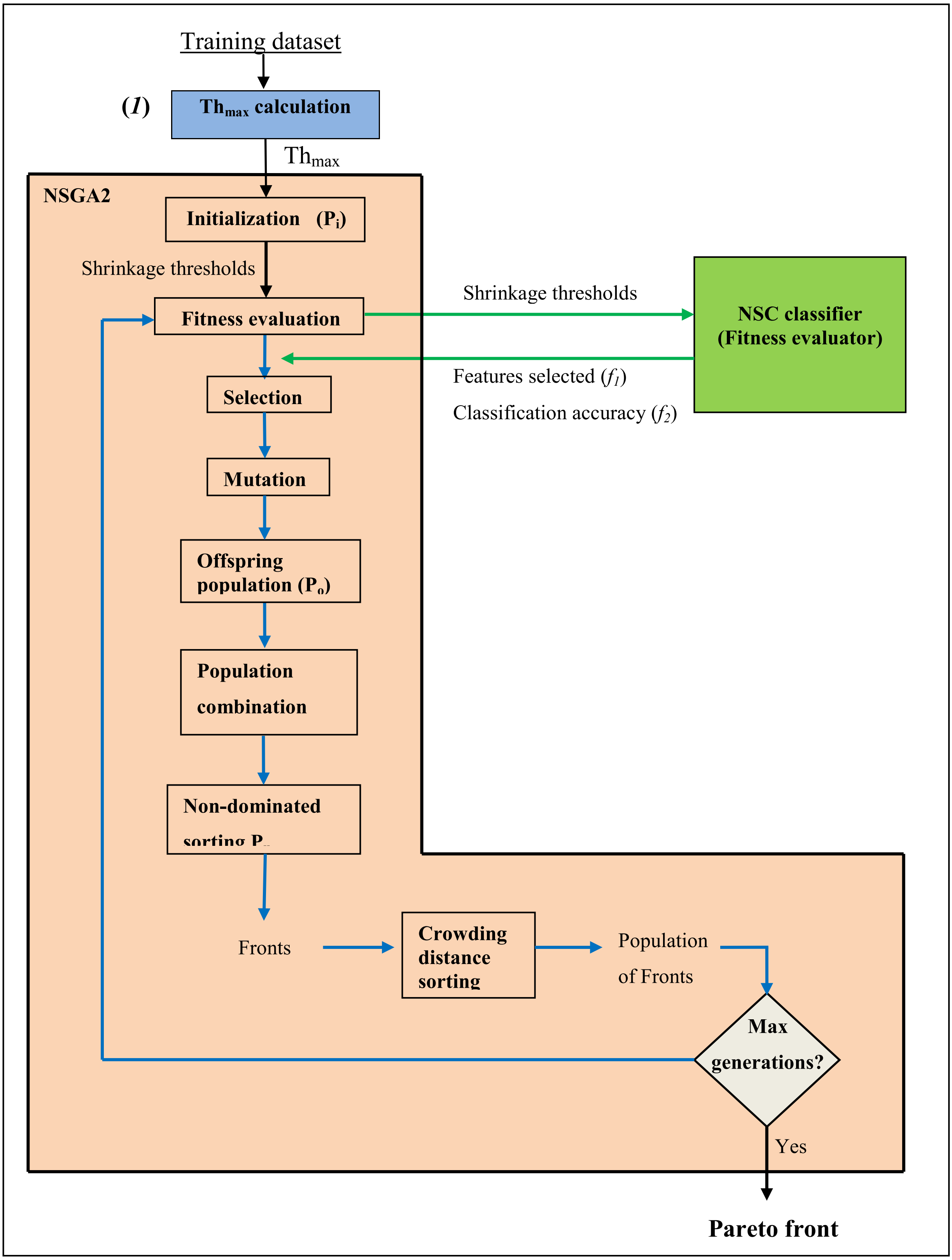


Figure 8-1 Framework of the proposed approach, NSC-NSGA2

8.2.1. Issues related to the proposed approach, NSC-NSGA2

Chromosomes encoding and fitness evaluation are similar to those used in NSC-GA. The same procedure for estimation of the initial range of values for the shrinkage threshold described in Section 5.2.1 is also used in NSC-NSGA2. The following section describes the issues associated with encoding chromosomes and fitness evaluation.

8.2.1.1. Encoding chromosomes

The aim of the proposed approach is to use a MOEA, specifically NSGA2 to find a Pareto front consisting of multiple shrinkage threshold values that are real numbers for NSC. Similar to the NSC-GA, the most appropriate encoding representation for chromosomes in this study would also be a real-encoding. But unlike the NSC-GA approach in which, each chromosome consists of a number of genes (shrinkage thresholds), in NSC-NSGA2, each chromosome consists of a single gene only, representing one shrinkage threshold value.

8.2.1.2. Fitness evaluation using NSC as a fitness evaluator

The NSC algorithm is also employed as a fitness evaluator in the NSC-NSGA2 approach for evaluating the fitness of the chromosomes using the training dataset. The NSC algorithm uses shrinkage threshold values to perform FS and classification. As a result, each shrinkage threshold (chromosome) is associated with a set of features and classification accuracy. To investigate the impact of the approach to using more than 2 objective function, two versions: NSC-NSGA2 and NSC-NSGA2* were implemented involving two and three objective functions respectively. The first two objective functions (f_1 and f_2) have been described in Section 5.3.2.2 and the third objective function (f_3) is defined in Equation 8.6.

The basic concepts of NSC algorithm (Tibshirani *et al*., 2002) has been reviewed in Section 2.3.3. The following sections describe the steps associated with the proposed approach, NSC-NSGA2. The parameters used to run NSC-NSGA2 are also described in

Table 8-1.

8.2.2. Steps of the proposed approach, NSC-NSGA2

8.2.2.1. Step 1: Th_{max} calculation

The same procedure for calculating Th_{max} described in Section 5.2.2.1 is employed here to find the Th_{max} value (upper bound shrinkage threshold value) for the respective dataset.

8.2.2.2. Step 2: Multi-objective evolutionary algorithm search optimization

The study uses NSGA2 and NSC to automatically obtain multiple optimized shrinkage threshold values for finding relevant features for classification. The following section describes Pareto-based MOEA in general and NSGA2 specifically.

The concept of Pareto optimality and dominance as defined by Coello and Lamont (2004), Ayala and Coelho (2008), and Fonseca and Fleming (1995) is:

$x \in X$ is a Pareto optimal if and only if $F(x) = (f_i(x),..,f_k(x))$ is not dominated by $F(x^*) = (f_1(x^*),..,f_k(x^*))$ where $x^* \in X$. A solution x_1 dominates x_2 if and only if $f(x_1)$ less than or equal $f(x_2)$, which means:

$$\forall i \in \{1..k\}, f_i(x_1) \leq f_i(x_2) \wedge \exists i \in \{1..k\} \ : f_i(x_1) < f_i(x_2)$$

if no other solutions dominate x_1, then x_1 is non-dominated. Thus the Pareto front is the set of non-dominated solutions.

Figure 8-2 illustrates the Pareto front with a set of solutions.

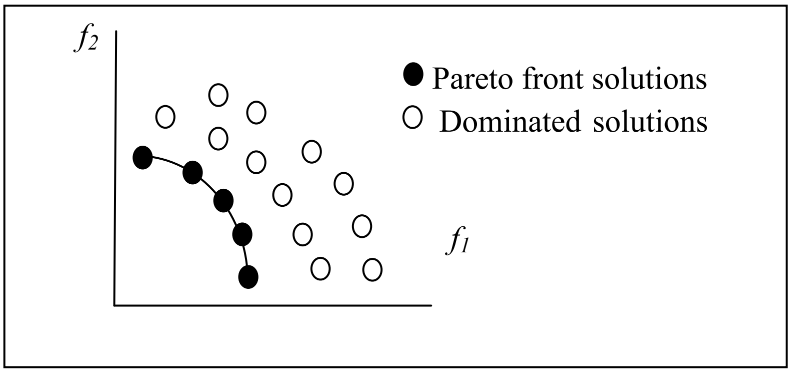


Figure 8-2 Pareto front solutions

According to Ayala and Coelho (2008), a good solution obtained from MOEA must be very close to the Pareto front and is also wide spread. In order to achieve this desire solution, MOEA first needs to find a solution set that is close to the Pareto front as possible and then search through the Pareto front to obtain a set of solution which is more diverse than the other solution sets in the Front.

MOEA selects non-dominated solutions (Pareto front) based on the Pareto ranking. The population is sorted according to Pareto dominance of individuals, and then all the non-dominated individuals are given the same rank which is a higher rank than the dominated individuals. The same rank is given for all non-dominated individuals so that they would have the equal probability of being chosen to reproduce offspring. According to Coello and Lamont (2004), the diversity of the Pareto front is maintained by different strategies such as fitness sharing and niching, clustering, and use of entropy. The use of elitist schemes is very popular in MOEA in recent years. With this elitist approach, a second population is used along with the main population to store the non-dominated solutions found during the evolutionary process. It is also used to improve the diversity of the solutions and to adjust the selection rate of the algorithm (Coello & Lamont, 2004). Another approach of using this elitist approach is to combine the parent population and its offspring population into a single population as in NSGA2 (Deb *et al.*, 2002) to maintain the elitist solutions (Coello & Lamont, 2004). The following section describes steps involving NSGA2 in the proposed approach.

NSGA2 incorporates the concept of Pareto front into MOEA (Deb *et al.*, 2002) which was developed based on NSGA (N. Srinivas & Kalyanmoy, 1994). NSGA2 is an improved version of NSGA in terms of less computational time, incorporating elitism to improve the performance of the algorithm and avoid losing good solutions, and not using a sharing parameter provided by the user (Deb *et al.*, 2002). The following figure illustrates the framework of NSGA2 with the major phases.

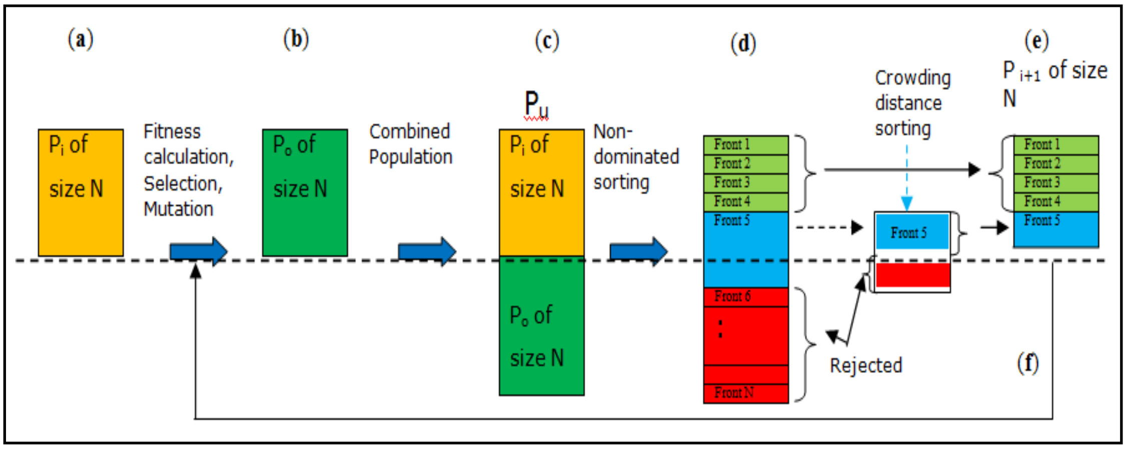


Figure 8-3 Major steps of NSGA2 adapted from Deb *et al.* (2002)

a) Population initialization

After the chromosome representation has been determined and the Th_{max} value has also been calculated, a population of chromosomes is then initialized. Each chromosome (shrinkage threshold value) is initialized to a real value generated randomly in the range [0, Th_{max}] using RNG. Figure 8-4 describes the procedure used to initialize the population and Figure 8-5 shows an example of an initial population.

```
Input:
        $Th_{max}$
        Size of population, $p$
Output:
        An initialized population of $p$ chromosomes
Steps:
    1. Set population ($I_p$) as 1dimensional array of size $p$ of real numbers
    2. Set $I_p = \{\emptyset\}$
    3. For counter from 1 to $p$
        a. Generate a real random number ($R_n$) in the range [1, $Th_{max}$] using a
           RNG
        b. Store $R_n$ to $I_p$[counter]
```

Figure 8-4 Initial population algorithm using RNG

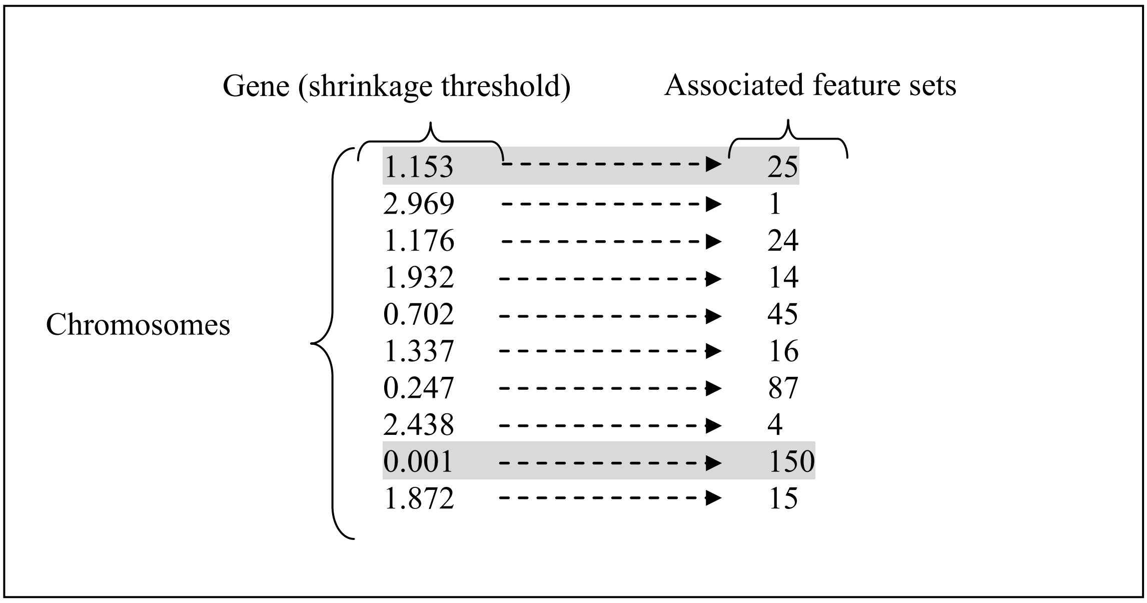


Figure 8-5 An example of an initial population with 10 chromosomes with shrinkage threshold values and the number of features in their corresponding sets

As seen in the example in Figure 8-5, each chromosome consists of only one shrinkage threshold value which has been initialized to be in the range between 0 and Th_{max}, that is [0, 3], Each shrinkage threshold value is associated with a set of features, for example, shrinkage threshold value of 1.153 resulted in a set of the most relevant 25 features and a value of 0.001 resulted in a set of the entire initial 150 features (highlighted row 1 and 9, respectively). Once the initial population has been initialized, the next step is to evaluate the fitness of chromosomes in the population.

b) Fitness evaluation in NSC-NSGA2

In this step, two sub-steps are carried out: firstly, the fitness for each chromosome in the population is calculated using two objective functions: f_1 and f_2 (or three objective functions in the case of NSC-NSGA2*), and secondly, chromosomes in the population are sorted using the non-dominated sorting algorithm (Deb *et al.*, 2002) shown in Figure 8-8.

The NSC algorithm described in Section 2.3.3 is employed here as a fitness evaluator to determine the fitness of the chromosomes associated with a training dataset. To obtain the fitness of the chromosome, firstly, the chromosome (shrinkage threshold) value is used in the NSC algorithm to obtain the corresponding set of features and secondly, this set of features is then used to construct a classifier that is used to classify the training data. The set of features and its classification result are then used in the calculation of the fitness of the chromosome. The two objective functions (f_1 and f_2) have been described in Section 5.3.2.2 and are shown as follows.

$$f_1 = (N_{total} - N_{att}) / N_{total} \tag{8.1}$$

$$f_2 = \frac{TP+TN}{TP+FP+TN+FN} \tag{8.2}$$

Objective function f_1 is designed for maximizing the fitness of chromosomes (solutions) that has a minimum number of features, i.e., the smaller the number of features selected the better the fitness for the chromosome, f_2 is designed for maximizing the fitness of chromosomes that has highest training classification accuracy, i.e., the higher the training classification accuracy the better the fitness for the chromosome.

c) Selection and mutation operators

Selection and mutation operators for real encodings are used in NSC-NSGA2. The *crowded tournament* selection is employed in NSGA2 for chromosome selection. The *crowded tournament* selection is a binary tournament selection with different selection criteria based on the rank and crowding distance of the chromosomes. That is, two chromosomes are selected randomly from the population to form a tournament group (i.e. the size of tournament group is two for a binary tournament) and the best chromosome of the group is selected based on the fitness ranked by the non-dominated sorting procedure (Deb *et al.*, 2002; Suzuki *et al.*, 1995). In the case of two chromosomes with different ranks then choose the one with a better rank. Otherwise, if the two chromosomes have the same rank, then the *crowding distance* algorithm is employed to calculate the crowding distance of the chromosomes and the chromosome with a smaller crowding distance is chosen (Deb *et al.*, 2002; Suzuki *et al.*, 1995). Tournament selection has been described in Section 2.4.4.2, the *crowded tournament* selection and the *crowding distance* algorithm (Deb *et al.*, 2002) are described in Figure 8-6 and Figure 8-7, respectively.

```
Input:
        Population front ($P$)
        Population ranks ($P_{rank}$)
Output:
        Best chromosome ($C_{best}$)
Steps:
    1. Set $k$ = size of binary tournament = 2
    2. For counter from 1 to $k$
            • Select a chromosome ($C_1$) randomly from $P$
            • Select a chromosome ($C_2$) randomly from $P$
    3. Compare the rank of $C_1$ and $C_2$ using $P_{rank}$
        i. If rank of $C_1$ = rank of $C_2$
            • Perform crowding distance algorithm
            • Select the chromosome ($C_{best}$) with a smaller crowding distance
        ii. Else
            • Select a chromosome ($C_{best}$) with the best rank
```

Figure 8-6 *Crowded tournament* selection algorithm used in Deb *et al.* (2002)

```
Input:
        Population ($P$)
        Objective functions $f\,[f_1..f_n]$
        Number of objective functions ($N_f$)
Output:
        Individual crowding distance
Steps:
    1. For each $n$ individual in $P$
            Initialize Individual distance ($I_d$) = 0
    2. For counter from 1 to $N_f$
        a. Sort P based on $f$
        b. Set $I_{d1} = I_{dn} = \infty$
    3. For counter = 2 to ($n$ -1)
            $I_{di} = I_{di} + ((I(i+1).m - I(i-1).m)/(\int_m^{max} - \int_m^{min}))$
                (where I(i).m = value of $m_{th}$ objective function of the $k_{th}$
                individual in i)
```

Figure 8-7 *Crowding distance* algorithm used in Deb *et al.* (2002)

Note that in Step 2b, the two boundary solutions, i.e., solutions with smallest and largest objective function values, are assigned a value of infinite distance (∞) so that the boundary solutions are always selected. Step 3 in Figure 8-7 is used to calculate the

Euclidean distance for the remaining solutions, i.e., solutions between the boundary solutions.

Gaussian distribution probability is employed as the mutation operator to modify the value of a gene in a chromosome. When mutating a single real value, Gaussian probability distribution function is first used to get a number and then adding it to the value being mutated to produce a new number (Hedvat *et al.*, 2003). The calculation of probability distribution, $P(x)$, for a value x is defined by Equation (8.3) and the mutation value, N_{mut} , is calculated by Equation (8.4).

$$P(x) = \left(\frac{1}{\sigma\sqrt{2\pi}}\right) e^{-\left(\frac{(x-\mu)^2}{2\sigma^2}\right)} \tag{8.3}$$

$$N_{mut} = x + P(x) \tag{8.4}$$

where x is a value of the gene (threshold), σ is a standard deviation, μ is a mean of the value.

σ =1 and μ=0 when mutating a chromosome with only one gene value, hence Equation (8.3) can be rewritten using Equation (8.5).

$$P(x) = \left(\frac{1}{\sqrt{2\pi}}\right) e^{-\left(\frac{x^2}{2}\right)} \tag{8.5}$$

d) Offspring population (P_o) generation

The new offspring produced from one cycle consisting of the selection and mutation process are then placed into the offspring population. The process of creating new offspring via the cycle consisting of the selection and mutation process is repeated until the population for new offspring of size N (the same size of parent population size) is obtained.

e) Union (combination) of two populations (P_i and P_o)

Two populations (in the same generation), the first being the parent population (P_i) of size N and the second being the offspring population (P_o) also of the size N, are then combined to make a larger population of size 2N, P_u.

f) Rank the individuals in the combined population, P_u, using the non-dominated sorting algorithm (Figure 8-8). All the best chromosomes of rank #1 are placed in Front #1, all the next best chromosomes of rank #2 are placed in Front #2, etc., and when the sorting algorithm has found a sufficient number of fronts having a specified number of chromosomes for the new population, it stops the sorting process (Deb *et al.*, 2002).

```
For each individual p in population P
        Initialise $S_p = \emptyset$
                  $n_p = 0$
        For each individual q in population P
           If p dominates q then
               Add q to $S_p$
           Else if q dominates p then
               $n_p = n_p+1$
           if $n_p=0$
             $p_{rank} = 1$
             add p to $Front_1$
        Set Front counter i =1
        While $Front_i \neq \emptyset$
          Set Q = $\emptyset$
           For each p in $Front_i$
             For each q in Sp
              $n_q = n_q-1$
                 if $n_q = 0$
                $q_{rank} = i+1$
                add q to Q
          i = i+1
          set $Front_i = Q$
```

Figure 8-8 Non-dominated sorting procedure used in Deb *et al.* (2002)

g) Generate a new population (P_{i+1}) of size N

After the ranked chromosomes in P_u were sorted into Fronts on the basis of their respective ranks, a new population (P_{i+1}) of size N is then created by populating it, starting with chromosomes from the front with the highest rank. The process continues to incorporate chromosomes, taken from a descending order of ranked fronts. In the event that there are more chromosomes in the ultimate Front to be included for completing a population of size N, chromosomes in this front are sorted using the crowding distance procedure first and the remaining slots in the population are then filled with the required number of “best chromosomes” from this front. Figure 8-9 showed the steps involved in generating the new population.

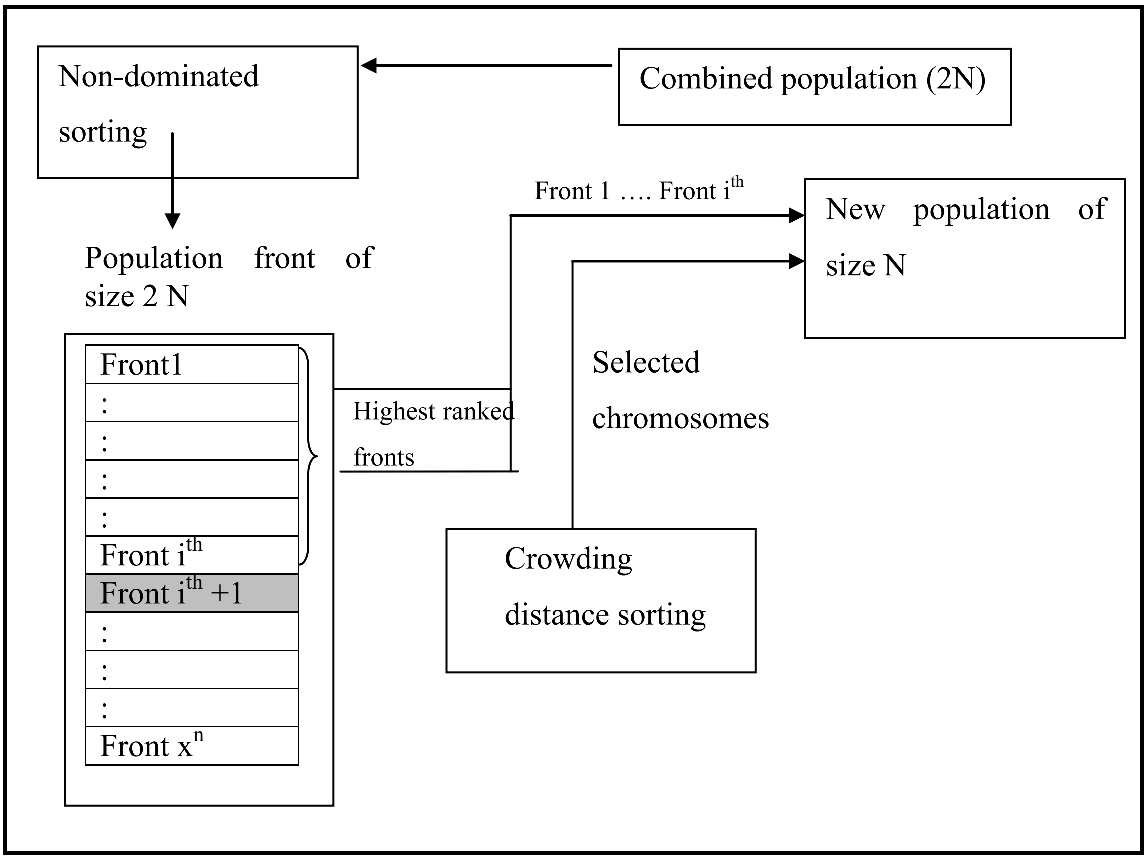
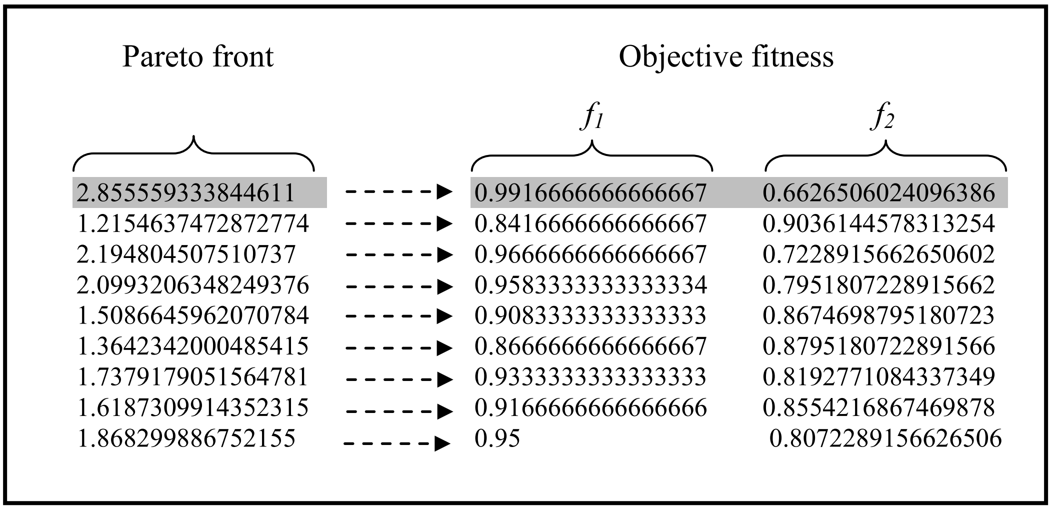


Figure 8-9 Steps for generating the new population from the combined population

h) Repeat the process

The new population P_{i+1} undergoes the next iteration consisting of all steps described above, i.e., from fitness evaluation to the step for generation of a new population. These iterations of steps are repeated until the termination condition is satisfied (i.e. the predefined maximum number of generations has been executed). A Pareto front is the output. The following figure shows an example of a Pareto front of shrinkage threshold solutions with their associated objective function values.



Pareto front		Objective fitness f_1	Objective fitness f_2
2.855559333844611	- - - - -▶	0.9916666666666667	0.6626506024096386
1.2154637472872774	- - - - -▶	0.8416666666666667	0.9036144578313254
2.194804507510737	- - - - -▶	0.9666666666666667	0.7228915662650602
2.0993206348249376	- - - - -▶	0.9583333333333334	0.7951807228915662
1.5086645962070784	- - - - -▶	0.9083333333333333	0.8674698795180723
1.3642342000485415	- - - - -▶	0.8666666666666667	0.8795180722891566
1.7379179051564781	- - - - -▶	0.9333333333333333	0.8192771084337349
1.6187309914352315	- - - - -▶	0.9166666666666666	0.8554216867469878
1.868299886752155	- - - - -▶	0.95	0.8072289156626506

Figure 8-10 An example of Pareto front of 9 shrinkage threshold solutions

Figure 8-10 shows an example of a Pareto front consisting of 9 shrinkage threshold value solutions listed in the “Pareto front” column with their associated objective function of f_1 and f_2 listed in the last 2 columns (Objective fitness column). For example for the 1st shrinkage threshold value, 2.8555 (highlighted) in the shrinkage threshold value column having associated f_1=0.9916 and f_2=0.6626 (highlighted) in the objective fitness column.

8.2.3. Parameter settings for NSGA2

According to Deb *et al*. (2002), the mutation rate used in their study was based on 1/*n* where *n* is the number of attributes. In this study, since the chromosome has only one attribute (shrinkage threshold value), the algorithm relies solely on a mutation operator to generate new offspring. To adapt to this situation the mutation rate of 1/*n* is used where n is the population size (Goldberg, 1989). The algorithm was executed with the population size of 100, and mutation rate of 0.01, i.e., 1/100. The complete set of parameter settings used in this study is shown in Table 8-1. As each chromosome consists of a single gene, crossover operations are not applicable.

Table 8-1 Parameter set used for NSC-NSGA2

Parameters	Values / Methods
Population Size	100
Chromosome Length - Real encoding	1
Mutation Probability	1 / Population size = 0.01
Generation	1000
Selection	Crowded tournament
Mutation	Gaussian probability distribution

8.3. Experiment results

Experiments were carried out to evaluate the proposed approach, NSC-NSGA2, in terms of obtaining the Pareto front of shrinkage threshold values associated with the NSC for the datasets described in Section 3.1. For each of the 7 datasets, 15 independent runs of the proposed approach were executed using the respective training data. For each run, a stratified 10 fold CV described in Section 3.2 was employed. Each shrinkage threshold solution on the Pareto front obtained from each run is used as input to the NSC algorithm to obtain its corresponding feature set. This feature set was then used to construct the corresponding NSC classifier to classify the unseen test data associated with the dataset. Where appropriate, the comparison of the performance of the proposed algorithm with existing work is based on classification accuracy and the selected feature sets.

Two common characteristics are applicable across the results from the evaluation of the approach using each of the seven datasets. These are:

- The classification results using each of the NSC classifiers on the respective unseen test dataset from each run are first recorded and the reported classification accuracy in the tables was an average of classification accuracy of these classifiers over the 15 independent runs.
- In terms of the selected feature sets that were obtained as part of the evaluation, the smaller feature set is a subset of the larger feature set. For example, in Table

8-2, the set with 18 features is the subset of the set with 19 features and similarly, the set with one feature is both a subset of the set of 18 as well as the set of 19 features.

8.3.1. Ray *et al.* AD data

The proposed algorithm, NSC-NSGA2, was executed 15 times with 10 fold CV on AD dataset using the NSGA2 parameter setting listed in Table 8-1. The results obtained from the 15 independent runs consists of 8 runs where their Pareto fronts has 10 shrinkage thresholds, 3 runs with Pareto fronts of 9 shrinkage thresholds, 3 runs with Pareto fronts of 8 shrinkage thresholds, and 1 run with Pareto fronts of 7 shrinkage thresholds. Using these shrinkage thresholds led to selected sets of features consisting of 1, 4, 5, 6, 7, 8, 9, 10, 11, 15, 16, 17, 18 and 19 features. The convergence plot with a typical Pareto optimal front from one of the 15 runs is shown in Figure 8-11 and the NSC classification results using each of these sets of features on the unseen test dataset are shown in Table 8-3.

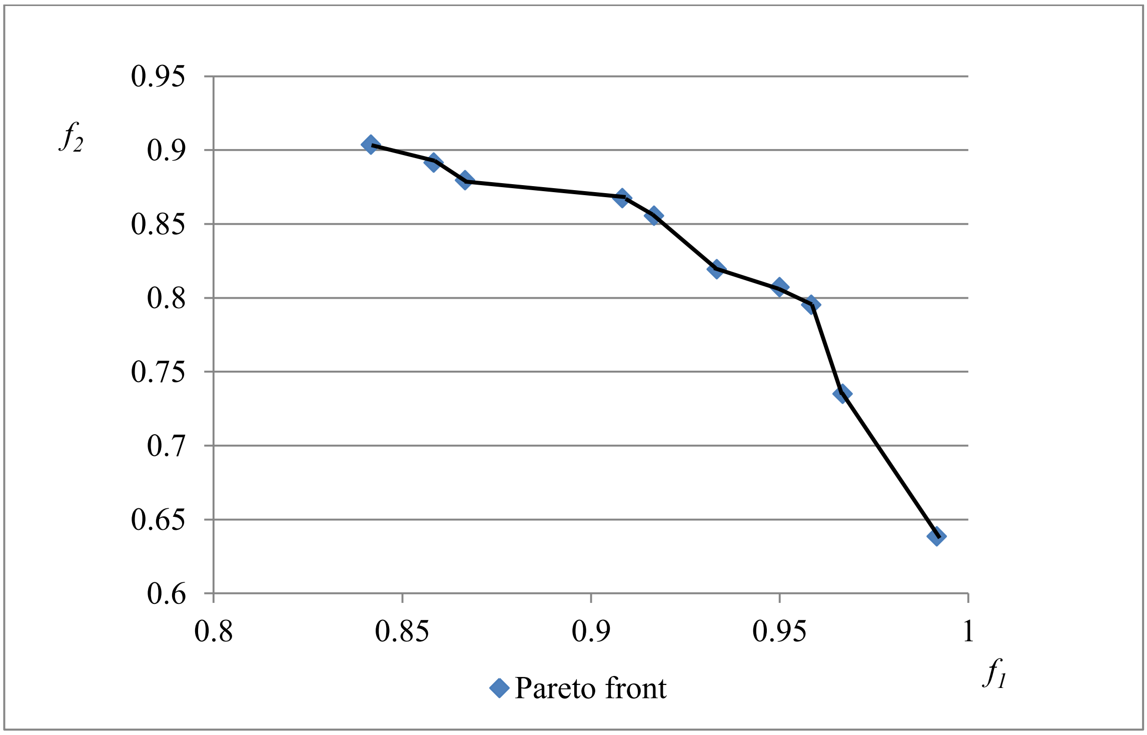


Figure 8-11 A typical Pareto front plot of objective function *f1* against f_2 for AD dataset

As seen in Table 8-2, the proposed approach found a set of 18 and 19 features which are the same set of 18 and 19 features found in (Ray *et al*., 2007). The approach also found the same set of 11 features reported in Chapter 5 using the NSC-GA approach.

Table 8-2 Sets of selected proteins using the proposed approach, NSC-NSGA2, for AD data

Proteins	Protein sets													
	19	18	17	16	15	11	10	9	8	7	6	5	4	1
PDGF-BB_1	√	√	√	√	√	√	√	√	√	√	√	√	√	√
RANTES_1	√	√	√	√	√	√	√	√	√	√	√	√	√	
IL-1a_1	√	√	√	√	√	√	√	√	√	√	√	√	√	
TNF-a_1	√	√	√	√	√	√	√	√	√	√	√	√	√	
EGF_1	√	√	√	√	√	√	√	√	√	√	√	√		
M-CSF_1	√	√	√	√	√	√	√	√	√	√	√			
ICAM-1_1	√	√	√	√	√	√	√	√	√	√				
IL-3_1	√	√	√	√	√	√	√	√	√					
IL-11_1	√	√	√	√	√	√	√	√						
GCSF_1	√	√	√	√	√	√	√							
ANG-2_1	√	√	√	√	√	√								
PARC_1	√	√	√	√	√									
GDNF_1	√	√	√	√	√									
TRAIL R4_1	√	√	√	√	√									
IL-8_1	√	√	√	√	√									
MIP-1d_1	√	√	√	√										
IGFBP-6_1	√	√	√											
MCP-3_1	√	√												
MDC_1	√													

Table 8-3 Classification results for the AD data using the sets of selected features from NSC-NSGA2 approach

Number of proteins	Average classification Accuracy (%) (Unseen Test data)
1	56.98
4	72.82
5	76
6, 7	81.52
9	82.6
8	82.78
10	83.4
11	87.7
17	91.3
16	91.63
15	92.39
18	93.84
19	94.56

From the results it can also be seen that NSC-NSGA2 produced a number of potential feature sets that demonstrates the tradeoffs between the numbers of selected features and the classification accuracy for the unseen test data. For example, the smallest feature set (with 1 feature), the resulting classifier has the lowest classification accuracy for the unseen test data (56.98%), whilst the largest feature set (19 features) the resulting classifier has the highest test classification accuracy (94.56%) on the unseen test data. This type of analysis provides more information than univariate statistics and biomedical researchers can use it to gain a better understanding of the possible correlations amongst the features as well as the joint behaviour of features in their datasets.

8.3.2. Alon *et al.* Colon cancer data

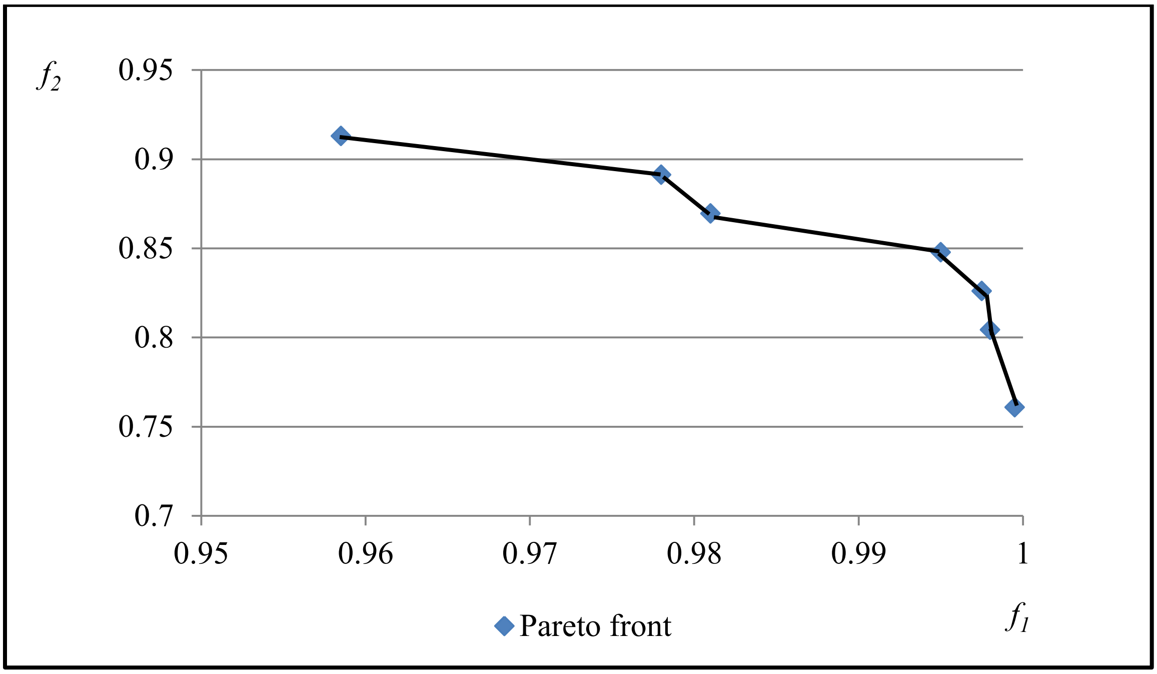


Figure 8-12 A typical Pareto front plot of objective function f_1 against f_2 for the Colon cancer dataset

Using the same experimental procedure described above, the proposed algorithm is evaluated using the Colon dataset. A typical convergence plot with a Pareto optimal front consisting of 7 solutions is shown in Figure 8-12. NSC-NSGA2 found optimal shrinkage threshold values from the 15 independent runs that consist of 8 runs with a Pareto front of 5 shrinkage thresholds, 4 runs with a Pareto front of 6 shrinkage thresholds, 2 runs with a Pareto front of 4 shrinkage thresholds and 1 run with a Pareto front of 7 shrinkage thresholds. Using these shrinkage thresholds led to selected sets of features consisting 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 21, 23, 38, 39, 40, 42, 43, 44, 45, 47, 48, 61, 62, 77, 83, 85, 87, 89 and 92 features (genes). Classification results associated with classifiers constructed from these sets of selected features are shown in Table 8-4. An interesting point here is that classifiers constructed using feature sets that are supersets of the set of 23 features all performed worse than those classifiers constructed from feature sets that are subsets of the set of 23 features. The set of 9 genes includes known biomarkers associated with Colon cancer from the literature. These are M76378, J02854, M63391, Z50753, T71025, R87126, U25138, M82919 and T92451 (highlighted genes in Table 8-4). Note that Table 8-4 lists only the sets that have up to

23 genes but the evaluation has been done for all sets of features obtained using NSC-NSGA2 and shown in Table 8-5.

Table 8-4 Sets of genes selected using the proposed approach, NSC-NSGA2 for Colon cancer data

Gene accession number	Gene sets											
	23	21	10	9	8	7	6	5	4	3	2	1
M76378	√	√	√	√	√	√	√	√	√	√	√	√
J02854	√	√	√	√	√	√	√	√	√	√	√	
M63391	√	√	√	√	√	√	√	√	√	√		
Z50753	√	√	√	√	√	√	√	√	√			
T71025	√	√	√	√	√	√	√	√				
R87126	√	√	√	√	√	√	√					
U25138	√	√	√	√	√	√						
M82919	√	√	√	√	√							
T92451	√	√	√	√								
M76378	√	√	√									
Z24727	√	√										
M76378	√	√										
T56604	√	√										
H43887	√	√										
R36977	√	√										
X86693	√	√										
X63629	√	√										
M36634	√	√										
T67077	√	√										
H06524	√	√										
T60778	√	√										
H67764	√											
X12671	√											

Table 8-5 Classification results for the Colon cancer data using the sets of selected features from NSC-NSGA2 approach

Number of genes	Average classification Accuracy (%) (Unseen Test data)
61 , 87, 62, 86	62.5
77, 83, 85, 92, 89	68.75
38 , 48	81.25
39, 40, 42, 43, 44, 45, 47	87.5
1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 21, 23	93.75

8.3.3. Leukemia cancer data

Using the same experimental procedure as before, the proposed algorithm was evaluated using the Leukemia cancer dataset. A typical Pareto optimal front from one of the 15 independent runs is shown in Figure 8-13.

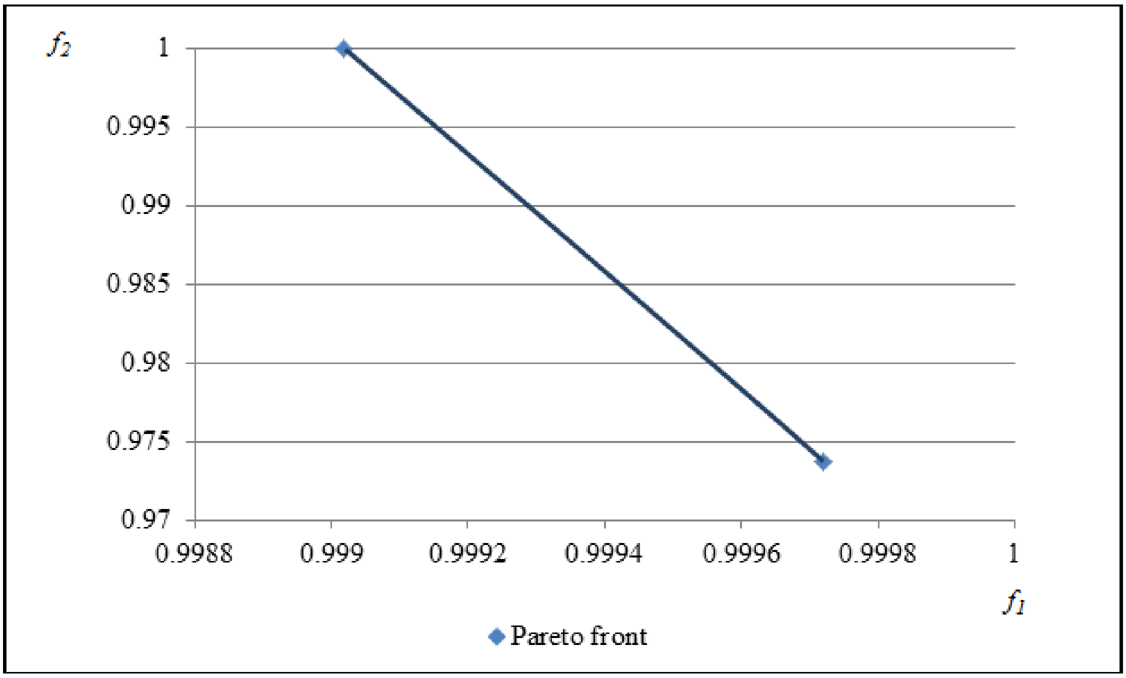


Figure 8-13 A typical plot of a Pareto front of objective function f_1 against f_2 for the Leukemia cancer dataset

The Shrinkage threshold solutions obtained from the 15 independent runs led to selected sets of features consisting 2, 7, 9, 10, 11 and 13 features. The sets of 2, 7, 9, 10, 11 and 13 features are listed in Table 8-6. Five genes are associated with known Leukemia biomarkers in the literature; namely M84526_at, U50136_mal_at, D49950_at, M16038_at and X17042_at (highlighted genes in Table 8-6). Classifiers constructed using the set with 2 and 13 genes produced the same average classification accuracy, 91.18%, on unseen test data, and classifiers constructed using the set of 7, 9, 10 and 11 genes produced the same average classification accuracy of 94.11%. The set of 13 genes having seven genes in common from the set of 9 genes reported in NSC-GA [7].

Table 8-6 Sets of genes selected using the proposed approach, NSC-NSGA2 for Leukemia cancer data

Gene accession number	Gene sets					
	13	11	10	9	7	2
M84526_at	√	√	√	√	√	√
U50136_rna1_at	√	√	√	√	√	√
D49950_at	√	√	√	√	√	
M16038_at	√	√	√	√	√	
M23197_at	√	√	√	√	√	
X17042_at	√	√	√	√	√	
X95735_at	√	√	√	√	√	
M55150_at	√	√	√	√		
M57710_at	√	√	√	√		
Y00787_s_at	√	√	√			
M27891_at	√	√				
U82759_at	√					
M28130_rna1_s_at	√					

Table 8-7 Classification results for the Leukemia cancer data using the sets of selected features from NSC-NSGA2 approach

Number of genes	Average classification Accuracy (%) (Unseen Test data)
2, 13	91.18
7, 9, 10, 11	94.12

8.3.4. Ovarian cancer data

Using the same experimental procedure as before, the proposed algorithm was evaluated using the Ovarian cancer dataset. A typical convergence plot of Pareto optimal front with 5 solutions is shown in Figure 8-14.

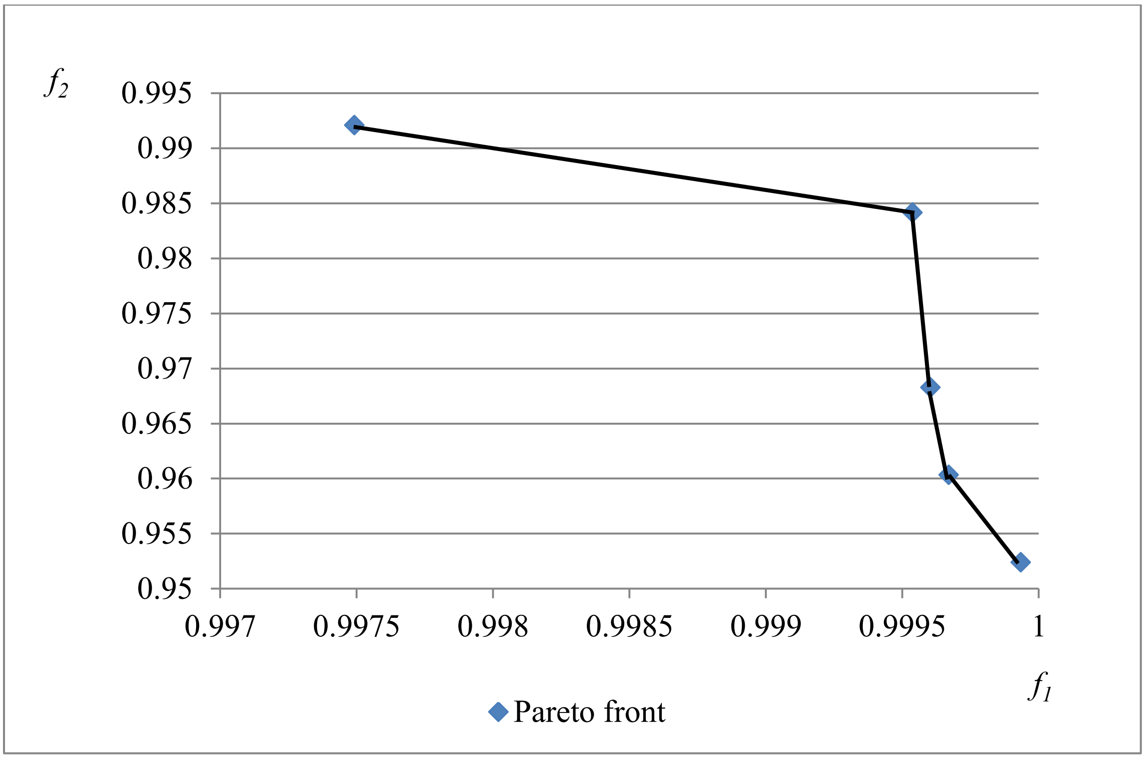


Figure 8-14 A typical Pareto front plot of f_1 against f_2 for Ovarian cancer dataset

The results obtained from the 15 independent runs consist of 8 runs, each with a Pareto front of 5 shrinkage thresholds; 4 runs, each with a Pareto front of 3 shrinkage thresholds and 3 runs, each with a Pareto front of 4 shrinkage thresholds. Using these shrinkage thresholds led to selected sets of features consisting 1, 5, 6, 7, 36, 37, 38, 207, 210, 212, 224, 227 and 230 features. The sets of 1, 5, 6, and 7 features are listed in Table 8-8. Classifiers constructed using the sets with 1, 5 and 6 peptides produced the same average classification accuracy, 96.85% , on the unseen test data, and the classifier constructed using the set of 7 peptides gives 96.06%. The approach also found the same set of 7 features reported in Chapter 5 using the NSC-GA approach, which is associated with known ovarian peptide biomarkers in the literature.

Table 8-8 Subsets of genes selected using the proposed approach, NSC-NSGA2 for Ovarian cancer data

Gene accession number	Gene sets			
	7	6	5	1
MZ245.24466	√	√	√	√
MZ244.66041	√	√	√	
MZ244.95245	√	√	√	
MZ245.53704	√	√	√	
MZ245.8296	√	√	√	
MZ244.36855	√	√		
MZ246.12233	√			

Table 8-9 Classification results for the Ovarian cancer data using the sets of selected features from NSC-NSGA2 approach

Number of Genes	Average classification accuracy (%) (Unseen Test data)		Overall average classification accuracy (%) (Unseen Test data)
	C1(Disease)	C2 (Normal)	
207, 210, 212, 224, 227	88.89	91.3	89.76
230	90.06	91.3	90.55
7, 36, 37, 38	97.53	93.48	96.06
1 , 5, 6	97.65	95.65	96.85

The column headings *C1* and *C2* in the table stand for average classification accuracy (%) on the Ovarian unseen test dataset for the *Disease* class and *Normal* class, respectively. The column heading “Overall average classification accuracy” stands for the overall average classification (%) for both classes on the Ovarian unseen test dataset for the 15 independent run. “Overall average classification accuracy” is calculated using Equation (5.3). From Table 8-9, the NSC classification results associated with the sets of features mostly showed similar levels of specificity and sensitivity, e.g., sensitivity (C1) is 97.65% and specificity (C2) is 95.65%, implying truly *not-at-risk* and *at-risk* cases will be correctly identified at a very high level of accuracy.

8.3.5. Lymphoma cancer data

Using the same experimental procedure as before, the proposed algorithm was evaluated using the Lymphoma cancer dataset. A typical convergence plot of Pareto optimal front with 4 solutions is shown in Figure 8-15.

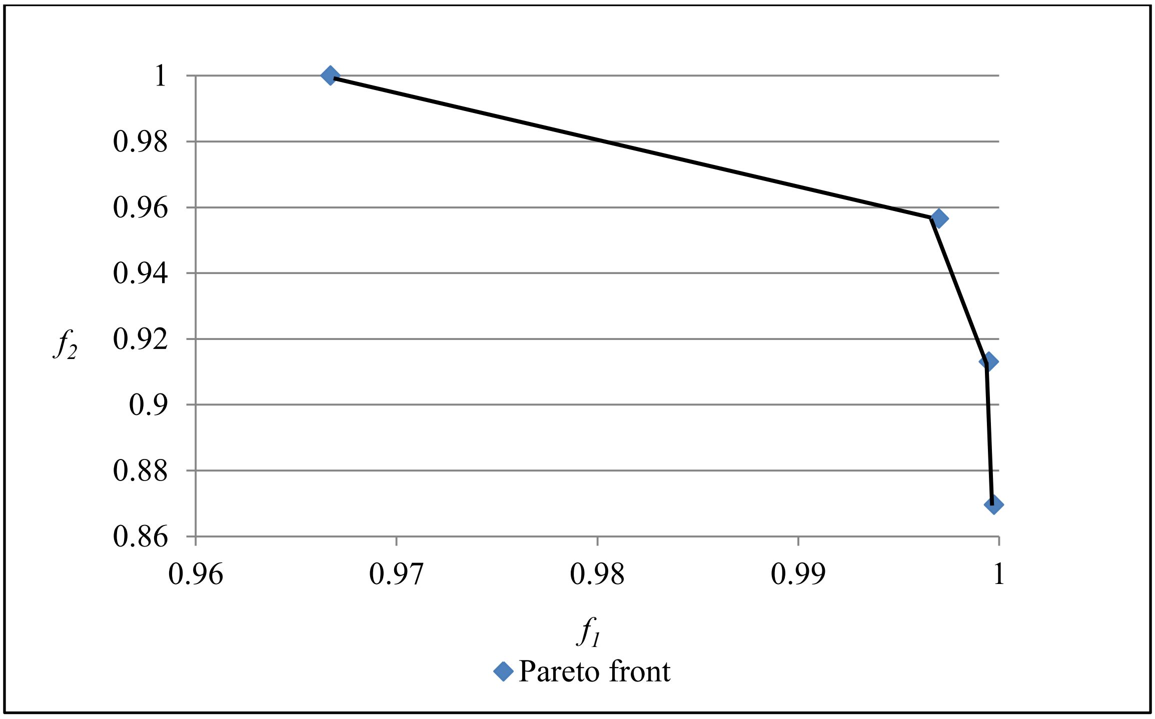


Figure 8-15 A typical Pareto front plot of f_1 against f_2 for Lymphoma cancer dataset

The results obtained from the 15 independent runs consist of 14 runs, each with a Pareto front of 4 shrinkage thresholds and 1 run with a Pareto front of 3 shrinkage thresholds.

Using these shrinkage thresholds led to selected sets of features consisting 1, 2, 7, 8, 12, 128, 133, 134 137, 139, 140, 141, 146, 149 and 164 features. The sets of 1, 2, 7, 8 and 12 features are listed in Table 8-10. Classifiers constructed using the set with 1 feature produced 68.18% average classification accuracy for the unseen test data, classifiers obtained using the set with 2 features produced 77.72%, classifiers constructed using the set with 7, 8 and 12 features produced the same average classification accuracy of 95.45%, and classifiers obtained using the set with 128, 133, 134, 137, 139, 140, 141, 146, 149 and 164 features produced 100% respectively. The approach also found the same set of 7, 12 and 128 features reported in Chapter 5 using the NSC-GA approach.

Table 8-10 Subsets of genes selected using NSC-NSGA2 for Lymphoma cancer data

Gene accession number	Gene sets				
	12	8	7	2	1
GENE3361X	√	√	√	√	√
GENE3329X	√	√	√	√	
GENE3327X	√	√	√		
GENE3330X	√	√	√		
GENE3332X	√	√	√		
GENE3258X	√	√	√		
GENE3256X	√	√	√		
GENE3328X	√	√			
GENE3314X	√				
GENE3260X	√				
GENE1252X	√				
GENE3967X	√				

Table 8-11 Classification results for the Lymphoma cancer data using the sets of selected features from NSC-NSGA2 approach

Number of genes	Average classification Accuracy (%) (Unseen Test data)
1	68.18
2	72.73
7, 8, 12	95.45
128, 133, 134, 137, 139, 140, 141, 146, 149, 164, 173	100

8.3.6. Lung cancer data

Using the same experimental procedure as before, the proposed was evaluated using the Lung cancer dataset. A typical convergence plot of Pareto optimal front with 3 solutions is shown in Figure 8-16.

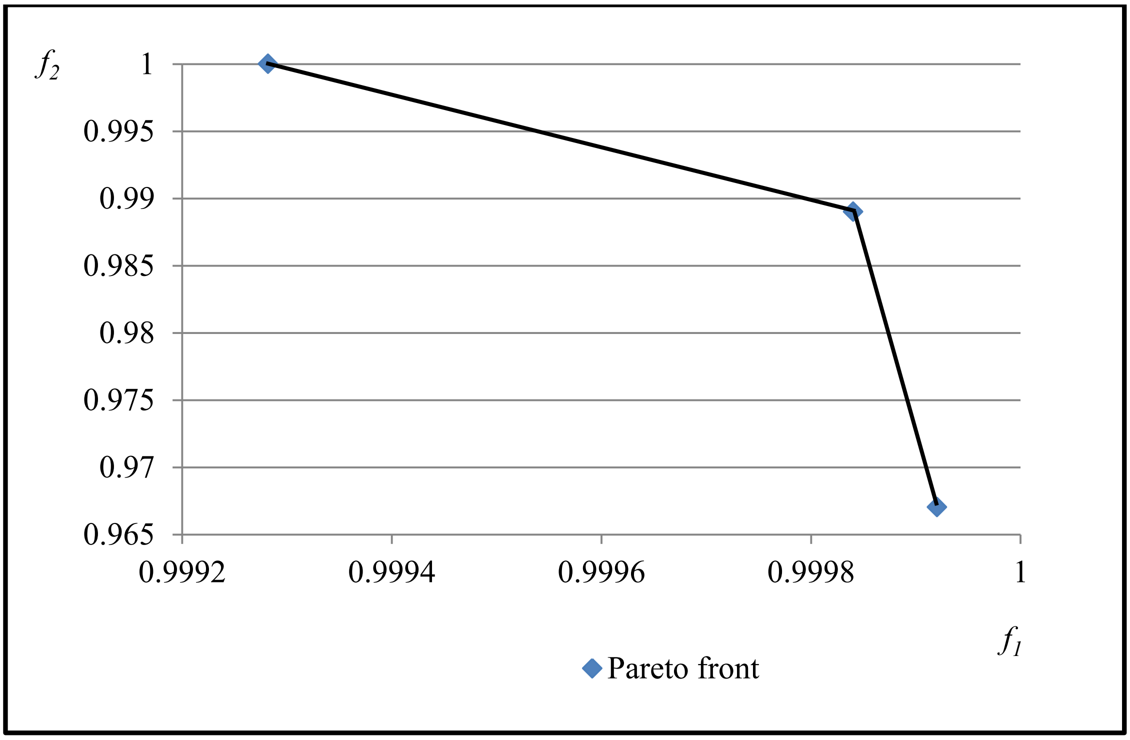


Figure 8-16 A typical Pareto front plot of f_1 against f_2 for Lung cancer dataset

The results obtained from the 15 independent runs consist of 14 runs, each with a Pareto front of 3 shrinkage thresholds and 1 run with a Pareto front of 4 shrinkage thresholds. Using these shrinkage thresholds led to selected sets of features consisting 1, 2, 3, 5, 8, 9 and 11 features. The sets of 1, 2, 3, 5, 8, 9 and 11 features are listed in Table 8-12. Classifiers constructed using the set with 1 feature produced 93.63% average classification accuracy for the unseen test data, classifiers obtained using the set with 2 features produced 94.62%, classifiers obtained using the set with 3 features produced 95.93%, and classifiers constructed using each of the sets with 5, 8, 9 and 11 features respectively, produced 100% respectively. The approach also found the same set of 8, 9 and 11 features reported in Chapter 5 using the NSC-GA approach.

Table 8-12 Subsets of genes selected using the proposed approach, NSC-NSGA2 for Lung cancer data

Gene accession number	Gene sets						
	11	9	8	5	3	2	1
40936_at	√	√	√	√	√	√	√
33328_at	√	√	√	√	√	√	
32551_at	√	√	√	√	√		
34320_at	√	√	√	√			
37157_at	√	√	√	√			
36533_at	√	√	√				
37716_at	√	√	√				
37954_at	√	√	√				
33833_at	√	√					
35823_at	√						
33327_at	√						

Table 8-13 Classification results for the Lung cancer data using the sets of selected features from NSC-NSGA2 approach

Number of genes	Average classification Accuracy (%) (Unseen Test data)
1	93.63
2	94.62
3	95.93
5, 8, 9, 11	100

8.3.7. Prostate cancer data

As mentioned previously in Section 3.1.6, the Prostate dataset consists of 12600 attributes, 77 Tumour (T) and 59 Normal (N) samples. The training set consisting of 52 T and 50N samples, and the unseen test set consisting of 25 T and 9 N samples.

Using the same experimental procedure as before, the proposed algorithm was evaluated using the Prostate cancer dataset. A typical convergence plot of Pareto optimal front with 4 solutions is shown in Figure 8-17.

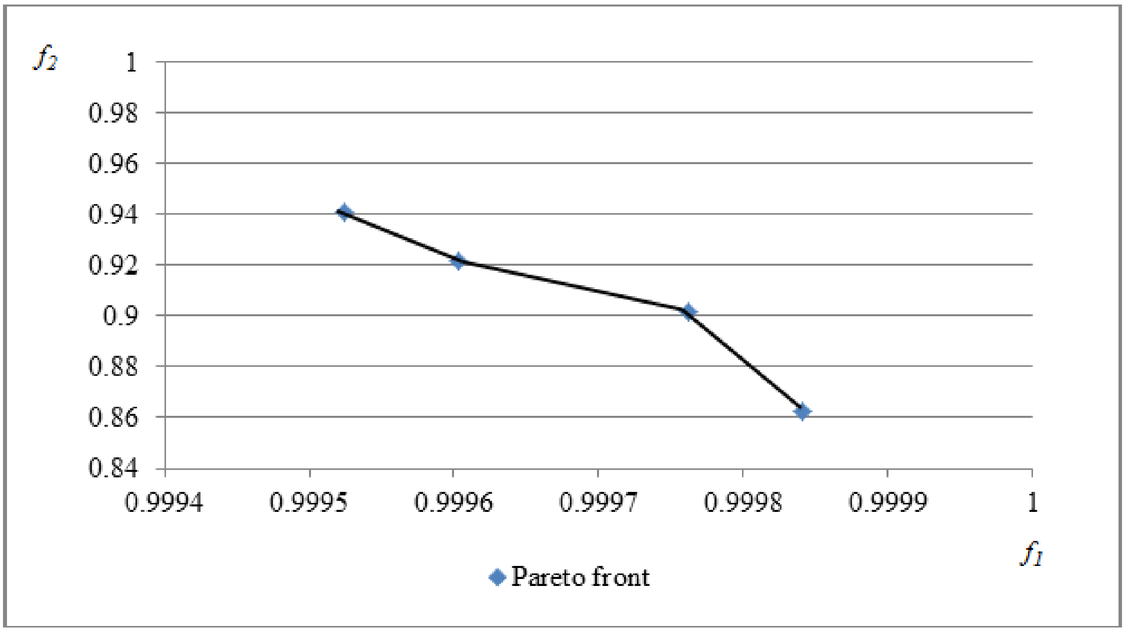


Figure 8-17 A typical Pareto front plot of f_1 against f_2 for Prostate cancer dataset

The results obtained from the 15 independent runs consist of 3 runs with a Pareto front of 4 shrinkage thresholds, 4 runs with a Pareto front of 3 shrinkage thresholds and 8 runs with a Pareto front of 2 shrinkage thresholds. Using these shrinkage thresholds led to selected sets of features consisting 1, 2, 3, 4, 5, 6 and 8 features. The sets of 1, 2, 3, 4, 5, 6 and 8 features are listed in Table 8-14. Classifiers constructed using the set with 1 feature produced 78.43% average classification accuracy for the unseen test data, classifiers obtained using the set with 2 features produced 82.48%, classifiers constructed using the set with 3 and 4 features produced 88.24% respectively, classifiers obtained using the set with 5 features produced 89.8%, classifiers obtained using the set with 6 features produced 90.2% and classifiers obtained using the set with 8 features produced 92.16%. The approach also found the same set of 6 features reported in Chapter 5 using the NSC-GA approach.

Table 8-14 Subsets of features selected using the proposed approach, NSC-NSGA2 for Prostate cancer data

Gene accession number	Gene sets						
	8	6	5	4	3	2	1
38406_f_at	√	√	√	√	√	√	√
37639_at	√	√	√	√	√	√	
41468_at	√	√	√	√	√		
769_s_at	√	√	√	√			
556_s_at	√	√	√				
31444_s_at	√	√					
39532_at	√						
31527_at	√						

Table 8-15 Classification results for the Prostate cancer data using the sets of selected features from NSC-NSGA2 approach

Number of genes	Average classification Accuracy (%) (Unseen Test data)
1	78.43
2	82.48
3	88.24
4	88.24
5	89.8
6	90.2
8	92.16

The following section compares the NSC-NSGA2 approach with NSC-GA from the perspectives of potential sets of features obtained via both approaches. Table 8-16 lists the sets of features (in terms of the number of features) and the average classification accuracy of their corresponding classifiers for the corresponding unseen test datasets.

Table 8-16 Summary of results for NSC-GA and NSC-NSGA2

Dataset	**NSC-GA ($f = f_1 + f_2$)**		**NSC-NSGA2 (f_1 and f_2)**	
	Number of features	Average Classification Accuracy (%) (Unseen Test data)	Number of features	Average Classification Accuracy (%) (Unseen Test data)
AD	11	89.45	1	56.98
			4	72.82
			5	76
			6, 7	81.52
			9	82.6
			8	82.78
			10	83.4
			11	87.7
			17	91.3
			16	91.63
			15	92.39
			18	93.84
			19	94.56
Colon	6	93.75	1, 2, 3, 4, 5, 6, 7, 8, 9, 10,	93.75
			38 , 48	81.25
			39, 40, 42, 43, 44, 45, 47	87.5
			61 , 87, 62, 86	62.5
			77, 83, 85, 92, 89	68.75
Leukemia	9	97.06	2, 13	91.18
			7, 9, 10, 11	94.12
Ovarian	7	96.06	1 , 5, 6	96.85
			7, 36, 37, 38	96.06
			207, 210, 212, 224, 227	89.76
			230	90.55
Lymphoma	7	95.45	1	68.18
			2	72.73
			7, 8, 12	95.45
			128, 133, 134, 137, 139, 140, 141, 146, 149, 164, 173	100
Lung	8	100	1	93.63
			2	94.62
			3	95.93
			5, 8, 9, 11	100
Prostate	6	90.2	1	78.43
			2	82.48
			3, 4	88.24
			5	89.8
			6	90.2
			8	92.16

Table 8-16 shows results obtained in Chapter 5 for NSC-GA where the objective function was an aggregation of the same two objective functions for NSC-NSGA2, that is, $f = 0.5\, f_1 + 0.5\, f_2$. In this formulation, both objective functions were given equal weightings and using a GA, single optimal sets of relevant features were obtained at the end of each run for AD, Colon, Leukemia, Ovarian, Lymphoma, Lung and Prostate cancer datasets. With the proposed approach in this chapter, NSC-NSGA2, two objective functions (f_1 and f_2) are assessed simultaneously and multiple optimal sets of relevant features are obtained for each dataset at the end of each run.

Having information as shown in Table 8-16 with regards to the joint classification behaviour of various sets of features allows the domain expert to make informed decision in terms of sets of features that would be selected for further investigations. For example in the case of the AD dataset, one can make decisions based on the tradeoffs between classification accuracy and size of feature set. The set of 6 features resulted in the same classification accuracy as the set of 7 features (i.e. 81.52%). The domain expert can examine the 7th feature and use domain knowledge to decide on it potential relevance and make decision on subsequent analysis. Equally it is interesting to further analyse the Colon cancer dataset where sets with 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 21 and 23 features respectively resulted in classifiers producing the same classification accuracy on the unseen test dataset (93.75%). It appears that a major contributing factor relates to 1 feature and thus may warrant further investigations into the relevance of the remaining features. A similar situation can also be seen with the Leukemia cancer dataset where sets with 7, 9, 10 and 11 features respectively resulted in classifiers returning the same classification (94.12%) on the unseen test dataset (a major contributing factor relates to 7 features); with Ovarian cancer dataset where sets with 1, 5 and 6 features, respectively, resulted in classifiers returning the same classification (96.85%) on the unseen test dataset (a major contributing factor relates to 1 features), and sets with 7, 36, 37 and 38 features, resulted in classifiers returning the same classification (96.06%) on the unseen test dataset (a major contributing factor relates to 7 features); with Lymphoma cancer dataset where sets with 7, 8 and 12 features, respectively, resulted in classifiers returning the same classification (95.45%) on the unseen test dataset (a major contributing factor relates to 7 features); with Lung cancer dataset where sets with 5, 8, 9 and 11 features, respectively, resulted in classifiers returning the same classification (100%) on the unseen test dataset (a major contributing factor relates to 5 features), and with Prostate cancer dataset where sets with 3 and 4 features, respectively, resulted in classifiers returning the same classification (88.24%) on the unseen test dataset (a major contributing factor relates to 3 features). This sort of information for analysis in bioinformatics is important as reducing the number of features to a smaller promising set for further investigations would reduce costs associated with future experiments and analysis.

8.4. NSC-NSGA2 with 3 objective functions

To further examine the proposed approach, the following section detailed work that investigated the impact of employing more than 2 objective functions for FS. In addition to f_1 and f_2, a 3rd objective function (*f3*) is also employed in NSC-NSGA2, and this is denoted as NSC-NSGA2*. f_3 is calculated using Equation (8.6).

$$f_3 = \left(\sum_1^k \frac{\sum_1^n d'_{ik}}{n}\right)/k \qquad (8.6)$$

where d'_{ik} is the positive shrunken relative difference of selected features

n is the total number of features selected

k is the number of classes

f_3 is an average of d'_{ik} for selected features

f3 is designed for maximizing the fitness of chromosomes that has a maximum shrunken relative difference, d'_{ik}, for the features selected. As mentioned previously, in NSC, the class centroid of attributes is shrunk toward the overall class centroid and attributes with at least one positive relative shrunken class centroid are considered as important and are selected (i.e. class centroids and overall class centroid are different). The attributes can be ranked based on the value of d'_{ik}. That is, the larger the value of d'_{ik} the better the rank of attributes. Therefore, *f3* is employed in the proposed approach to maximize the set that consists of attributes with better ranks, (i.e. the best overall average value of d'_{ik} for the set), with the aim to improve the fitness evaluation for chromosomes that leads to the selection of smaller feature sets with the same or higher classification accuracy.

Table 8-17 Result from applying 2 and 3 objective functions for the proposed approach, NSC-NSGA2, on AD and Leukemia dataset

Dataset	**NSC-NSGA2** (f_1, f_2)		**NSC-NSGA2*** (f_1, f_2,f_3)	
	Number of features	Average Classification Accuracy (%) (Unseen Test data)	Number of features	Average classification Accuracy (%) (Unseen Test data)
AD	1	56.98	1	56.07
	4	72.82	4	72.83
	5	76	5	75.47
	6, 7	81.52	6	79.98
	9	82.6	7	81.52
	8	82.78	8	83.02
	10	83.4	9	82.60
	11	87.7	10	83.52
	17	91.3	11	89.72
	16	91.63	12	90.21
	15	92.39	13, 14	92.39
	18	93.84	16	91.67
	19	94.56	17	91.3
			18, 19	94.56
			33	92.39
			38	90.21
			67	85.86
Leukemia	2, 13	91.18	2, 3, 11, 13, 15	91.18
	7, 9, 10, 11	94.12	7	92.02
			8, 10	94.12
			62	85.29
			176, 336, 884, 889	88.23

From the limited analysis using the AD dataset with 120 features and the Leukemia dataset with 7129 features, it can be seen from Table 8-17 that the NSC-NSGA2* approach resulted in a bigger number of different sets of selected features when compared to the approach with 2 objective functions (NSC-NSGA2). Among these additional sets of selected features, some sets are smaller but have the same average NSC classification accuracy, e.g., for AD dataset, the sets with 13 and 14 features using the 3 objective approach that gave the same classification accuracy (92.39%) as that of the set with 15 features using the 2 objective approach, for Leukemia dataset, the set with 8 features using the NSC-NSGA2* approach that gave the same classification accuracy (94.12%) as that of the sets with 9, 10 and 11 features using the NSC-NSGA2 approach.

8.5. The proposed approach, NSC-NSGA2 with Mahalanobis distance measure

According to Bandyopadhyay and Saha (2013, p. 60), "*similarity measurement is essential for performing classification*", thus in order to investigate the impact of employing a different similarity distance measure in the NSC classifier on the Pareto front obtained from NSC-NSGA2, the study carried out a further experiment to replace Euclidean distance in the NSC classifier with Mahalanobis distance. This is one of the most common distance measures that has been used for feature-based similarity search, specifically in datasets where correlation exists between features (Emrich *et al*., 2013).

Using the same experimental procedure outlined in Section 8.3, NSC$_M$-NSGA2 (NSC with Mahalanobis distance) was evaluated using the Leukemia cancer dataset and the NSGA2 parameter settings listed in Table 8-1 as a proof of concept. A typical Pareto optimal front from one of the 15 runs is shown in Figure 8-18.

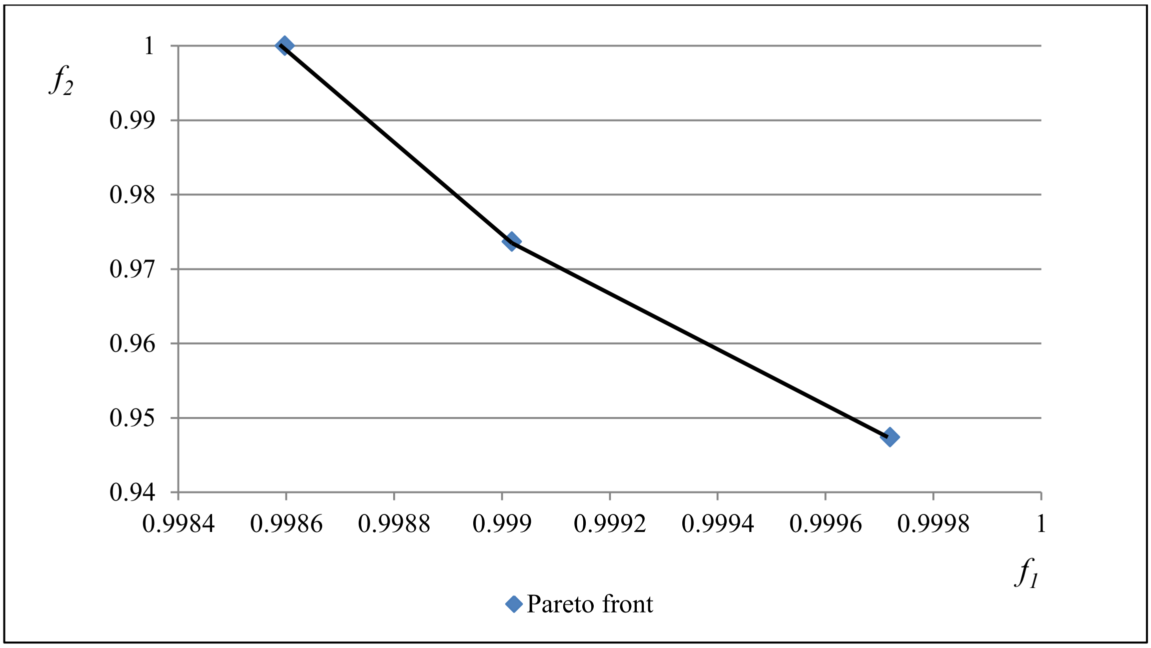


Figure 8-18 A typical Pareto front plot of f_1 against f_2 for Leukemia cancer dataset using the NSC$_M$-NSGA2 approach

The results obtained from running 15 independent runs of NSC$_{M}$-NSGA2 led to the selection of sets with 2, 5, 6, 7, 8, 9, 10, 11 features.

Table 8-18 Sets of genes obtained by NSC$_{M}$-NSGA2 for the Leukemia cancer data

Gene accession number	Gene sets							
	11	10	9	8	7	6	5	2
M84526_at	√	√	√	√	√	√	√	√
U50136_rna1_at	√	√	√	√	√	√	√	√
D49950_at	√	√	√	√	√	√	√	
M16038_at	√	√	√	√	√	√	√	
M23197_at	√	√	√	√	√	√	√	
X17042_at	√	√	√	√	√	√		
X95735_at	√	√	√	√	√			
M55150_at	√	√	√	√				
M57710_at	√	√	√					
Y00787_s_at	√	√						
M27891_at	√							

As seen in Table 8-18, sets of 2, 5, 6, 7, 8, 9, 10 and 11 genes obtained are the same as those using NSC-NSGA2. The sets of 5, 6 and 8 genes are additional sets obtained using NSC$_{M}$-NSGA2.

Table 8-19 Results from NSC-NSGA2 and NSC$_{M}$-NSGA2 for Leukemia cancer dataset

Dataset	Euclidean NSC-NSGA2 (f_1 and f_2)		Mahalanobis NSC$_{M}$-NSGA2 (f_1 and f_2)	
	Number of genes	Average Classification Accuracy (%) (Unseen Test data)	Number of genes	Average Classification Accuracy (%) (Unseen Test data)
Leukemia	2, 13	91.18	2 , 5	88.24
			6, 9	94.12
	7, 9, 10, 11	94.12	7, 8	97.06
			10, 11	100

The classification results using classifiers constructed using these sets of features on the unseen test data are shown in Table 8-19. It is interesting to note that NSC$_M$-NSGA2 did not produce the set of 13 features associated with NSC-NSGA2 and that there are some differences in the classification accuracy associated with the different classifiers from the two approaches. For example, the classifiers constructed from the set of 7 and 8 features obtained the same average classification accuracy of 97.06% in comparison to the classifier constructed using the set of 9 features from NSC-NSGA2 that obtained 94.12% average classification accuracy. Similar to other analysis in this study, the optimal shrinkage threshold values obtained from NSC$_M$-NSGA2 can be slightly different from those obtained using NSC-NSGA2 but are still in the range that map to the same set of features but may slightly impact on the classification accuracy. From this limited analysis here, it can be seen that the use of another similarity measure in the approach can produce some different sets of features. This implies that to comprehensively analyse biological datasets, researchers should examine them using techniques that support different similarity measures and a number of selection criteria. Having information in Table 8-19 can help biomedical researchers to make informed decisions about sets of features that would be selected for further investigations. For example it will be interesting to examine the set of 6 features obtained via NSC$_M$-NSGA2 and the set of 7 features obtained via NSC$_M$-NSGA2 in terms of the 7th feature (set of 6 being the subset of 7 features) in terms of its known biological relevance to the specific disease.

8.6. Summary

This chapter has described the proposed approach of incorporating NSC and MOEA (NSGA2) to automatically search for multiple optimal shrinkage threshold values for NSC. The approach used NSC as an evaluator to evaluate the fitness of the candidate shrinkage threshold values, utilized the MOEA (NSGA2) as a multi-objective search algorithm to search for Pareto front of multiple shrinkage threshold values that lead to the selection of corresponding sets of relevant features. The proposed approach was evaluated using 7 public biomedical datasets: AD, Colon, Leukemia, Ovarian, Lymphoma, Lung and Prostate cancer data. The proposed approach shows the effectiveness of using a multi objective approach, NSC-NSGA2, over a single aggregated objective function used in NSC-GA involving a single objective approach as described in Chapter 5.

This chapter has also described work that incorporated 3 objective functions in NSC-NSGA2 for studying the impact of using more than two objective functions for FS. This approach was evaluated using the AD and Leukemia dataset. The results from the study showed that the approach NSC-NSGA2* obtained a bigger number of different sets of selected features when compared to the approach using 2 objective functions (NSC-NSGA2). In some cases, NSC-NSGA2* obtained some sets having a smaller number of features that produced classifiers that obtained the same average NSC classification accuracy as those associated with a classifier constructed from a superset of features.

To examine the impact of using a different similarity measure, this study implemented $\mathrm{NSC_M}$-NSGA2 where Mahalanobis distance is used in NSC instead of Euclidian distance. The approach was evaluated using the Leukemia cancer dataset. The results showed that some additional sets of features were produced and their corresponding classifiers produced similar classification results. This implies that to comprehensively analyse biological datasets, researchers need to examine them using techniques that support different similarity measures and a number of selection criteria.

9. Conclusion and Future work

This thesis presented the investigation of evolutionary-based FS techniques for analysing biological datasets acquired via mass throughput technologies. These biological datasets are typically high dimensional with only a small number of samples; making the task of their analysis especially challenging. Section 9.1 summarises the key findings from this study and Section 9.2 outlines suggestions for future work.

9.1. Conclusion

As the area of bioinformatics become increasingly "data rich", the need for appropriate techniques that can be used for a comprehensive analysis of these huge volumes of data is imperative. This thesis contributed towards a better understanding of the development of evolutionary-based FS techniques for analysing biological data from mass throughput technologies. The thesis also demonstrated the impact of employing different similarity measure in NSC and showed the need to consider classifier-biased when examining the sensitivity and specificity associated with a specific classifier constructed from a set of features.

This study has addressed the following aims:

- **Aim 1**: To investigate and develop FS algorithms that incorporates various evolutionary strategies, specifically investigating the use of evolutionary strategies in conjunction with Rough Set Theory and Nearest Shrunken Centroid;
- **Aim 2**: To evaluate the developed algorithms in terms of finding the "most relevant" biomarkers contained in biological datasets and
- **Aim 3**: To evaluate the *goodness* of extracted feature subsets for relevance (examined in terms of existing biomedical domain knowledge and classification accuracy form the perspectives of sensitivity and specificity associated with different classifiers). The project aims to generate sets of features for construction of good predictive models for classifying diseased samples from control.

In addressing Aim 1, this study has developed evolutionary-based FS techniques that incorporated GA, MA and MOEA. The first approach involved the development of RST-GA, a hybrid approach involving the GA, RST and k-means. This approach is described in Chapter 4 and consisted of 3 phases: feature reduction, distinction table generation and FS via GA optimization. In the first phase, features of high dimensional data were reduced effectively. In this phase, quartile statistics was employed to generate initial starting centroids for k-means clustering. The final centroids obtained from k-means were used for the feature reduction process. In the second phase, the criteria (i.e. generation rules) used in Banerjee, Mitra, & Banka's study (2007) was also applied to generate a distinction table with a smaller dimension. Finally, in the third phase, GA was employed to search for optimal feature sets based on the distinction table generated in the previous phase. The study showed that the smaller feature sets obtained using RST-GA produced classifiers that gave similar classification accuracy for the Colon cancer dataset and the Leukemia data in comparison to the results reported in Banerjee, *et al*. (2007).

A second approach described in Chapter 5, NSC-GA, incorporates NSC and GA to automatically search for an optimal range of shrinkage threshold values for the NSC. The NSC is a deterministic FS algorithm which selects the same set of features for shrinkage threshold values in the same range. The optimal shrinkage thresholds are used in NSC to obtain the corresponding sets of selected features. The study showed that the feature sets obtained using NSC-GA are smaller. Corresponding classifiers constructed from these feature sets produced similar or higher classification accuracy for seven datasets in comparison with other NSC-based approaches reported in previous studies. While the sets of relevant features obtained using the NSC-GA from every independent run is more consistent, multiple sets consisting of features where the smaller sets are subsets of the bigger sets were also obtained from the runs of the NSC-GA. This is important in terms of allowing biomedical researchers to investigate the sets of features for biological relevance in subsequent clinical studies.

To continue the exploration of evolutionary approaches for FS in biological data, Chapter 6 described an approach MA for automatically finding optimal shrinkage thresholds for NSC in an attempt to further improve upon NCS-GA. The aim was to explore improvements that can be made on the NSC-GA approach. The impact of incorporating MA with NSC for finding shrinkage threshold values automatically are reduced computational time and obtaining the same feature set over different runs of NSC-MA. Chapter 8 described NSC-NSGA2, a hybrid approach incorporating NSGA2 and NSC to automatically find the Pareto front associated with optimal shrinkage threshold values for the NSC. Unlike GA which involved a single objective function, the aim here was to examine the impact of incorporating a MOEA with NSC with the use of multiple objective functions for obtaining multiple shrinkage threshold solutions. Unlike existing techniques, the developed approaches here support FS by simultaneously considering tradeoffs between a number of criteria (e.g. high classification accuracy and a small number of features). Multiple sets of potential features (biomarkers) obtained via the developed approach can be further investigated to explore both diagnostic and biological relevance.

Lastly, this study examined the impact of using different similarity measures in NSC-GA and NSC-NSGA2. Euclidean distance is the distance measure originally used in NSC to assign data points to different classes. Chapter 7 described the approach of implementing different similarity distance measures (i.e. Mahalanobis, Pearson and Mass distance) in the NSC classifier and incorporating NSC and GA to automatically search for optimal shrinkage threshold values for NSC. The use of distance measures such as Mahalanobis overcomes some of the limitations associated Euclidean distance (e.g. assumption that the features are uncorrelated). From the perspective of using a different distance measure in a multi-objective approach, $\mathrm{NSC_M}$-NSGA2 was implemented using Mahalanobis distance in NSC instead of Euclidian distance. As a proof of concept, it was evaluated using the Leukemia cancer dataset. Additional sets of selected features were obtained and their corresponding classifiers produced similar classification results. This implies that to comprehensively analyse biological datasets, researchers need to examine them using techniques that support different similarity measures and a number of selection criteria.

In addressing Aim 2 and Aim 3, seven datasets and the evaluation strategy described in Chapter 3 were used to evaluate the developed approaches in this study. The dimensionality of these datasets ranged from 120 to 15,154 attributes. In terms the relevance and the “goodness” of the selected sets of features, these were evaluated by constructing different classifiers using the suite of classifiers from WEKA and examining the corresponding classification accuracy on unseen test datasets (in terms of diagnostic relevance). From these analyses, the study demonstrated that the use of specific classifiers may have an impact on the sensitivity and specificity obtained using a set of features in classification and recommended that in DM for finding suitable sets of biological markers, a number of classifiers should be employed to examine the diagnostic relevance. This will avoid incidences of dismissing sets of features with high discriminatory capabilities that should be further investigated in early diagnostic test developments. From the perspective of biological relevance, this study is limited to examining the relevance of the extracted feature sets against known biomarkers from literature associated with the relevant domains. For example, Table 4-3 listed genes found by the RST-GA approach which are already known in the biomedical literature to be associated with the Colon Cancer. The common features selected across different approaches for the seven datasets are listed in Table 9-1.

Table 9-1 Common features selected across difference approaches

Datasets	Approaches					
	NSC-GA	NSC$_{M}$-GA	NSC$_{P}$-GA	NSC$_{MD}$-GA	NSC-MA	NSC-NSGA2
	Common features					
AD	PDGF-BB_1, RANTES_1, IL-1a_1, TNF-a_1					
Colon	T71025, M63391, R87126, M76378, T92451, J02854,					
Leukemia	M27891, M84526, M96326					
Lymphoma	GENE3327X, GENE3329X, GENE3361X					
Lung	33328_at, 40936_at					
Ovarian	MZ244.36855, MZ244.66041, MZ244.95245, Z245.24466, MZ245.8296, MZ245.53704, MZ246.12233					
Prostate	41468_at, 37639_at, 38406_f_at, 769_s_at, 556_s_at					

Since the primary theme being the investigation of evolutionary approaches for analysis of biological datasets, the NSC-GA approach was developed to first explore the use of GA to find the shrinkage threshold value for NSC. The next logical step from NSC-GA was to investigate how this technique can be further improved, leading to the development of NSC-MA, the use of memetic algorithm. Another venue of improvements for the NSC relates to impact of similarity measures used and the investigation here led to the development of NSC$_M$-GA, NSC$_p$-GA and NSC$_{MD}$-GA. Finally, NSC-NSGA2 was developed to explore the use of multiple objectives for feature selection, improving upon the previous approaches in this study that involved a single objective.

The following table also summarises the advantages and limitations of the proposed approaches.

Table 9-2 Advantages and limitation of the developed approaches

Proposed approaches	Advantages	Limitations
RST-GA	• Number of attributes is reduced before applying the GA. • Less computational time	Feature instability
NSC-GA	• Feature stability • Explore interaction of features	More computational time compared to the RST-GA approach.
NSC-MA		
NSC$_M$-GA		
NSC$_P$-GA		
NSC$_{MD}$-GA		
NSC-NSGA2	• Multiple sets of features obtained in one run • Feature stability • Explore interaction of features	

9.2. Future work

Future directions from this research could examine:

- Investigations and development of FS techniques that combines RST and different evolutionary algorithms such as MA and other MOEA approaches for analysing biological datasets. Existing work involving evolutionary-based RST for analysis of biological data is limited, probably owing to its computational intensiveness.
- Incorporation of RST into the developed approaches of NSC-GA, NSC$_M$-GA, NSC$_P$-GA and NSC$_{MD}$-GA to reduce computational time of these approaches. Here, RST can be used as a feature reduction algorithm to reduce the number of features for high dimensional data as an initial step before the NSC-GA approach is used to optimize the search of optimal sets of features.

- Due to time constrains, the investigations involving MOEA has only examined NSGA2 and a maximum of 3 objectives. Subsequent investigations could examine the use of other MOEAs and the impact of employing more than 3 objective functions.
- NSC-based approaches are very much targeted for analysis of bioinformatics data, extended to being applied to other high dimensional biological data generated using other techniques. Potentially, the RST-GA approach can be applied to any domain for feature selection.

References

Ahn, Chang Wook, & Ramakrishna, RS. (2010). A diversity preserving selection in multiobjective evolutionary algorithms. *Applied Intelligence, 32*(3), 231-248.

Ahujaa, Ravindra K, Orlinb, James B, & Tiwaric, Ashish. (2000). A greedy genetic algorithm for the quadratic assignment problem. *Computers & Operations Research, 27*, 917-934.

Alander, Jarmo T. (1992). *On optimal population size of genetic algorithms.* Paper presented at the CompEuro'92.'Computer Systems and Software Engineering', Proceedings.

Alba, Enrique, Garcia-Nieto, José, Jourdan, Laetitia, & Talbi, E-G. (2007). *Gene selection in cancer classification using PSO/SVM and GA/SVM hybrid algorithms.* Paper presented at the Evolutionary Computation, 2007. CEC 2007. IEEE Congress on.

Albrechtsson, tord jonson1 elin, Axelson, jan, Heidenblad, markus, Ludmilagorunova, bertil johansson, & Höglund, mattias. (2001). Altered expression of TGFB receptors and mitogenic effects of TGFB in pancreatic carcinomas. *INTERNATIONAL JOURNAL OF ONCOLOGY, 19*, 71-81.

Aliferis, C., Statnikov, A, & Samrdinos, L. (2006). Challenges in the analysis of mass-throughput data: A technical commentary from the statistical machine learning perpective. Retrieved October 19, 2009, from http://ncbi.nlm.nih.gov/pmc/articles/PMC2675497/pdf/cin-02-133.pdf

Alizadeh, A., Eisen, B., Davis, E., Ma, C., Lossos, I. S., Rosenwald, A., . . . Yu, X. (2000). Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. *Nature, 403*(6769), 503-511.

Alon, Uri, Barkai, Naama, Notterman, Daniel A, Gish, Kurt, Ybarra, Suzanne, Mack, Daniel, & Levine, Arnold J. (1999). Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proceedings of the National Academy of Sciences, 96*(12), 6745-6750.

Alzheimer's Association. (2012). 2013 Alzheimer's Disease facts and figures. *Alzheimer's and Dementia: The Journal of the Alzheimer's Association, 8*, 131-168.

Alzheimer's Australia. (2012). Statistics. Retrieved January 25, 2013, from http://www.fightdementia.org.au/about-us/statistics.aspx

Ambroise, C., & McLachlan, G.J. (2002). Selection bias in gene extraction on the basis of microarray gene-expression data. *Proceedings of the National Academy of Sciences, 99*(10), 6562-6566.

American Cancer Society. (2011). Global Cancer Facts & Figures 2nd Edition. Retrieved January 30, 2013, from http://www.cancer.org/acs/groups/content/@epidemiologysurveilance/documents/document/acspc-027766.pdf

Arai, Kohei, & Barakbah, Ali Ridho. (2007). Hierarchical K-means: an algorithm for centroids initialization for K-means. *Reports of the Faculty of Science and Engineering, 36*(1), 25-31.

Ayala, Helon Vicente Hultmann, & Coelho, Leandro dos Santos. (2008). A multiobjective genetic algorithm applied to multivariable control optimization. *ABCM Symposium Series in Mechatronics, 3*, 736-745.

Back, T., Hoffmeister, F., & Schwefel, H.P. (1991). *A survey of evolution strategies.* Paper presented at the Proc of the 4th Int. Genetic Algorithms Conference, CA: Morgan Kaufmann Publishers.

Bair, Eric, & Tibshirani, Robert. (2004). Semi-supervised methods to predict patient survival from gene expression data. *PLoS biology, 2*(4), e108.

Bala, Jerzy, Huang, Jeffrey, Vafaie, Haleh, DeJong, Kenneth, & Wechsler, Harry. (1995). *Hybrid learning using genetic algorithms and decision trees for pattern classification.* Paper presented at the International Joint Conference on Artificial Intelligence.

Bandyopadhyay, Sanghamitra, & Saha, Sriparna. (2013). Similarity Measures *Unsupervised Classification* (pp. 59-73): Springer.

Banerjee, M., Mitra, S., & Banka, H. (2007). Evolutionary Rough Feature Selection in Gene Expression Data. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, 37*(4), 622-632.

Batsch, Nicole, & Mittelman, Mary. (2012). World Alzheimer Report 2012, Overcoming the Stigma of demitia (pp. 5).

Beasley, David, Martin, RR, & Bull, DR. (1993). An overview of genetic algorithms: Part 1. Fundamentals. *University computing, 15*, 58-58.

Biological marker. (n.d.). *Collins English Dictionary – Complete and Unabridged. (1991, 1994, 1998, 2000, 2003*). Retrieved May 22, 2014, from http://www.thefreedictionary.com/biological+marker

Bradley, Paul S, & Fayyad, Usama M. (1998). *Refining Initial Points for K-Means Clustering*.

Cancer Research UK. (2012). World cancer factsheet. Retrieved January 30, 2013, from http://publications.cancerresearchuk.org/downloads/Product/CS_FS_WORLD_A4.pdf

Cao, L., Lee, H. P., Seng, C. K., & Gu, Q. (2003). Saliency Analysis of Support Vector Machines for Gene Selection in Tissue Classification. *Neural Computing &amp; Applications, 11*(3), 244-249. doi: 10.1007/s00521-003-0362-3

Chaiyaratana, Nachol, Piroonratana, Theera, & Sangkawelert, Nuntapon. (2007). Effects of diversity control in single-objective and multi-objective genetic algorithms. *Journal of Heuristics, 13*(1), 1-34.

Cho, Sunyoung, Kim, Boyeon, Park, Eunhea, Chang, Yunseok, Kim, Jongwoo, Chung, Kyungchun, . . . Kim, Hyuntaek. (2003). Automatic Recognition of Alzheimer's Disease Using Genetic Algorithms and Neural Network. *Computational Science—ICCS 2003*, 680-680.

Clarke, Robert, Ressom, Habtom, Wang, Antai, Xuan, Jianhua, Liu, Minetta, Gehan, Edmund, & Wang, Yue. (2008). The properties of high dimensional data spaces: implications for exploring gene and protein expression data. *Nature Review - Cancer, 8*(37), 12.

Clarke, William L, Anderson, Stacey, Farhy, Leon, Breton, Marc, Gonder-Frederick, Linda, Cox, Daniel, & Kovatchev, Boris. (2005). Evaluating the clinical accuracy of two continuous glucose sensors using Continuous glucose–error grid analysis. *Diabetes Care, 28*(10), 2412-2417.

Coello, C.A.C., & Lamont, G.B. (2004). *Applications of multi-objective evolutionary algorithms* (Vol. 1): World Scientific.

Dang, Vinh Quoc, & Lam, Chiou-Peng. (2013). *NSC-NSGA2: Optimal Search for Finding Multiple Thresholds for Nearest Shrunken Centroid*. Paper presented at the IEEE International Conference on Bioinformatics and Biomedicine (BIBM), Shanghai, China.

Dang, Vinh Quoc, Lam, Chiou-Peng, & Lee, Chang Su. (2011). *Incorporating genetic algorithm into rough feature selection for high dimensional biomedical data.* Paper presented at the IT in Medicine and Education (ITME), 2011 International Symposium on Vol. 2, pp. 283-287. IEEE.

Dang, Vinh Quoc, Lam, Chiou-Peng, & Lee, Chang Su. (2013). *NSC-GA: Search for optimal shrinkage thresholds for nearest shrunken centroid.* Paper presented at the Computational Intelligence in Bioinformatics and Computational Biology (CIBCB), 2013 IEEE Symposium on (pp. 44-51). IEEE.

Dash, M., & Liu, H. (1997). Feature selection for classification. *Intelligent Data Analysis, 1*, 131-156.

Dash, M., & Liu, H. (2003). Consistency-based search in feature selection. *Elsevier Artificial Intelligence*(151), 22. doi: 10.1016/S0004-3702(03)00079-1

Datta, Susmita, & Datta, Somnath. (2003). Comparisons and validation of statistical clustering techniques for microarray gene expression data. *Bioinformatics, 19*(4), 459-466. doi: 10.1093/bioinformatics/btg025

Dawkins, Richard. (2006). *The Selfish Gene: --with a new Introduction by the Author*: Oxford University Press, USA.

de Paula, Mateus Rocha, Ravetti, Martín Gómez, Berretta, Regina, & Moscato, Pablo. (2011). Differences in abundances of cell-signalling proteins in blood reveal novel biomarkers for early detection of clinical Alzheimer's disease. *PloS one, 6*(3), e17481.

Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation, 6*(2), 182 - 197.

Deb, K., & Reddy, A.R. (2003). Reliable classification of two-class cancer data using evolutionary algorithms. *BioSystems, 72*(1-2), 111-129.

DeJong, K. A. (1975). *Analysis of the behavior of a class of genetic adaptive systems.* (Doctor of Philosophy (Computer and communication sciences)), The University of Michigan, Michigan.

DeJong, Kenneth A, & Spears, William M. (1990). Learning concept classification rules using genetic algorithms: DTIC Document.

Deshpande, Raamesh, VanderSluis, Benjamin, & Myers, Chad L. (2013). Comparison of Profile Similarity Measures for Genetic Interaction Networks. *PloS one, 8*(7), e68664.

Diesinger, Isabel, Bauer, Christine, Brass, Nicole, Schaefers, Hans-Joachim, Comtesse, Nicole, Sybrecht, Gerhard, & Meese, Eckart. (2002). Toward a more complete recognition of immunoreactive antigens in squamous cell lung carcinoma. *International journal of cancer, 102*(4), 372-378.

Dieterle, Frank Jochen. (2003). Multianalyte quantifications by means of integration of artificial neural networks, genetic algorithms and chemometrics for time-resolved analytical data.

Ding, C., & Peng, H. (2005). Minimum redundancy feature selection from microarray gene expression data. *Journal of bioinformatics and computational biology, 3*(02), 185-205.

Dong, Juzhen, Zhong, Ning, & Ohsuga, Setsuo. (1999). Probabilistic rough induction: the GDT-RS methodology and algorithms *Foundations of Intelligent Systems* (pp. 621-629): Springer.

Duval, Béatrice, & Hao, Jin-Kao. (2010). Advances in metaheuristics for gene selection and classification of microarray data. *Briefings in bioinformatics, 11*(1), 127-141.

Eiben, A. E., & Smith, J. E. (2003). *Introduction to Evolutionary Computing*. Berlin: Springer.

Eiben, A. E., & Smith, J. E. (2007). *Introduction to evolutionary computing*. Berlin Heidelberg: Springer.

Elbeltagi, Emad, Hegazy, Tarek, & Grierson, Donald. (2005). Comparison among five evolutionary-based optimization algorithms. *Advanced Engineering Informatics, 19*(1), 43-53.

Emrich, Tobias, Jossé, Gregor, Kriegel, Hans-Peter, Mauder, Markus, Niedermayer, Johannes, Renz, Matthias, . . . Züfle, Andreas. (2013). Optimal Distance Bounds for the Mahalanobis Distance *Similarity Search and Applications* (pp. 175-181): Springer.

Fan, Jianqing, & Fan, Yingying. (2008). High dimensional classification using features annealed independence rules. *Annals of statistics, 36*(6), 2605.

Felix, Reynaldo, & Ushio, Toshimitsu. (1999). *Rough sets-based machine learning using a binary discernibility matrix.* Paper presented at the Intelligent Processing and Manufacturing of Materials, 1999. IPMM'99. Proceedings of the Second International Conference on.

Fonseca, Carlos, & Fleming, Peter. (1995). An overview of evolutionary algorithms in multiobjective optimization. *Evol. Comput., 3*(1), 1-16. doi: http://dx.doi.org/10.1162/evco.1995.3.1.1

Foss, A. (2010). *High-dimentional data mining: Subspace clustering, outlier detection and applications to classification.* (Doctor of Philosophy), University of Alberta, Columbia.

Franken, Holger, Lehmann, Rainer, Häring, Hans-Ulrich, Fritsche, Andreas, Stefan, Norbert, & Zell, Andreas. (2011). Wrapper-and ensemble-based feature subset selection methods for biomarker discovery in targeted metabolomics *Pattern Recognition in Bioinformatics* (pp. 121-132): Springer.

Fraser, Gordon, & Arcuri, Andrea. (2011). Evolutionary generation of whole test suites. *2011 11th International Conference on Quality Software (QSIC)*, 31-40.

Goldberg, D. E. (1989). *Genetic Algorithm in Search, Optimization, and Machine Learning*. Alabama: Addison-Wesley.

Goldberg, D. E., & Deb, Kalyanmoy. (1991). *A Comparative Analysis of Selection Schemes Used in Genetic Algorithms*. Paper presented at the Foundations of Genetic AlgorithmsSan Francisco, CA: Morgan Kaufmann (1991) , p. 69--93. .

Golub, Todd R, Slonim, Donna K, Tamayo, Pablo, Huard, Christine, Gaasenbeek, Michelle, Mesirov, Jill P, . . . Caligiuri, Mark A. (1999). Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science, 286*(5439), 531-537.

Gordon, Gavin J, Jensen, Roderick V, Hsiao, Li-Li, Gullans, Steven R, Blumenstock, Joshua E, Ramaswamy, Sridhar, . . . Bueno, Raphael. (2002). Translation of microarray data into clinically relevant cancer diagnostic tests using gene expression ratios in lung cancer and mesothelioma. *Cancer Research, 62*(17), 4963-4967.

Grefenstette, John J. (1986). Optimization of control parameters for genetic algorithms. *Systems, Man and Cybernetics, IEEE Transactions on, 16*(1), 122-128.

GSAEmulator. (n.d.). Retrieved 29 March, 2013, from http://www.clopinet.com/isabelle/Projects/RFE/gene.html

Guyon, Isabelle. (2007). Lecture 2: Introduction to feature selection. Retrieved March 10, 2010, from http://velblod.videolectures.net/2007/pascal/bootcamp07_vilanova/guyon_isabelle/L2_featselect1.pdf

Guyon, Isabelle, & Elisseeff, André. (2006). An introduction to feature extraction. *Feature Extraction*, 1-25.

Hall, Mark A. (1999). *Correlation-based feature selection for machine learning.* (Doctoral Dissertation), The University of Waikato.

Hall, Mark A. (1999). *Correlation-based feature selection for machine learning.* (Doctor of Philosophy), The University of Waikato, Hamilton. Retrieved from http://www.cs.waikato.ac.nz/~mhall/thesis.pdf

Hall, Mark, Frank, Eibe, Holmes, Geoffrey, Pfahringer, Bernhard, Reutemann, Peter, & Witten, Ian H. (2009). The WEKA data mining software: an update. *SIGKDD Explor. Newsl., 11*(1), 10-18. doi: 10.1145/1656274.1656278

Han, Jiawei, & Kamber, Micheline. (2006). *Data mining concepts and techniques* (2 ed.). Delhi: India: Morgan Kaufmann.

Han, Jiawei, Kamber, Micheline, & Pei, Jian. (2006). *Data mining: concepts and techniques*: Morgan kaufmann.

Hanczar, B., Courtine, M., Benis, A., Hennegar, C., Clément, K., & Zucker, J.D. (2003). Improving classification of microarray data using prototype-based feature selection. *ACM SIGKDD Explorations Newsletter, 5*(2), 23-30.

Handels, H., Rob, Th., Kreusch, J., Wolf, H. H., & Poppl, S. J. (1998). Feature selection for optimized skin tumor recognition using genetic algorithms. *elsevier artificial inteligence in medicine, 16*(1999), 297. doi: 0933.3657/99/S

Harik, Georges R, & Lobo, Fernando G. (1999). *A parameter-less genetic algorithm.* Paper presented at the Proceedings of the genetic and evolutionary computation conference.

Hasegawa, T, Isobe, K, Tsuchiya, Y, Oikawa, S, Nakazato, H, Nakashima, I, & Shimokata, K. (1993). Nonspecific crossreacting antigen (NCA) is a major member of the carcinoembryonic antigen (CEA)-related gene family expressed in lung cancer. *British journal of cancer, 67*(1), 58.

Hedvat, Cyrus V, Comenzo, Raymond L, Teruya-Feldstein, Julie, Olshen, Adam B, Ely, Scott A, Osman, Keren, . . . Nimer, Stephen D. (2003). Insights into extramedullary tumour cell growth revealed by expression profiling of human plasmacytomas and multiple myeloma. *British journal of haematology, 122*(5), 728-744.

Hinoda, Yuji, Saito, Tehzoh, Takahashi, Hiroki, Itoh, Fumio, Adachi, Masaaki, & Imai, Kohzoh. (1997). Induction of nonspecific cross-reacting antigen mRNA by interferon-γ and anti-fibronectin receptor antibody in colon cancer cells. *Journal of gastroenterology, 32*(2), 200-205.

Hoa, Nguyen S, & Son, Nguyen Hung. (1996). *Some efficient algorithms for rough set methods.* Paper presented at the Proceedings of the sixth International Conference on Information Processing Management of Uncertainty in Knowledge Based Systems.

Hoefsloot, H. (2013). Bioinformatics and Statistics: statistical analysis and validation. Amsterdam: Royal Society of Chemistry

Hooper, C., Lovestone, S., & Sainz-Fuertes, R. (2008). Alzheimer's Disease, Diagnosis and the Need for Biomarkers. *Biomarker Insights, 3*, 317.

Hu, Qinghua, Yu, Daren, Liu, Jinfu, & Wu, Congxin. (2008). Neighborhood rough set based heterogeneous feature subset selection. *Information sciences, 178*(18), 3577-3594.

Huang, Zhexue. (1998). Extensions to the k-means algorithm for clustering large data sets with categorical values. *Data Mining and Knowledge Discovery, 2*(3), 283-304.

Huerta, Edmundo Bonilla, Duval, Béatrice, & Hao, Jin-Kao. (2006). A hybrid GA/SVM approach for gene selection and classification of microarray data *Applications of Evolutionary Computing* (pp. 34-44): Springer.

Ilonen, Jarmo, Kamarainen, Joni-Kristian, & Lampinen, Jouni. (2003). Differential evolution training algorithm for feed-forward neural networks. *Neural Processing Letters, 17*(1), 93-105.

Inza, Inaki, Larranaga, Pedro, Blanco, Rosa, & Cerrolaza, Antonio J. (2004). Filter versus wrapper gene selection approaches in DNA microarray domains. *Elsevier Intelligence in medicine, 31*, 103. doi: 10.1016/j.artmed.2004.01.007

Jaaman, Saiful Hafizah, Shamsuddin, Siti Mariyam, Yusob, Bariah, & Ismail, Munira. (2009). A predictive model construction applying rough set methodology for Malaysian stock market returns. *International Research Journal of Finance and Economics*(30), 218.

Jirapech-Umpai, Thanyaluk, & Aitken, Stuart. (2004). Feature selection and classification for microarray data analysis: Evolutionary methods for identifying predictive genes. *BMC Bioinformatics, 2005*(6), 148. doi: 10.1186/1471-2105-6-148

Jourdan, Laetitia, Dhaenens, Clarisse, & Talbi, El-Ghazali. (2001). A genetic algorithm for feature selection in data-mining for genetics. *Proceedings of the 4th Metaheuristics International ConferencePorto (MIC'2001)*, 29-34.

Kannan, S Senthamarai, & Ramaraj, N. (2010). A novel hybrid feature selection via Symmetrical Uncertainty ranking based local memetic search algorithm. *Knowledge-Based Systems, 23*, 580-585.

Kim, Gilhan, Kim, Yeonjoo, Lim, Heuiseok, & Kim, Hyeoncheol. (2010). An MLP-based feature subset selection for HIV-1 protease cleavage site analysis. *Artificial intelligence in medicine, 48*(2), 83-89.

Klassen, M., & Kim, N. (2009). *Nearest shrunken centroid as feature selection for microarray data.* Paper presented at the ICATA (Computers and Their Applications).

Kohavi, Ron, & John, George H. (1997). Wrappers for feature subset selection. *Artificial intelligence, 97*(1), 273-324.

Koumousis, Vlasis K, & Katsaras, Christos P. (2006). A saw-tooth genetic algorithm combining the effects of variable population size and reinitialization to enhance performance. *Evolutionary Computation, IEEE Transactions on, 10*(1), 19-28.

Krasnogor, Natalio, & Smith, Jim. (2005). A tutorial for competent memetic algorithms: model, taxonomy, and design issues. *Evolutionary Computation, IEEE Transactions on, 9*(5), 474-488.

Krizek, Pavel. (2008). *Feature selection: stability, algorithms, and evaluation.* (PhD Doctoral thesis), Czech Technical University, Prague. Retrieved from ftp://cmp.felk.cvut.cz/pub/cmp/articles/krizek/Krizek-TR-2008-16.pdf

Kumar, Rajeev, & Rockett, Peter. (2002). Improved sampling of the Pareto-front in multiobjective genetic optimizations by steady-state evolution: a Pareto converging genetic algorithm. *Evolutionary Computation, 10*(3), 283-314.

Lancashire, Lee, Rees, Robert, & Ball, Graham. (2008). Identification of gene transcript signatures predictive for estrogen receptor and lymph node status using a stepwise forward selection artificial neural network modelling approach. *Elsevier Artificial intelligence in medicine*(43), 99-111. doi: 10.1016/j.artmed.2008.03.001

Laping, Nicholas J. (1999). DNA encoding human MAD proteins: Google Patents.

Leale, Guillermo, Milone, Diego H, Bayá, Ariel, Granitto, Pablo M, Stegmayer, Georgina, CIDISI, UTN-FRSF, & CIFASIS, UPCAM France. (2013). *A novel* *clustering approach for biological data using a new distance based on Gene Ontology*. Paper presented at the 14th Argentine Symposium on Articial Intelligence, ASAI.

Lee, JW, Lee, JB, Park, M, & Song, SH. (2005). An extensive evaluation of recent classification tools applied to microarray data. *Computation Statistics and Data Analysis, 48*, 869 - 885.

Levner, Ilya. (2005). Feature selection and nearest centroid classification for protein mass spectrometry. *BMC Bioinformatics, 6*(1), 68.

Li, Heng, & Love, Peter. (1997). Using improved genetic algorithms to facilitate time-cost optimization. *Journal of Construction Engineering and management, 123*(3), 233-237.

Li, Jinyan, & Liu, Huiqing. (2002). Kent Ridge Bio-medical Data Set Repository, 2002.

Li, Leping, Weinberg, Clarice R., Darden, Thomas A., & Pedersen, Lee G. (2001). Gene selection for sample classification based on gene expression data: study of sensitivity to choice of parameters of the GA/KNN method. *Bioinformatics, 17*(12), 1131-1142. doi: 10.1093/bioinformatics/17.12.1131

Li, Li, Jiang, Wei, Li, Xia, Moser, Kathy L, Guo, Zheng, Du, Lei, . . . Rao, Shaoqi. (2005). A robust hybrid between genetic algorithm and support vector machine for extracting an optimal feature gene subset. *Genomics, 85*(1), 16-23.

Li, Yifeng, Liu, Yihui, & Bai, Li. (2008). *Genetic algorithm based feature selection for mass spectrometry data.* Paper presented at the BioInformatics and BioEngineering, 2008. BIBE 2008. 8th IEEE International Conference on.

Liu, Juan, & Iba, Hitoshi. (2002). *Selecting informative genes using a multiobjective evolutionary algorithm.* Paper presented at the Evolutionary Computation, 2002. CEC'02. Proceedings of the 2002 Congress on.

Lu, Yijuan, Tian, Qi, Neary, Jennifer, Liu, Feng, & Wang, Yufeng. (2008). Adaptive discriminant analysis for microarray-based classification. *ACM Transactions on Knowledge Discovery from Data (TKDD), 2*(1), 5.

Ma, Shuangge, & Huang, Jian. (2008). Penalised feature selection and classification in bioinformatics. *Briefings in bioinformatics - Oxford*, 12. doi: 10.1093/bib/bbn027

MacQueen, J. (1967). *Some Methods for classification and Analysis of Multivariate Observations*. Paper presented at the Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability, Berkeley.

Mahalanobis, Prasanta Chandra. (1936). *On the generalized distance in statistics.* Paper presented at the Proceedings of the National Institute of Sciences of India.

Markov, Zdravko, & Russell, Ingrid. (2006). *An introduction to the WEKA data mining system.* Paper presented at the ACM SIGCSE Bulletin.

MayoClinic. (2013). Alzheimer's Disease: Tests and diagnosis. from http://www.mayoclinic.com/health/alzheimers-disease/DS00161/DSECTION=tests-and-diagnosis

McLachlan, GJ. (1999). Mahalanobis distance. *Resonance, 4*(6), 20-26.

Merz, Peter, & Freisleben, Bernhard. (1999). Fitness landscapes and memetic algorithm design. *New ideas in optimization*, 245-260.

Midelfart, Herman, Komorowski, Jan, Nørsett, Kristin, Yadetie, Fekadu, Sandvik, Arne K, & Lægreid, Astrid. (2002). Learning Rough Set Classifiers from Gene Expressions and Clinical Data. *Fundamenta Informaticae, 53*, 155-183.

Miller, Brad L, & Goldberg, David E. (1995). Genetic Algorithms, Tournament Selection, and the Effects of Noise. *Urbana, 51*, 61801.

Milton, John. (2009). *Analysis and improvement of genetic algorithms using concepts from information theory.* University of Technology Sydney.

Mitra, S., & Hayashi, Y. (2006). Bioinformatics with soft computing. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, 36*(5), 616-635.

Mitra, Sushmita, & Banka, Haider. (2006). Multi-objective evolutionary biclustering of gene expression data. *Pattern Recognition, 39*(12), 2464-2477.

Mukherjee, S., Tamayo, P., Slonim, D., Verri, A., Golub, T., Mesirov, JP, & Poggio, T. (1998). *Support Vector Machine Classification of Microarray Data.* Paper presented at the AI Memo 1677, Massachusetts Institute of Technology.

Nannen, V., Smit, S. K., & Eiben, A. E. (2008). Costs and benefits of tuning parameters of evolutionary algorithms. *Proc. 10th International Conference on Parallel Problem Solving from Nature, PPSN X, Dortmund*, 528–538.

Nazeer, KA Abdul, & Sebastian, MP. (2009). *Improving the Accuracy and Efficiency of the k-means Clustering Algorithm.* Paper presented at the Proceedings of the World Congress on Engineering.

Nishioka, Kenya, Vilariño-Güell, Carles, Cobb, Stephanie A, Kachergus, Jennifer M, Ross, Owen A, Hentati, Emna, . . . Farrer, Matthew J. (2010). Genetic variation of the mitochondrial complex I subunit< i> NDUFV2</i> and Parkinson's disease. *Parkinsonism & related disorders, 16*(10), 686-687.

Ong, Bun Theang, & Fukushima, Masao. (2011). Genetic algorithm with automatic termination and search space rotation. *Memetic Computing, 3*(2), 111-127.

Paul, Sushmita, & Maji, Pradipta. (2014). A New Rough-Fuzzy Clustering Algorithm and its Applications. In B. V. Babu, A. Nagar, K. Deep, M. Pant, J. C. Bansal, K. Ray & U. Gupta (Eds.), *Proceedings of the Second International Conference on Soft Computing for Problem Solving (SocProS 2012), December 28-30, 2012* (Vol. 236, pp. 1245-1251): Springer India.

Pawlak, Zdzisław. (1982). Rough sets. *International Journal of Computer & Information Sciences, 11*(5), 341-356.

Pawlak, Zdzisław. (1997). Rough set approach to knowledge-based decision support. *European journal of operational research, 99*(1), 48-57.

Pearson, Karl. (1895). Note on regression and inheritance in the case of two parents. *Proceedings of the Royal Society of London, 58*(347-352), 240-242.

Peng, Sihua, Xu, Qianghua, Ling, ZXuefeng, Peng, Xiaoning, & Du, Wei. (2003). Molecular classification of cancer types from microarray data using the combination of genetic algorithms and support vector machines. *Elsevier - Federation of European Biochemical Societies, 555*, 362.

Perez, Meir, & Marwala, Tshilidzi. (2012). *Microarray data feature selection using hybrid genetic algorithm simulated annealing.* Paper presented at the Electrical & Electronics Engineers in Israel (IEEEI), 2012 IEEE 27th Convention of.

Petricoin, EF, Ardekani, AM, Hitt, BA, Levine, PJ, Fusaro, VA, Steinberg, SM, . . . Liotta, LA. (2002). Use of proteomic patterns in serum to identify ovarian cancer. *The Lancet, 359*(9306), 572 - 577.

Portinale, L, & Saitta, L. (2002). Feature Selection: State of the Art. *Portinale, L., Saitta, L. Feature Selection*, 1-22.

Pujari, Arun K. (2001). *Data mining techniques*. Hyderabad: Universities press (India).

Punitha, A., & Santhanam, T. (2008). Correlated rough set based classificatory decomposition for breast cancer diagnosis using fuzzy art neural network. *Asian Journal of Information Technology, 1*(1), 6.

Qin, Yan. (1999). *Comparison Of Different Chromosome Representations Used By Genetic Algorithms For Scheduling Engineering Missions.* Citeseer.

Ravetti, M., & Moscato, P. (2008). Identification of a 5-Protein Biomarker Molecular Signature for Predicting Alzheimer's Disease. *Plos One, 3*(8), 12.

Ray, S., Britschgi, M., Herbert, C., Takeda-Uchimura, Y., Boxer, A., Blennow, K., . . . Karydas, A. (2007). Classification and prediction of clinical Alzheimer's diagnosis based on plasma signaling proteins. *Nature medicine, 13*(11), 1359-1362.

Ray, S., & Wyss-coray, A. (2010). COLLECTION OF BIOMARKERS FOR DIAGNOSIS AND MONITORING OF ALZHEIMER'S DISEASE IN BODY FLUIDS: US Patent 20,100,124,756.

Rocha de Paula, Mateus, Gómez Ravetti, Martín, Berretta, Regina, & Moscato, Pablo. (2011). Differences in Abundances of Cell-Signalling Proteins in Blood Reveal Novel Biomarkers for Early Detection Of Clinical Alzheimer's Disease. *PLoS ONE, 6*(3), e17481. doi: 10.1371/journal.pone.0017481

Saeys, Yvan, Inza, Inaki, & Larranaga, Pedro. (2007). A review of feature selection techniques in bioinformatics. *23*(19), 2507-2517. doi: 10.1093/bioinformatics/btm344

Safe, Martín, Carballido, Jessica, Ponzoni, Ignacio, & Brignole, Nélida. (2004). On stopping criteria for genetic algorithms *Advances in Artificial Intelligence–SBIA 2004* (pp. 405-413): Springer.

Schaffer, J. David. (1985). *Multiple Objective Optimization with Vector Evaluated Genetic Algorithms*. Paper presented at the Proceedings of the 1st International Conference on Genetic Algorithms.

Shaik, Jahangheer S, & Yeasin, Mohammed. (2007). A unified framework for finding differentially expressed genes from microarray experiments. *BMC Bioinformatics, 8*(1), 347.

Sharpe, Peter K, & Glover, Robin P. (1999). Efficient GA based techniques for classification. *Applied Intelligence, 11*(3), 277-284.

Siegel, Rebecca, Naishadham, Deepa, & Jemal, Ahmedin. (2013). Cancer statistics, 2013. *CA: A Cancer Journal for Clinicians, 63*(1), 11-30. doi: 10.3322/caac.21166

Singh, Dinesh, Febbo, Phillip G, Ross, Kenneth, Jackson, Donald G, Manola, Judith, Ladd, Christine, . . . Richie, Jerome P. (2002). Gene expression correlates of clinical prostate cancer behavior. *Cancer Cell, 1*(2), 203-209.

Ślezak, Dominik, & Wróblewski, Jakub. (2007). Roughfication of numeric decision tables: The case study of gene expression data. *Rough Sets and Knowledge Technology*, 316-323.

Snyder, Lawrence V, & Daskin, Mark S. (2006). A random-key genetic algorithm for the generalized traveling salesman problem. *European Journal of Operational Research, 174*(1), 38-53.

Somorjai, R.L., Dolenko, B., & Baumgartner, R. (2003). Class prediction and discovery using gene microarray and proteomics mass spectroscopy data: curses, caveats, cautions. *Bioinformatics, 19*(12), 1484-1491. doi: 10.1093/bioinformatics/btg182

Srinivas, M., & Patnaik, L. M. (1994). Adaptive Probabilities of Crossover Genetic in Mu tation and Algorithms. *IEEE, 24*(4), 656-667.

Srinivas, N., & Kalyanmoy, Deb. (1994). Muiltiobjective optimization using nondominated sorting in genetic algorithms. *Evol. Comput., 2*(3), 221-248.

Stein, Gary, Chen, Bing, Wu, Annie S, & Hua, Kien A. (2005). *Decision tree classifier for network intrusion detection with GA-based feature selection.* Paper presented at the Proceedings of the 43rd annual Southeast regional conference-Volume 2.

Stoeckel, Jonathan, & Fung, Glenn. (2007). SVM feature selection for classification of SPECT images of Alzheimer's disease using spatial information. *Knowledge and Information Systems,, 11*(2), 243-258. doi: 1550-4786/05

Sun, Y., Babbs, C. F., & Delp, E. J. (2005). *A Comparison of Feature Selection Methods for the Detection of Breast Cancers in Mammograms: Adaptive Sequential Floating Search vs. Genetic Algorithm.* Paper presented at the Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005. 27th Annual International Conference.

Suzuki, HIROYUKI, Takahashi, KAZUHIRO, & Shibahara, SHIGEKI. (1995). Evidence for the presence of two amino-terminal isoforms of neurofibromin, a gene product responsible for neurofibromatosis type 1. *The Tohoku journal of experimental medicine, 175*(4), 225-233.

Swiniarski, Roman W. (2001). Rough sets methods in feature selection and classification. *Int. J. Appl. Math. Comput. Sci, 11*(3), 582.

Tai, F., & Pan, W. (2007). Incorporating prior knowledge of gene functional groups into regularized discriminant analysis of microarray data. *Bioinformatics, 23*(23), 3170-3177.

Tan, P., Steinbach, M., & Kumar, V. (2006). *Introduction to data mining* (China ed.): Pearson Education Asia Ltd and Post & Telecom Press.

The McCusker foundation for Alzheimer's disease research. (2010a). Diagnosis of Alzheimer's disease. Retrieved June 2, 2010, from http://www.alzheimers.com.au/alzheimers/diagnosis.php

The McCusker foundation for Alzheimer's disease research. (2010b). Gene Hunting. Retrieved March 16, 2010, from http://www.alzheimers.com.au/research/genes.php

Tibshirani, R., Hastie, T., Narasimhan, B., & Chu, G. (2002). Diagnosis of multiple cancer types by shrunken centroids of gene expression. *Proc Natl Acad Sci USA, 99*(10), 6567 - 6572.

Tibshirani, R., Hastie, T., Narasimhan, B., & Chu, G. (2003). Class prediction by nearest shrunken centroids, with applications to dna microarrays. *Statistical Science, 18*(1), 104 - 117.

Trojanowskl, J. Q. . (2004). Biomarkers of Alzheimer’s. Retrieved May 22, 2009, from http://www.loni.ucla.edu/ADNI/About/BioMarker.pdf

Tsukasaki, Kunihiro, Tanosaki, Sakae, DeVos, Sven, Hofmann, Wolf K, Wachsman, William, Gombart, Adrian F, . . . Nagai, Kazuhiro. (2004). Identifying progression-associated genes in adult T-cell leukemia/lymphoma by using oligonucleotide microarrays. *International journal of cancer, 109*(6), 875-881.

Vafaie, Haleh, & De Jong, Kenneth. (1992). *Genetic algorithms as a tool for feature selection in machine learning.* Paper presented at the Tools with Artificial Intelligence, 1992. TAI'92, Proceedings., Fourth International Conference on.

Visalakshi, N Karthikeyani, & Thangavel, K. (2009). Impact of normalization in distributed k-means clustering. *international Journal of Soft computing, 4*(4), 168-172.

Wang, Dong, Wang, Juan, Lu, Ming, Song, Fei, & Cui, Qinghua. (2010). Inferring the human microRNA functional similarity and functional network based on microRNA-associated diseases. *Bioinformatics, 26*(13), 1644-1650.

Wang, Hongfeng, Wang, Dingwei, & Yang, Shengxiang. (2009). A memetic algorithm with adaptive hill climbing strategy for dynamic optimization problems. *Soft Computing-A Fusion of Foundations, Methodologies and Applications, 13*(8), 763-780.

Wang, S., & Zhu, J. (2007). Improved centroids estimation for the nearest shrunken centroid classifier. *Bioinformatics, 23*(8), 972-979.

West, Mike, Nevins, Joseph R, Goldschmidt, Pascal, & Seo, David. (2005). Atherosclerotic phenotype determinative genes and methods for using the same: Google Patents.

Williams, Amy J, Khachigian, Levon M, Shows, Thomas, & Collins, Tucker. (1995). Isolation and characterization of a novel zinc-finger protein with transcriptional repressor activity. *Journal of Biological Chemistry, 270*(38), 22143-22152.

Witten, I. H., & Frank, E. (2005). *Data mining practical machine learning tools and techniques (2nd ed.)*: Amsterdam: Morgan Kaufmann Publishers.

Wölfel, Matthias, & Ekenel, Hazim Kemal. (2005). *Feature weighted Mahalanobis distance: Improved robustness for Gaussian classifiers.* Paper presented at the 13th European Signal Processing Conference.

Wood, Ian A, Visscher, Peter M, & Mengersen, Kerrie L. (2007). Classification based upon gene expression data: bias and precision of error rates. *Bioinformatics, 23*(11), 1363-1370.

Wright, Alden H. (1990). *Genetic Algorithms for Real Parameter Qptimization.* Paper presented at the FOGA.

Wroblewski, Jakub. (1995). *Finding minimal reducts using genetic algorithms.* Paper presented at the Proceedings of Second International Joint Conference on Information Science.

Wu, Fengjie. (2001a). *A Framework for Memetic Algorithms.* (Master of Science in Computer Science), University of Auckland, Auckland.

Wu, Fengjie. (2001b). *A Framework for Memetic Algorithms.* (Doctoral Dissertation), University of Auckland.

Yang, Jihoon, & Honavar, Vasant. (1998). Feature subset selection using a genetic algorithm *Feature extraction, construction and selection* (pp. 117-136): Springer.

Yedla, Madhu, Pathakota, Srinivasa Rao, & Srinivasa, TM. (2010). Enhancing K-means clustering algorithm with improved initial center. *International Journal of computer science and information technologies, 1*(2), 121-125.

Yeung, Ka Yee, & Bumgarner, Roger E. (2003). Multiclass classification of microarray data with repeated measurements: application to cancer. *Genome Biology, 4*(12), 19.

Yeung, KY, & Bumgarner, RE. (2003). Multiclass classification of microarray data with repeated measurements: application to cancer. *Genome Biol, 4*, R83-R83.

Yeung, KY, Bumgarner, RE, & Raftery, AE. (2005). Bayesian model averaging: development of an improved multi-class, gene selection and classification tool for microarray data. *Bioinformatics, 21*, 2394 - 2402.

Yona, Golan, Dirks, William, Rahman, Shafquat, & Lin, David M. (2006). Effective similarity measures for expression profiles. *Bioinformatics, 22*(13), 1616-1622.

Yu, L., & Liu, H. (2004). Efficient feature selection via analysis of relevance and redundancy. *The Journal of Machine Learning Research, 5*, 1205-1224.

Zhang, Hongbin, & Sun, Guangyu. (2002). Feature selection using tabu search method. *Pattern Recognition, 35*(3), 701-711.

Zhong, Ning, Dong, Ju-Zhen, & Ohsuga, Setsuo. (1998). Data mining: A probabilistic rough set approach *Rough Sets in Knowledge Discovery 2* (pp. 127-144): Springer.

Zhong, Ning, Dong, Juzhen, & Ohsuga, Setsuo. (2001). Using rough sets with heuristics for feature selection. *Journal of Intelligent Information Systems, 16*(3), 199-214.

Zhu, Z., Ong, Y., & Dash, M. (2007). Wrapper–filter feature selection algorithm using a memetic framework. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, 37*(1), 70-76.