Fall 12-17-2014

Data Mining Analysis of the Parkinson's Disease

Xiaoyuan Wang

DATA MINING ANALYSIS OF THE PARKINSON'S DISEASE

by

XIAOYUAN WANG

Under the Direction of Yichuan Zhao PhD

ABSTRACT

Biological research is becoming increasingly database driven and statistical learning can be used to discover patterns in the biological data. In the thesis, the supervised learning approaches are utilized to analyze the Oxford Parkinson's disease detection data and build models for prediction or classification. We construct predictive models based on training set, evaluate their performance by applying these models to an independent test set, and find the best methods for predicting whether people have Parkinson's disease. The proposed artificial neural network procedure outperforms with the best and highest prediction accuracy, while the logistic and probit regressions are preferred statistical models which can offer better interpretation with the higher prediction accuracy compared to other proposed data mining approaches.

INDEX WORDS: Supervised Learning, Cross Validation, Prediction Analysis, Classification, Model Selection, ROC Curve, AUC.

DATA MINING ANALYSIS OF THE PARKINSON'S DISEASE

by

XIAOYUAN WANG

A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of

Master of Science

in the College of Arts and Sciences

Georgia State University

2014

DATA MINING ANALYSIS OF THE PARKINSON'S DISEASE

by

XIAOYUAN WANG

Committee Chair: Yichuan Zhao

Committee: Ruiyan Luo

Xin Qi

Electronic Version Approved:

Office of Graduate Studies

College of Arts and Sciences

Georgia State University

December 2014

ACKNOWLEDGEMENTS

I would like to express my appreciation to my advisor Professor Yichuan Zhao for his exceptional guidance and support throughout my studies and experience at the Georgia State Univeristy. He has taught me innumerable lessons in my scholarship and research. I offer immense gratitude to my committee consisting of Professor Ruiyan Luo and Professor Xin Qi. I am grateful to my family for a lifetime of support and understanding. Finally, I thank my colleagues at the Department of Mathematics and Statistics for their friendship and motivation.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

1.1 Chapter Overview

This chapter will describe the Parkinson's disease (PD) and the PD dataset to be used in this thesis. There are particular needs and opportunities where data mining techniques can be used to discover knowledge from a biological data set and predict diseases. This chapter is organized as follows. (i) Section 1.2 discusses the Parkinson's disease and (ii) Section 1.3 describes the PD dataset created by Max Little of the University of Oxford.

1.2 Parkinson's Disease

Parkinson's disease (PD) is a chronic progressive neurodegenerative disorder disease which includes symptoms such as tremors, rigidity, bradykinesia, and flat facial expression symptoms (Marsden, 1994). Additional experimental and clinical observations prove that PD patients are good candidates for contracting a dynamical disease (Beuter and Vasilakos, 1995). PD also impairs patients' other functions such as walking, mood, behavior, thinking, and sensation. Symptoms of PD include reduced loudness, breathiness, roughness, decreased energy in the higher parts of the harmonic spectrum and exaggerated vocal tremor (Beuter and Vasilakos, 1995). Nowadays, in North America over one million people have been impaired by PD, and most of them are over the age of 50. Data shows that the probability of a person developing PD dramatically increases after age of 60. Although medication may relieve symptoms, PD patients are not able to fully recover (Beuter and Vasilakos, 1995).

As a type of neurological disease, the Parkinson's disease (PD) may affect phonation of the patients. Approximately 90% of the patients display types of vocal impairments including vocal sound (dysphonia) and speech (dysarthria) (Ho et al., 2008). Considering that clinical intervention is difficult to provide for elderly patients, telemonitering systems are useful for detecting PD by analyzing vocal signals. Thus, this thesis utilizes the dataset for PD

prediction that focuses on speech signals.

1.3 PD Dataset

The dataset to be used was created by Max Little of the University of Oxford, in collaboration with the National Center for Voice and Speech, Denver, Colorado, who recorded the speech signals (Little et al., 2007). The Oxford PD dataset has 195 voices recordings from 31 people, and each of which has 22 features (called "predictors" or "explanatory variables" throughout this study). Among these 31 individuals, 23 of them are healthy. As a range of biomedical voice measurements, these 22 features include the mean of vocal fundamental frequency, the maximum, minimum, variation of fundamental frequency, variation in amplitude, ratio of noise to tonal components on the voice, nonlinear dynamical complexity, signal fractal scaling exponents, and nonlinear measures of fundamental frequency variations. All features describing characteristics of the speech presented in the records are computed from voice and speech signals. The detailed characteristics of each feature are shown in Table 1.1. Furthermore, each observation is labeled by a response variable indicating whether the people have PD or not. The response variable is called "status" in the PD dataset.

Mining bioinformatics is an emerging area at the intersection between bioinformatics and data mining. Mining bioinformatics presents benefits to the data mining, bioinformatics, and medicine communities. This thesis demonstrates several data mining techniques for classifying individuals with PD (status=1) or without PD (status=0) and differentiating healthy people from those with PD by finding the patterns within the dataset and constructing predictive models.

Table 1.1 Characteristic Features of PD Dataset (Adapted From Ramani and Sivagami, 2011)

Feature Number	Feature Name	Description
F1	MDVP: Fo (Hz)	Average vocal fundamental
F2	MDVP: Fhi (Hz)	Maximum vocal fundamental frequency
F3	MDVP: Flo (Hz)	Minimum vocal fundamental frequency
F4	MDVP: Jitter (%)	Kay Pentax MDVP jitter as percentage
F5	MDVP: Jitter (Abs)	Kay Pentax MDVP absolute jitter in microseconds
F6	MDVP: RAP	Kay Pentax MDVP relative amplitude perturbation
F7	MDVP: PPQ	Kay Pentax MDVP five-point period perturbation quotient
F8	Jitter: DDP	Average absolute difference of differences between cycles, divided by the average period
F9	MDVP: Shimmer	Key Pentax MDVP local shimmer
F10	MDVP: Shimmer (dB)	Key Pentax MDVP local shimmer in decibels
F11	Shimmer: APQ3	3 point amplitude perturbation quotient
F12	Shimmer: APQ5	5 point amplitude perturbation quotient
F13	MDVP: APQ	Kay Pentax MDVP eleven-point amplitude perturbation quotient
F14	Shimmer: DDA	Average absolute difference between consecutive differences between the amplitude of consecutive periods
F15	NHR	Noise to harmonic ratio
F16	HNR	Harmonics to noise ratio
F17	RPDE	Recurrence period density entropy
F18	DFA	Detrended fluctuation analysis
F19	spread1	Nonlinear measure of fundamental frequency
F20	spread2	Nonlinear measure of fundamental frequency
F21	D2	Pitch period entropy
F22	PPE	Pitch

2 METHODOLOGY OVERVIEW

2.1 Chapter Overview

Most of the statistical learning problems can be solved by supervised methods. This chapter introduces the concept of supervised learning to be demonstrated on the PD dataset in the next few chapters. This chapter is organized as follows. (i) Section 2.2 provides the introduction of supervised learning and (ii) Section 2.3 provides the purpose of current study.

2.2 Supervised learning

In the domain of the supervised learning, for each observation of the predictor measurement(s) $x_i, i = 1, \ldots, n$, there is an associated response measurement y_i. The aim of supervised learning is to specify a relationship between the predictors and response (which is also called the "dependent" variable throughout this thesis). Many classical statistical learning methods, such as linear regression, decision trees, logistic regression, and support vector machine are in the domain of supervised learning. Typically, to apply supervised data mining techniques, the values of the dependent variable must be known for an adequately large part of the dataset, and the goal of the supervised learning algorithm is to minimize the prediction error with respect to given inputs. Many supervised learning data mining projects use a large amount of observations. Usually these observations are separated into the training set and test set. The date set used to construct the data mining model is called the training dataset, which contains most of the data points. However, most algorithms work very hard on the training dataset and tend to over-fit the data. Consequentially, the discovered relationship between response and predictors in the training set might not hold in general. As the test set is independent from the training set, and it can be used to evaluate the performance of a model by making a prediction against the test set using a model that has been developed based on the training set. Usually, the test set is used to obtain the performance characteristics, such as

accuracy rate, specificity, sensitivity, false positive rate, false negative rate, and so on. These criteria can provide a reasonable estimate for developing a model of the completely unseen data.

2.3 Purpose of the Study

The current effort applies numerous supervised learning algorithms, such as generalized linear model (logistic and probit regression), linear and quadratic discriminant analysis, decision trees, random forest, support vector machine, artificial neural network, etc., to analyze the PD dataset. The objective of this thesis is to perform supervised learning algorithms on the PD dataset and evaluate their performance in discriminating healthy people from PD patients. The following chapters are dedicated to discuss and apply various statistical learning algorithms to predict potential PD patients.

3 DATA EXPLORATION

3.1 Chapter Overview

This chapter provides the explanatory data analysis of the PD data and explains the training and test sets to be used for model construction and evaluation. This chapter is organized as follows. (i) Section 3.2 provides the exploratory data analysis and (ii) Section 3.3 shows the training set and test set of the PD dataset.

3.2 Exploratory Data Analysis

The measurements in the PD dataset have been generally introduced in Section 1.3. In order to better elucidate, this Section shows the quantitative and exploratory data analyses of the dataset.

Table 3.1 describes basic statistics of all observations by each feature. It is obvious that the magnitude of these variables varies a lot. Thus, all feature variables are normalized for the further statistical analysis and learning. Based on the normalized dataset, Figure 3.1 shows the scatter plot matrix of partial features in the PD dataset and Figure 3.2 displays the boxplot of partial features (F1, F2, F4, F5, F7, F8, F9, F11, F14, F17, F20, and F22) by two groups with “status” = 1 and “status” = 0. It is noted that the distribution of some features are significantly different between healthy subjects (status=0) and disease (status=1). This gives us the opportunity to find the relationship between explanatory variables (22 features from F1 to F22) and the dependent variable (healthy status is 0 or 1), and build decent models to predict potential PD patients.

Table 3.1 Basic Statistics of All Explanatory Variables

Feature Number	Mean	StDev	Minimum	Q1	Median	Q3	Maximum
F1	154.23	41.39	88.33	117.57	148.79	182.77	260.11
F2	197.1	91.49	102.1	134.9	175.8	224.2	592.0
F3	116.32	43.52	65.48	84.29	104.31	140.02	239.17
F4	0.006	0.005	0.002	0.003	0.005	0.007	0.033
F5	4.4e-05	3.5e-05	7.0e-06	2.0e-05	3.0e-05	6.0e-05	2.6e-04
F6	0.003	0.003	0.001	0.002	0.003	0.004	0.02
F7	0.0034	0.003	0.001	0.002	0.0027	0.0034	0.02
F8	0.010	0.009	0.002	0.005	0.007	0.012	0.064
F9	0.030	0.019	0.001	0.017	0.023	0.038	0.119
F10	0.282	0.195	0.085	0.149	0.221	0.350	1.302
F11	0.016	0.010	0.005	0.008	0.013	0.020	0.056
F12	0.018	0.012	0.006	0.010	0.013	0.022	0.079
F13	0.024	0.017	0.007	0.013	0.018	0.029	0.138
F14	0.047	0.03	0.014	0.027	0.039	0.061	0.169
F15	0.025	0.04	0.001	0.006	0.117	0.026	0.315
F16	21.886	4.426	8.441	19.198	22.085	25.076	33.047
F17	0.499	0.104	0.257	0.421	0.496	0.588	0.685
F18	0.718	0.055	0.574	0.674	0.722	0.761	0.825
F19	-5.684	1.09	-7.965	-6.450	-5.721	-5.046	-2.434
F20	0.227	0.083	0.006	0.174	0.219	0.279	0.450
F21	2.382	0.383	1.423	2.099	2.362	2.636	3.671
F22	0.206	0.09	0.045	0.137	0.194	0.207	0.527

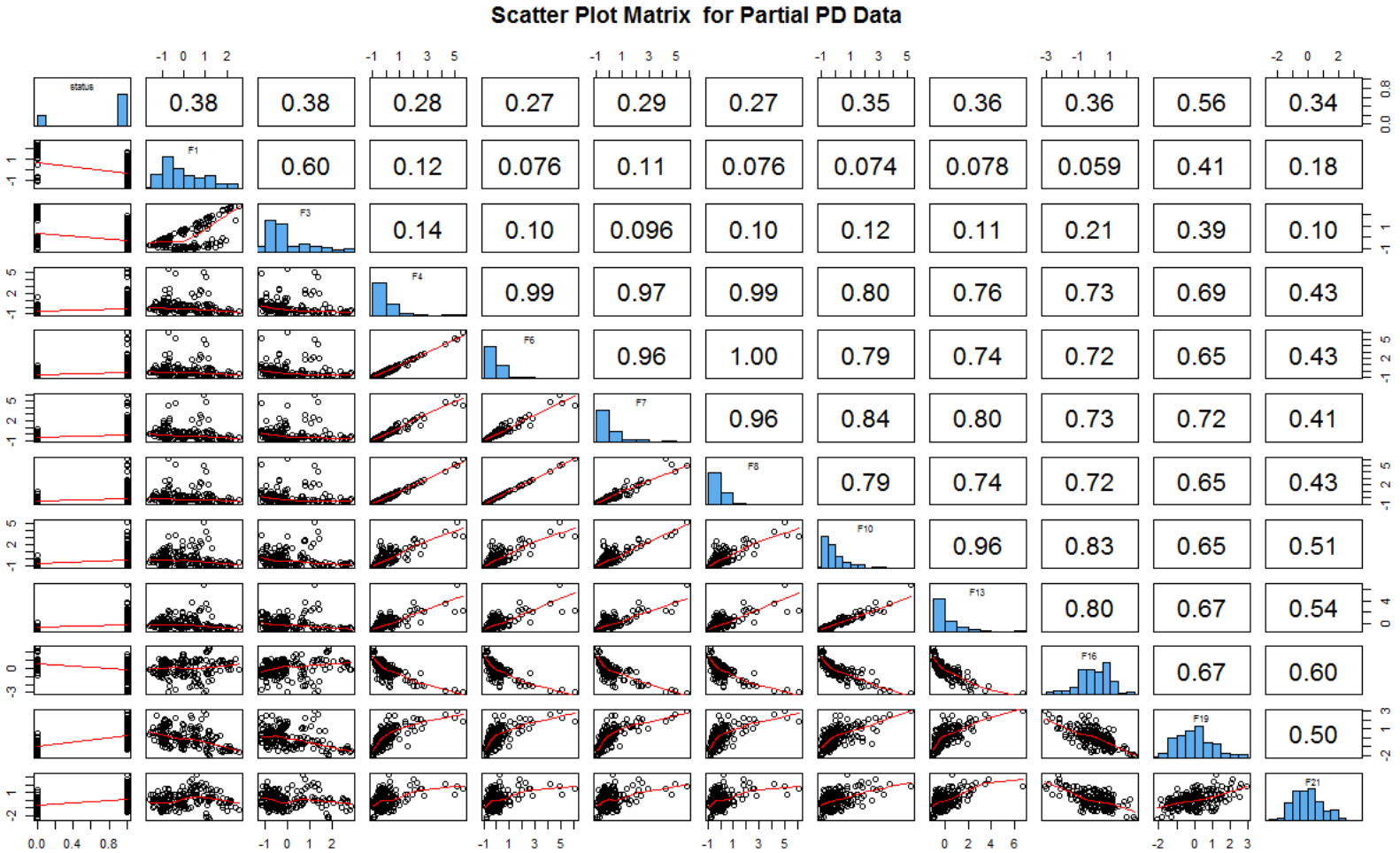


Figure 3.1 The Scatter Plot Matrix of Partial Explanatory Variables

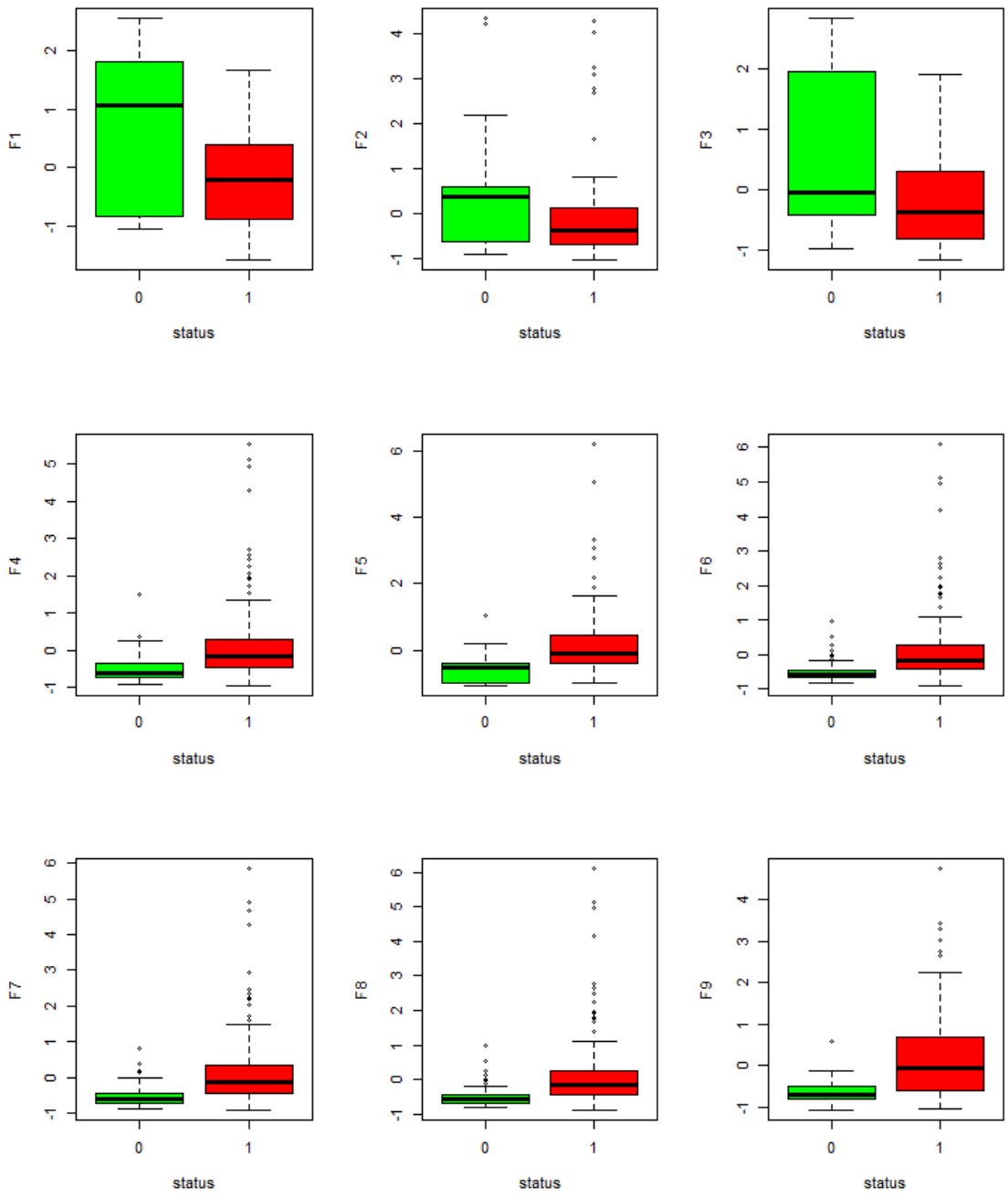


Figure 3.2 The Boxplot of Partial Explanatory Variables

3.3 The Training and Test Sets

For supervised learning, each observation paired consists of an input object and a desired output value. A supervised learning algorithm analyzes the training dataset and produces an inferred function, which should predict the correct output value for any valid input object. This requires the learning algorithm to generalize from the training dataset to unseen situations in a "reasonable" way.

Prior to applying any learning algorithm against the PD dataset, the dataset is split into training and test sets as discussed in the Section 2.2. For the current study, two-thirds (2/3) of the observations are used as the training set and one-third (1/3) are used as the test set. All statistical learning methods discussed in this thesis are applied to the same training and test sets. The histogram of response - “status” in the training set, test set, and original set (before splitting) is shown in Figure 3.3 to ensure that the ratios of the number of observations with PD to those without PD in different datasets are approximately equal.

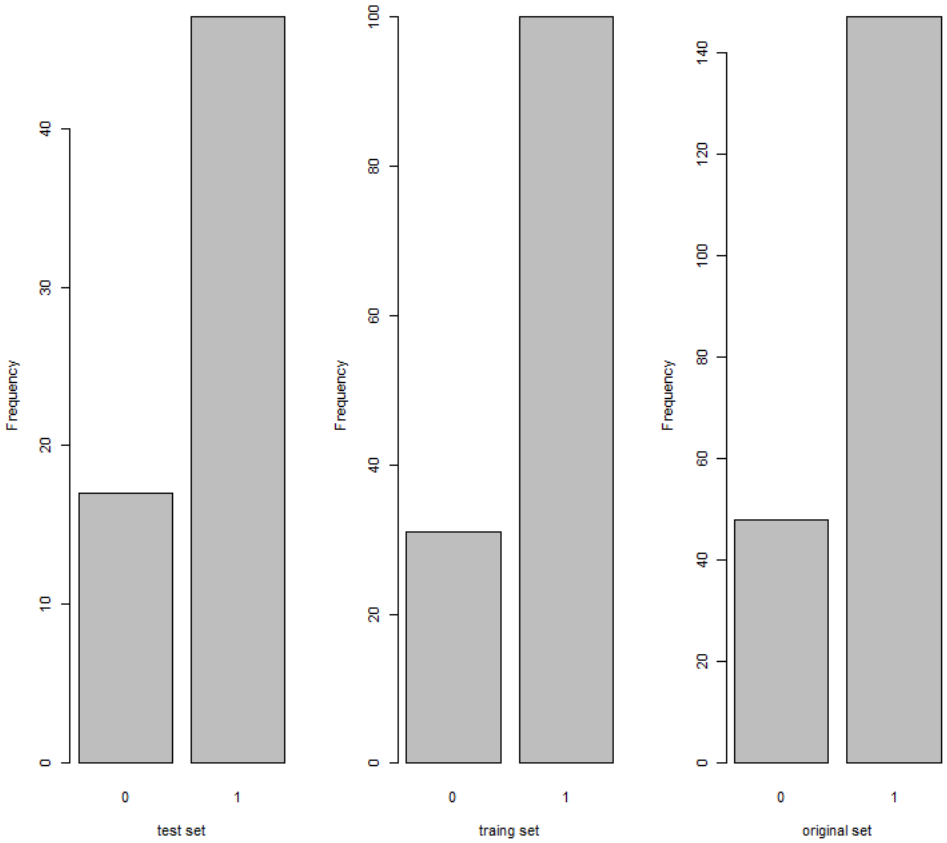


Figure 3.3 Histogram of Response in the Test Set, Training Set, and Original Set

4 STEPWISE LOGISTIC REGRESSION

4.1 Chapter Overview

This chapter will review and apply logistic regression model to the PD data. This chapter is organized as follows. (i) Section 4.2 provides the review of the logistic regression, (ii) Section 4.3 shows the application of the method, and (iii) Section 4.4 evaluates the performance of the model in predicting potential PD patients.

4.2 Review of the Method

Generalized linear model (GLM) is an extension of ordinary least square (OLS) linear regression model and allows the models to fit to the data following any member of a set of exponential-family distribution, which includes the normal, binomial, Poisson, gamma, and other distributions. The GLM generalizes the linear regression model by introducing a link function which transforms the expectation of the response variable to the linear model. As the logit link function is one of the most popular binomial functions, when response variables are binary (taking only the values of 0 and 1), the logistic regression can be applied to binary response and measures the relationship between a categorical response and one or many categorical or numerical explanatory variables.

For logistic regression, suppose $\mathbf{x}$ is a p-dimensional vector of explanatory variables and y is the binary response (for example, $y=1$ if a disease is present and $y=0$ if a disease is not present). Then the probability $\pi=\Pr\left(y=1|\mathbf{x}\right)$ is the response probability to be modeled, and the logistic regression has the form: $\log\left(\frac{\pi}{1-\pi}\right)=\alpha+\boldsymbol{\beta}'\mathbf{x}$ or equivalently $\pi=\frac{e^{\alpha+\boldsymbol{\beta}'\mathbf{x}}}{1+e^{\alpha+\boldsymbol{\beta}'\mathbf{x}}}$, where α is the intercept parameter and $\boldsymbol{\beta}=\left(\beta_1,\beta_2,\cdots,\beta_p\right)'$ is the vector of p dimensional slope parameters. The above formulas reveal that the probability of response $y=1$ is equal to the value of logistic function of the linear regression model. Unlike the OLS linear regression,

the logistic regression coefficients are usually estimated by using maximum likelihood estimation techniques.

As in many other regression models, deciding significant variables is essential in GLM as well. The response variable is more often only related to a subset of explanatory variables, but not all of them. It is preferred to yield a model with better prediction accuracy and interpretation. There are several classical and effective techniques for this purpose, and two popular ideas for variable selection for the GLM are: (i) subset selection, which identifies a subset of all predictors that are believed to be related to the response, and (ii) regularization, which penalizes models with the extreme coefficient of the regression model. For the regularization, the ridge regression and the lasso are two well-known methods for shrinking the regression coefficients towards zero by penalizing models with extreme parameter values. These two regularized regression algorithms will not be demonstrated in this study, and we only use the subset selection method to identify the final regression models. For example, employing massive computational power, "exhaustive regression" is designed to look at all possible linear models and recommends the best one. However, the exhaustive regression cannot be applied easily with too many explanatory variables due to computational constraints. The stepwise method explores a more restricted set of models, which is computationally efficient and an attractive alternative to the best subset selection. There are three classical approaches of stepwise methods: (i) forward selection, (ii) backward selection, and (iii) mixed selection.

In this study, we apply the stepwise method on GLM to select the most significant variables to make prediction and choose a model by evaluating Akaike information criterion ($AIC = -2 \times log\, L + 2 \times p$, where L is the likelihood and p is the number of estimated parameters in the GLM.) in a backward stepwise algorithm. To seek the model in a set of candidate models that gives the best balance between model fit and complexity, the model

with lowest AIC is preferred.

4.3 Application of the Method

For the PD data, we have 195 observations in the original dataset; however we have split the data into the training set, which has 131 observations, and the test set, which has 64 observations. In this section, the logistic regression (with stepwise) is constructed based on the training set.

One of the critical assumptions of regression is that the independent variables are not the linear combination of each other. Before building the logistic regression model, we will calculate the VIF (variance inflation factor) for each variable. The variable with VIF greater than 5 should not be included in the regression model. A common rule of thumb is that if the VIF is greater than 5, then the multicollinearity is high. Table 4.1 displays the VIF of all variables and we will see that only variables F2, F3, F17, F18, F20, and F22 have the VIFs less than 5, which means that we will only use these six variables for building the logistic regression model.

Table 4.1 VIF for Each Variable

Variable	VIF
F1	8.36
F2	1.79
F3	2.71
F4	206.28
F5	57.15
F6	1535941.54
F7	126.78
F8	1535616.63
F9	808.62
F10	121.70
F11	17442562.63
F12	100.48
F13	70.09
F14	17431523.44
F15	14.93
F16	8.68
F17	3.90
F18	3.69
F19	19.36
F20	2.68
F21	3.49
F22	29.93

The R function - “glm” with link function “logit” is used to build a logistic regression model between the dependent variable and the potential predictors, and the R function – “stepAIC” is used to perform stepwise model selection by AIC. The main effects regression model will be constructed as follows.

The initial full model is:

$Status \sim F2 + F3 + F17 + F18 + F20 + F21$.

The final model after stepwise selection is:

$Status \sim F2 + F17 + F18 + F20 + F21$.

Table 4.2 provides the summary results for the initial model's estimates, and Table 4.3 provides the summary results for the final model's estimates (after stepwise selection). We find that the final model is with the lowest AIC of 106.44 and all the parameter estimates for selected predictors (F2, F17, F18, F20, and F21) are roughly significant (at 0.1 level). The performance of the final main effects model will be evaluated against the test dataset in the next section.

Table 4.2 Initial Main Effects Model (Full Model) Estimates of Logistic Regression

	Estimate	**Std. Error**	**Z Value**	**Pr(>|z|)**
(Intercept)	1.77	0.35	5.12	0.00
F2	-0.51	0.29	-1.74	0.08
F3	-0.36	0.31	-1.14	0.26
F17	0.46	0.33	1.40	0.16
F18	0.70	0.35	1.99	0.05
F20	0.63	0.41	1.55	0.12
F21	1.19	0.43	2.76	0.01

Table 4.3 Final Main Effects Model Estimates of Logistic Regression

	Estimate	**Std. Error**	**Z Value**	**Pr(>|z|)**
(Intercept)	1.83	0.35	5.28	0.00
F2	-0.49	0.29	-1.68	0.09
F17	0.61	0.30	2.03	0.04
F18	0.78	0.34	2.29	0.02
F20	0.64	0.40	1.61	0.11
F21	1.18	0.42	2.84	0.00

The above logistic regression model is a main effects model. However, we might believe that including interaction terms in the regression can make the model more flexible and expect a better regression model. Interaction effects represent the combined effects of variables on the dependent variable. The impact of one variable depends on the level of the other variable when an interaction effect is present. To test if interactions are significant in the logistic regression model, we will construct a new logistic regression model with interaction terms (only two-way interactions will be considered in this study). The variable selection scheme will be applied to remove unnecessary interaction terms and determine whether the interaction terms are needed in the predictive model. The variable selection scheme can also compare the performance of the regression model with and without interactions in predicting potential Parkinson's disease.

The initial full model with main effects and interactions is:

$$\begin{aligned} Status \sim\ & F2 + F3 + F17 + F18 + F20 + F21 + F2 \times F3 + F2 \times F17 \\ & + F2 \times F18 + F2 \times F20 + F2 \times F21 + F3 \times F17 + F3 \times F18 + \\ & F3 \times F20 + F3 \times F21 + F17 \times F18 + F17 \times F20 + F17 \times F21 \\ & + F18 \times F20 + F18 \times F21 + F20 \times F21. \end{aligned}$$

After the stepwise selection by AIC, we have the final model as:

$$\begin{aligned} Status \sim\ & F2 + F3 + F17 + F18 + F20 + F21 \\ & + F2 \times F18 + F2 \times F20 + F2 \times F21 + F3 \times F17 + F3 \times F18 \\ & + F3 \times F21 + F17 \times F20 + F18 \times F20 + F18 \times F21. \end{aligned}$$

The summary results for the initial main effects and interactions model's estimates and the final model's estimates are provided by Tables 4.4 and 4.5, respectively. We observe that almost all the parameter estimates for predictors in the final model are significant (at 0.1 level), but the regressors F17 and F21 have relatively large *p*-values. The reason for including the predictors F17 and F12 in the model is due to the hierarchical principle, which states that if we include an interaction in a model, we should also include the main effects, even if the *p*-values associated with their coefficients are not significant (James et al., 2013). We note

that the AIC of the final model is 84.90, while the AIC of the full model is 93.35. The performance of the final main effects and interactions model will be evaluated against the test dataset in the next section. Furthermore, we will compare performance of the models with the interaction and without interaction for the recommendation of the prediction model.

Table 4.4 Initial Main Effects and Interactions Model (Full Model) Estimates of Logistic Regression

	Estimate	**Std. Error**	**Z Value**	**Pr(>|z|)**
(Intercept)	5.95	1.97	3.02	0.00
F2	3.19	2.92	1.09	0.27
F3	-2.09	1.39	-1.50	0.13
F17	0.60	1.15	0.52	0.60
F18	2.54	1.44	1.77	0.08
F20	5.83	2.35	2.48	0.01
F21	0.34	1.59	0.21	0.83
F2:F3	-2.80	2.29	-1.22	0.22
F2:F17	0.00	1.45	0.00	1.00
F2:F18	5.77	2.44	2.36	0.02
F2:F20	11.17	4.03	2.77	0.01
F2:F21	-5.67	2.23	-2.54	0.01
F3:F17	-1.49	1.19	-1.25	0.21
F3:F18	-2.73	1.42	-1.91	0.06
F3:F20	-0.33	2.23	-0.15	0.88
F3:F21	2.71	2.11	1.28	0.20
F17:F18	0.71	0.67	1.06	0.29
F17:F20	1.27	0.97	1.30	0.19
F17:F21	0.07	1.31	0.06	0.95
F18:F20	2.02	1.03	1.96	0.05
F18:F21	-2.57	1.21	-2.13	0.03
F20:F21	-0.26	1.09	-0.24	0.81

Table 4.5 Final Main Effects and Interactions Model Estimates of Logistic Regression

	Estimate	**Std. Error**	**Z Value**	**Pr(>|z|)**
(Intercept)	5.65	1.63	3.48	0.00
F2	4.63	2.33	1.99	0.05
F3	-2.20	1.14	-1.93	0.05
F17	0.73	0.60	1.21	0.22
F18	2.69	1.37	1.97	0.05
F20	5.86	1.91	3.08	0.00
F21	-0.08	0.99	-0.08	0.94
F2:F18	5.09	2.13	2.40	0.02
F2:F20	11.10	3.30	3.36	0.00
F2:F21	-5.28	1.79	-2.96	0.00
F3:F17	-1.38	0.82	-1.69	0.09
F3:F18	-2.46	1.16	-2.11	0.03
F3:F21	1.59	0.84	1.90	0.06
F17:F20	1.51	0.85	1.77	0.08
F18:F20	1.96	0.97	2.02	0.04
F18:F21	-2.23	1.07	-2.08	0.04

4.4 Model evaluation

The model performance is quantified by scoring the test set and computing the predicted probability of PD for each patient. Using the cut-off value of 0.5 for the predicted probability of Parkinson's disease, we derive the proportion of observations in the test set correctly classified as PD patients or correctly classified as non-PD patients by the logistic regression model. The sensitivity and specificity are achieved at the same time. Figures 4.1 and 4.2 below display the posterior probability of each observation with Parkinson's disease for the main effects model and the model with main effects and interactions. Tables 4.6 and 4.7 provide the summary results (cut-off value of 0.5) of applying the model to the test set for these two models respectively.

Table 4.6 Actual versus Predicted Parkinson's Disease in the Test Set (Main Effects Logistic Regression Model)

	Predicted		
Actual	**0**	**1**	**Total**
0	12	5	17
1	3	44	47
Total	15	49	64

Table 4.7 Actual versus Predicted Parkinson's Disease in the Test Set (Main Effects and Interactions Logistic Regression Model)

	Predicted		
Actual	**0**	**1**	**Total**
0	15	2	17
1	4	43	47
Total	19	45	64

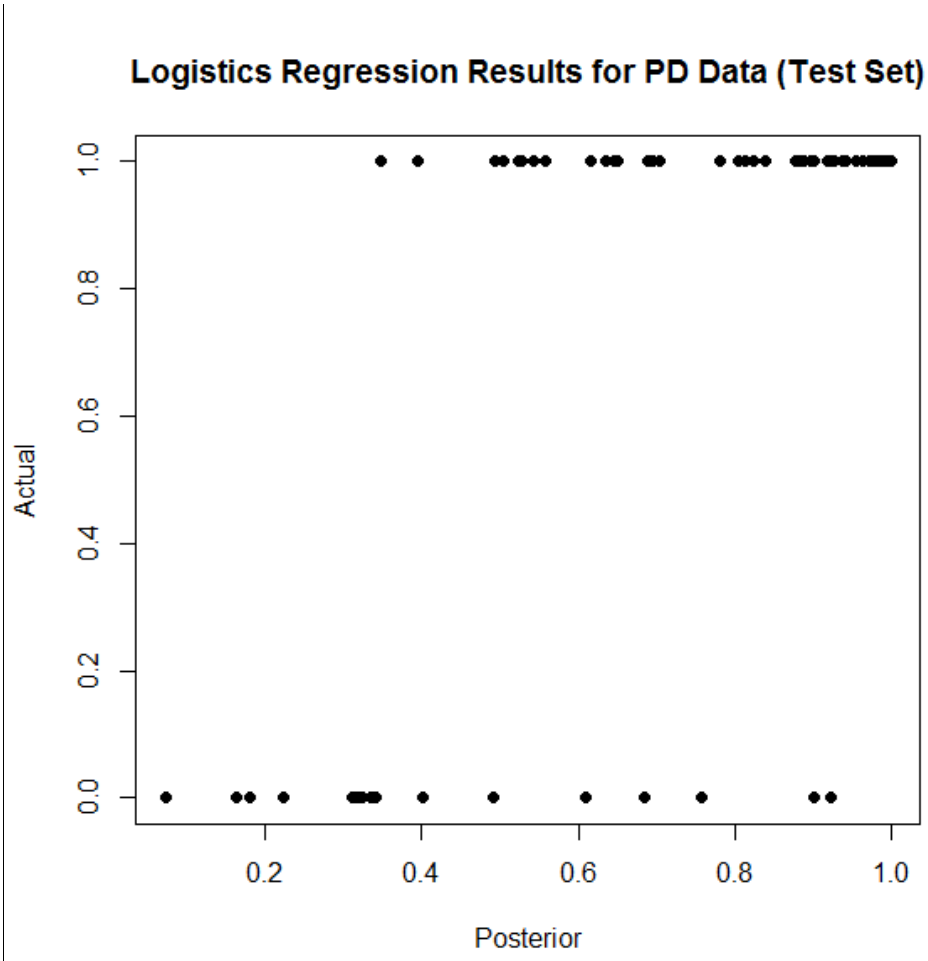


Figure 4.1 The Posterior Probability of Each Observation in the Test Set (Main Effects Logistic Regression Model)

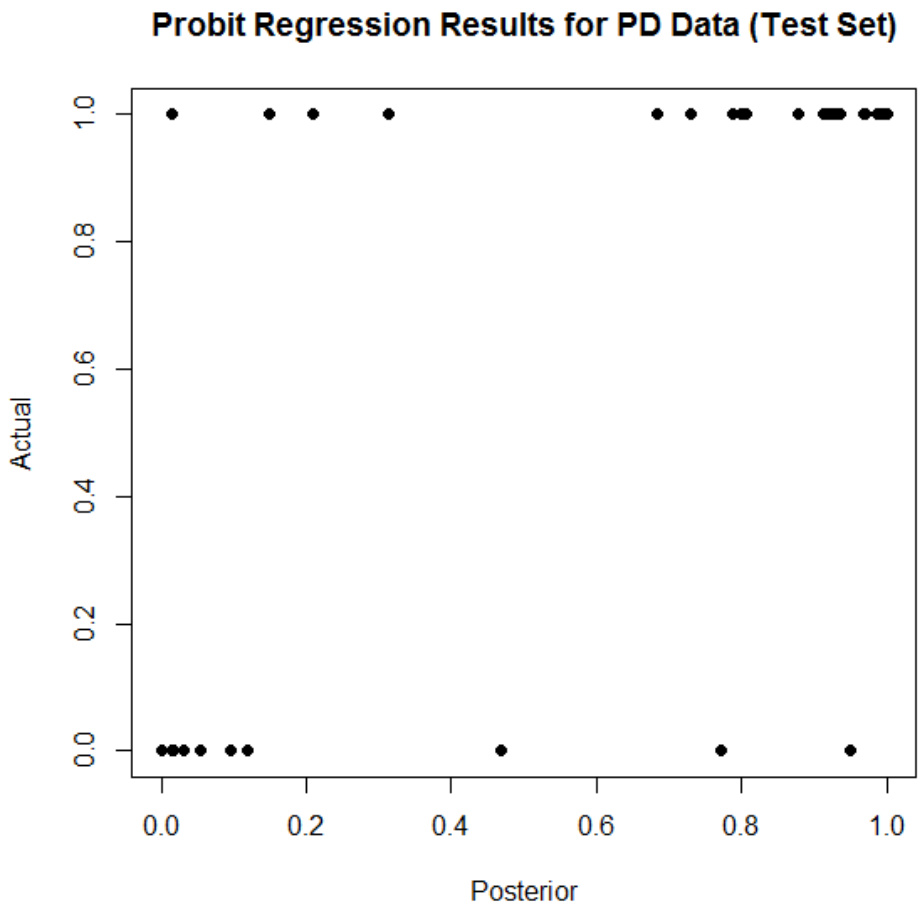


Figure 4.2 The Posterior Probability of Each Observation in the Test Set (Main Effects and Interactions Logistic Regression Model)

By comparing the above two confusion matrixes, we observe that the main effects and interaction models offers us a better prediction performance. For this better logistic regression model, out of a total of 64 observations in the test set sample, 47 are classified as PD patients while 17 are classified as non-PD patients. The predicted observations with PD are 45 against 19 observations without PD. Forty-three observations with PD are correctly predicted while 15 observations without PD are correctly predicted. This leads to a proportion of correctly predicted observations of about 90.32%. The sensitivity is 91.49%, which indicates that 91.49% of observations in the test set identified as "PD patients" are classified by the predictive model as "PD patients". The specificity is 88.24%, which indicates that the 88.24% of observations in the test set actually identified as "Non-PD patients" are truly classified by the predictive model as "Non-PD patients."

However, we notice that the threshold (cut-off value) used above is fixed at 0.5 and the performance of the models might be improved if we vary the thresholds. The ROC (Receiver Operating Characteristic) curve is a popular graph that illustrates the performance of a binary classifier through simultaneously displaying the true positive and false positive rates for all possible thresholds. Figure 4.3 displays a sample of ROC curve in red. The dotted line represents the "no information" classifier, and the ideal ROC curve passes through the top left corner, indicating a high true positive rate and low false positive rate. The overall performance of a classifier, summarized over all possible thresholds, is given by the area under the ROC curve (AUC). As an ideal ROC curve passes through the top left corner. A larger area under the curve (AUC) implies a better classifier. The ROC analysis provides tools to compare and select possibly optimal models.

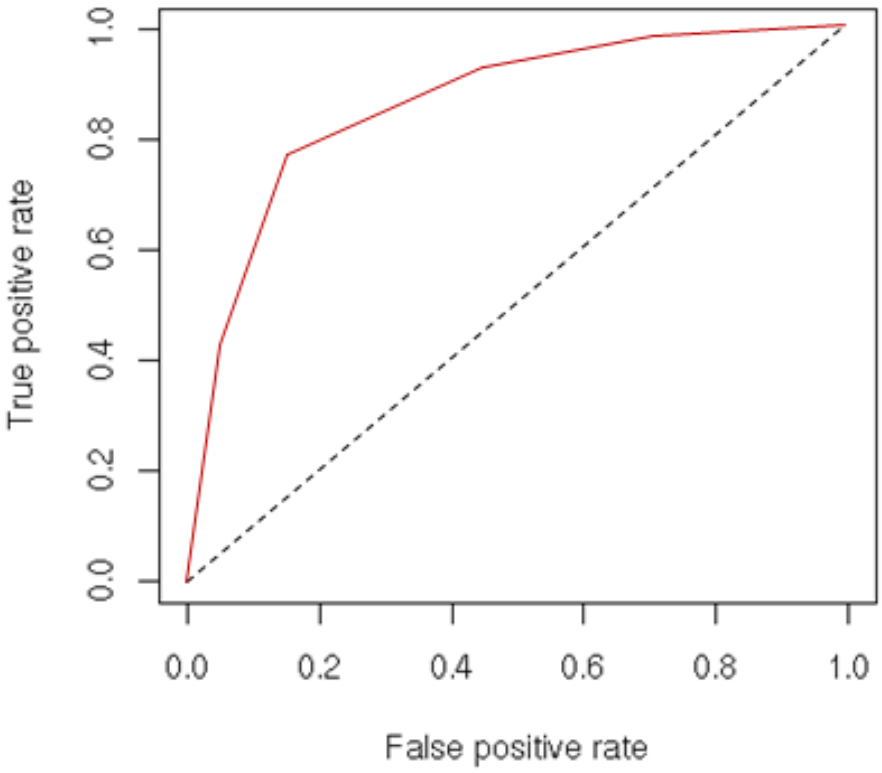


Figure 4.3 A Sample of ROC Curve

Back to the logistic regression, for both final models, we adjust the cut-off value from 0 to 1 and observe the false positive and true positive rates of the prediction in the test set. Figure 4.4 shows the ROC curves of both logistic regression models. The ROC curves conclude that logistic regression model with main effects and interaction terms is better than the main effects model, which is consistent with the conclusion we get by comparing the fusion matrixes.

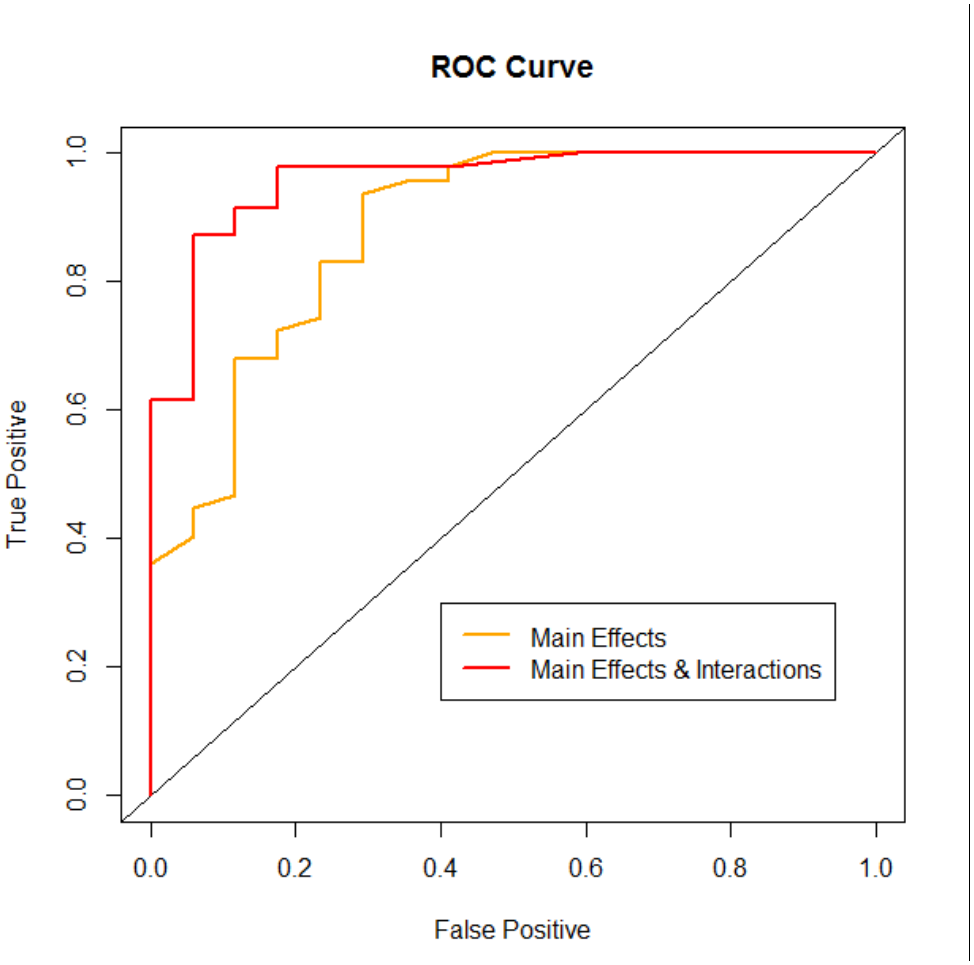


Figure 4.4 The ROC Curves for Logistic Regression Models

5 STEPWISE PROBIT REGRESSION

5.1 Chapter Overview

This chapter reviews and applies a probit regression model to the PD data. This chapter is organized as follows. (i) Section 5.2 provides the review of the probit regression, (ii) Section 5.3 shows the application of the method, and (iii) Section 5.4 evaluates the performance of the model in predicting potential PD patients.

5.2 Review of the Method

The probit regression is very similar to the logistic regression discussed in the previous chapter. As a generalized linear model, the probit regression, which employs the probit link function, is another type of regression when the dependent variable is binary. Again, let us suppose the $\mathbf{x}$ is a vector of explanatory variables and y is the binary response (0 or 1), then $\pi = \Pr\left(y = 1 \middle| \mathbf{x}\right) = \Phi\left(\alpha + \boldsymbol{\beta}'\mathbf{x}\right)$, where Pr denotes the probability and Φ is the cumulative distribution function (CDF) of the standard Gaussian/Normal distribution response model. The maximum likelihood estimation method can be used to estimate the probit regression model.

5.3 Application of the Method

In this section, the probit regression (with stepwise) is constructed based on the training set as usual. As in the previous analysis for the logistic regression, the main effects model and the model with main effects and interactions terms are thoroughly explored in this section.

The R function - “glm” with link function “probit” is used to build a probit regression model, and the R function - “stepAIC” is used to perform stepwise model selection by AIC.

We conduct the analysis for the main effects from Section 4.3’s VIF calculation to the probit regression model.

From it we start with the following initial full model:

Status ~ F2 + F3 + F17 + F18 + F20 + F21.

The final model after stepwise selection is:

Status ~ F2 + F17 + F18 + F20 + F21.

Tables 5.1 and 5.2 provide the summary results for the estimates of the initial and final models respectively. We find that the final model comes with the lowest AIC of 106.39 and all parameter estimates for final predictors (F2, F17, F18, F20, and F21) are significant or nearly significant (at 0.1 level).

Table 5.1 Initial Main Effects Model (Full Model) Estimates of Probit Regression

	Estimate	**Std. Error**	**Z Value**	**Pr(>|z|)**
(Intercept)	1.00	0.18	5.47	0.00
F2	-0.30	0.16	-1.83	0.07
F3	-0.22	0.18	-1.24	0.21
F17	0.27	0.19	1.45	0.15
F18	0.39	0.20	2.01	0.04
F20	0.36	0.22	1.58	0.11
F21	0.66	0.23	2.83	0.00

Table 5.2 Final Main Effects Model Estimates of Probit Regression

	Estimate	**Std. Error**	**Z Value**	**Pr(>|z|)**
(Intercept)	1.04	0.18	5.69	0.00
F2	-0.29	0.16	-1.76	0.08
F17	0.36	0.17	2.12	0.03
F18	0.45	0.19	2.37	0.02
F20	0.35	0.22	1.58	0.11
F21	0.67	0.23	2.97	0.00

To test whether adding the interaction terms can improve the prediction performance, we conduct the analysis for the probit regression model with main effects and interactions. As we did in the previous chapter, we only consider the two-way interactions to avoid making the model unnecessarily complicated and over-fitting.

The initial model with main effects and interaction terms is:

$$Status \sim F2 + F3 + F17 + F18 + F20 + F21 + F2 \times F3 + F2 \times F17 + F2 \times F18 + F2 \times F20 + F2 \times F21 + F3 \times F17 + F3 \times F18 + F3 \times F20 + F3 \times F21 + F17 \times F18 + F17 \times F20 + F17 \times F21 + F18 \times F20 + F18 \times F21 + F20 \times F21.$$

After the stepwise selection by AIC, we have the final model as:

$$Status \sim F2 + F3 + F17 + F18 + F20 + F21 + F2 \times F18 + F2 \times F20 + F2 \times F21 + F3 \times F17 + F3 \times F18 + F3 \times F21 + F17 \times F20 + F18 \times F20 + F18 \times F21.$$

Tables 5.3 and 5.4 provide the results for the estimates of the full and final models. The final model with main effects and interactions comes with AIC value of 84.3 and almost all its predictors are significant (at 0.1 level). We may notice that the predictors F17 and F21 have relatively large *p*-values, but they are still included in the final model, this is due to the hierarchical principle that has been discussed in Section 4.3.

In the next section, we will evaluate and compare the performance of the model with main effect terms and the model with main effects and interactions terms.

Table 5.3 Initial Main Effects and Interactions Model (Full Model) Estimates of Probit Regression

	Estimate	**Std. Error**	**Z Value**	**Pr(>|z|)**
(Intercept)	3.33	1.07	3.12	0.00
F2	1.93	1.59	1.22	0.22
F3	-1.18	0.76	-1.55	0.12
F17	0.30	0.61	0.50	0.62
F18	1.46	0.78	1.87	0.06
F20	3.43	1.27	2.70	0.01
F21	0.04	0.86	0.04	0.96
F2:F3	-1.43	1.24	-1.16	0.25
F2:F17	-0.01	0.82	-0.02	0.99
F2:F18	3.24	1.33	2.44	0.01
F2:F20	6.58	2.22	2.96	0.00
F2:F21	-3.30	1.22	-2.70	0.01
F3:F17	-0.90	0.67	-1.35	0.18
F3:F18	-1.56	0.79	-1.97	0.05
F3:F20	-0.10	1.22	-0.08	0.93
F3:F21	1.42	1.15	1.23	0.22
F17:F18	0.37	0.38	0.97	0.33
F17:F20	0.78	0.51	1.52	0.13
F17:F21	-0.06	0.70	-0.08	0.94
F18:F20	1.17	0.58	2.00	0.05
F18:F21	-1.46	0.67	-2.19	0.03
F20:F21	-0.14	0.58	-0.24	0.81

Table 5.4 Final Main Effects and Interactions Model Estimates of Probit Regression

	Estimate	**Std. Error**	**Z Value**	**Pr(>|z|)**
(Intercept)	3.20	0.87	3.69	0.00
F2	2.62	1.26	2.08	0.04
F3	-1.16	0.61	-1.90	0.06
F17	0.45	0.34	1.33	0.19
F18	1.56	0.74	2.10	0.04
F20	3.32	1.03	3.23	0.00
F21	-0.03	0.54	-0.05	0.96
F2:F18	2.88	1.16	2.48	0.01
F2:F20	6.35	1.80	3.53	0.00
F2:F21	-2.97	0.98	-3.04	0.00
F3:F17	-0.76	0.45	-1.71	0.09
F3:F18	-1.38	0.64	-2.16	0.03
F3:F21	0.92	0.45	2.06	0.04
F17:F20	0.86	0.45	1.92	0.06
F18:F20	1.11	0.54	2.05	0.04
F18:F21	-1.26	0.58	-2.16	0.03

5.4 Model evaluation

The model performance is quantified by scoring the test set and computing for each patient the predicted probability of PD. Using the cut-off value of 0.5 for predicted probability of Parkinson's disease, we can calculate the accuracy, sensitivity, and specificity for the probit regression model by using the test set.

Figures 5.1 and 5.2 display the posterior probability of each observation with Parkinson's disease for the final main effects model and main effects and interactions model respectively. Tables 5.5 and 5.6 provide the confusion matrixes that summarize results (cut-off value of 0.5) of applying both models to the test set separately.

Table 5.5 Actual versus Predicted Parkinson's Disease in the Test Set (Main Effects Probit Regression Model)

	Predicted		
Actual	**0**	**1**	**Total**
0	12	5	17
1	4	43	47
Total	16	48	64

Table 5.6 Actual versus Predicted Parkinson's Disease in the Test Set (Main Effects and Interactions Probit Regression Model)

	Predicted		
Actual	**0**	**1**	**Total**
0	15	2	17
1	4	43	47
Total	19	45	64

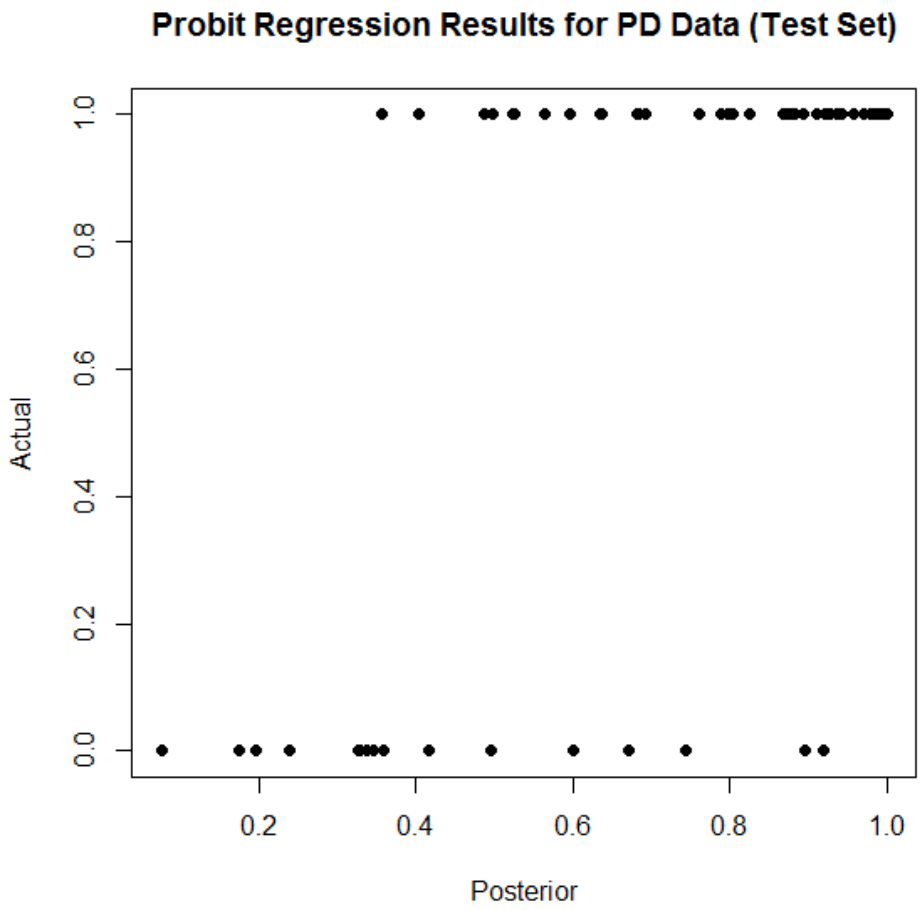


Figure 5.1 The Posterior Probability of Each Observation in the Test Set (Main Effects Probit Regression Model)

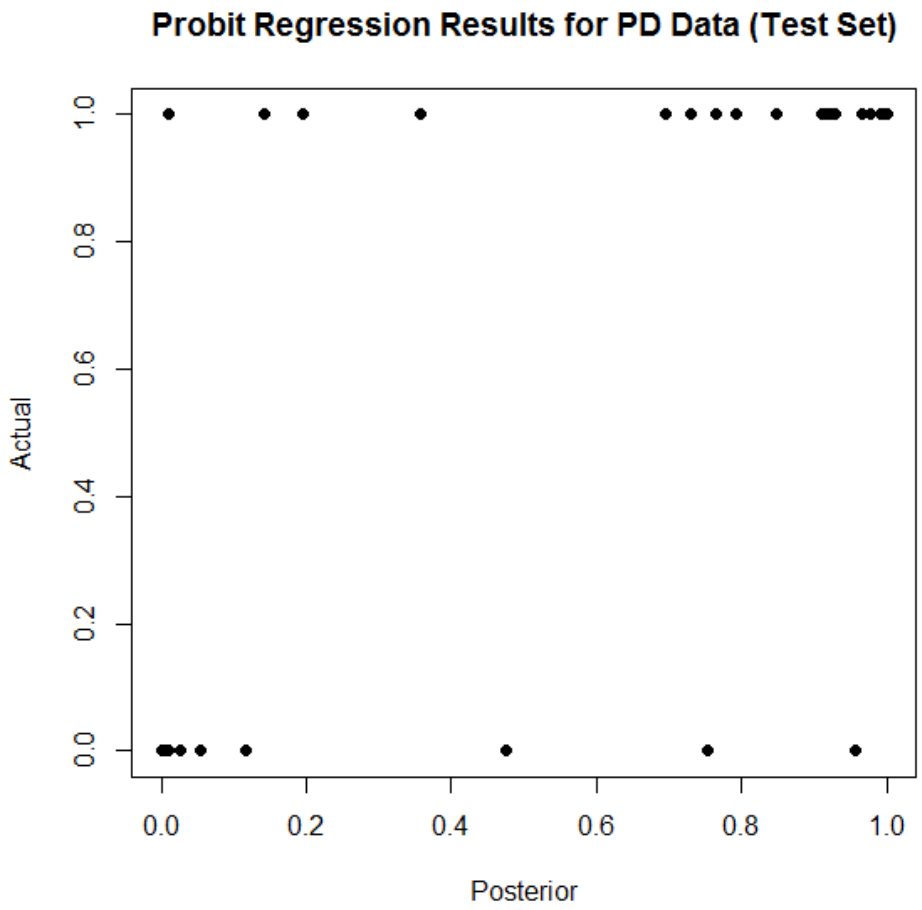


Figure 5.2 The Posterior Probability of Each Observation in the Test Set (Main Effects and Interactions Probit Regression Model)

For the probit regression, confusion matrixes reveal that the model with main effects and interactions is better and its overall correct classification rate is 90.63%, its sensitivity is 91.4%, and its specificity is 88.24%. Furthermore, Figure 5.3 shows ROC curves for both models, which demonstrate that the probit regression model with main effects and interactions outperforms the main effects model.

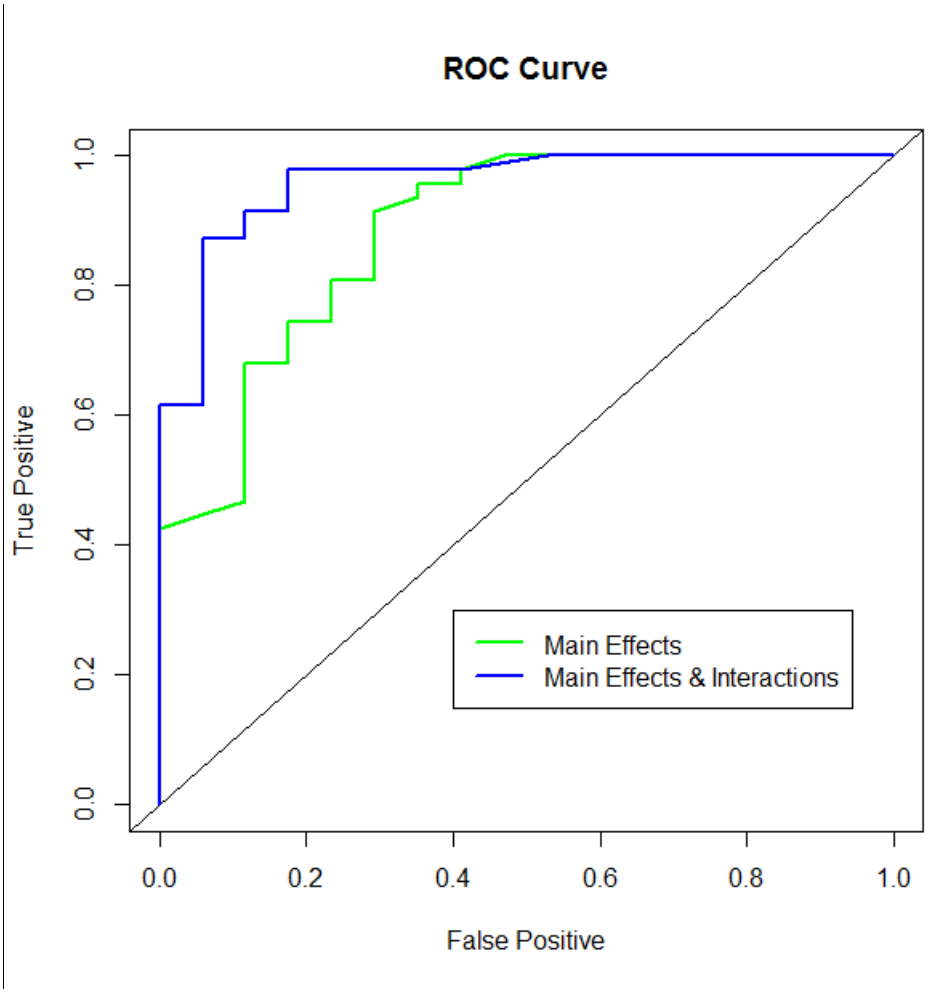


Figure 5.3 The ROC Curves for **Probit** Regression Models

6 LINEAR/QUADRATIC DISCRIMINANT ANALYSIS

6.1 Chapter Overview

This chapter reviews and applies Linear/ Quadratic Discriminant Analysis to the PD data. The sections of this chapter are organized as follows. (i) Section 6.2 provides the review of the Linear/Quadratic Discriminant Analysis, (ii) Section 6.3 shows the application of the method, and (iii) Section 6.4 evaluates the performance of the models.

6.2 Review of the Method

Assume that we have a set of observations $\mathbf{X}$ with a known class Y and we would like to predict Y given only an observation x. To do this, it is assumed that $\mathbf{X} = (\mathbf{X_1}, \mathbf{X_2}, \cdots, \mathbf{X_p})$ follows a multivariate normal distribution with a class-specified mean vector and a common variance-covariance matrix. We write $\mathbf{X} \sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, where $\boldsymbol{\mu}$ (the mean of $\mathbf{X}$) is a vector with p components, and $\boldsymbol{\Sigma}$ is the $p \times p$ variance-covariance matrix. Furthermore, the multivariate normal distribution has the form:

$f_{\mathbf{X}}(x_1, x_2, \cdots, x_p) = \frac{1}{\sqrt{(2\pi)^p |\boldsymbol{\Sigma}|}} \exp\left(-\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^T \Pi^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right)$, where $|\boldsymbol{\Sigma}|$ is the determinant of $\boldsymbol{\Sigma}$. The linear discriminant analysis (LDA) classifier assumes that the observation in the k^{th} class comes from a multivariate normal distribution $N(\mu_k, \boldsymbol{\Sigma})$, which μ_k is a class-specific mean vector and $\boldsymbol{\Sigma}$ is the common variance-covariance matrix across all classes. Moreover, let $f_k(X) \equiv \Pr(X = \mathbf{x}, y = k)$ denotes the density function of X for an observation that comes from the k^{th} class, and π_k denotes the prior probability that a randomly chosen observation comes from the k^{th} class. The Bayes' theorem shows that

$\Pr(y = k | X = \mathbf{x}) = \frac{\pi_k f_k(\mathbf{x})}{\sum_{i=1}^{K} \pi_i f_i(\mathbf{x})}$. Under the aforementioned assumptions, by combining the

density function for the k^{th} class with the Bayes' theorem, the optimal solution is to predict observation $X = \mathbf{x}$ as being from the class with the largest value of $\delta_k(\mathbf{x}) = \mathbf{x^T \Sigma^{-1} \mu_k} - 1/2\,\mathbf{\mu^T \Sigma^{-1} \mu_k} + \log \pi_k$, which is a linear function of $\mathbf{x}$. The x implies that the LDA decision rule depends on $\mathbf{x}$ only through a linear combination of its elements. In the application, we have to estimate μ_i, π_i, Σ (where i = 1,2,..,k) using training data:

$\hat{\pi}_k = N_k / N$ where N_k is the number of class-k observations and N is the number of all observations; $\hat{\mu}_k = \sum_{g_i = k} x_i / N_k$; and $\hat{\mathbf{\Sigma}} = \sum_{k=1}^{K} \sum_{g_i = k} (x_i - \hat{\mu}_k)(x_i - \hat{\mu}_k)^T / (N - K)$.

For the LDA classifier, we assume that the observations within each class are from a multivariate normal distribution with a class-specified mean vector and a common variance-covariance matrix. For the quadratic discriminate analysis (QDA) classifier, we assume that each class has its own variance-covariance matrix. In the QDA, we suppose that the observation in the k^{th} class comes from a multivariate normal distribution $N(\mu_k, \mathbf{\Sigma}_k)$, where μ_k and Σ_k are a class-specific mean vector and a variance-covariance matrix respectively. Under this assumption the Bayes' optimal solution suggests to assign an observation $X = \mathbf{x}$ to the class with the largest value of $\delta_k(\mathbf{x}) = -1/2(\mathbf{x - \mu_k})^T \mathbf{\Sigma}_k^{-1} (\mathbf{x - \mu_k}) + \log \pi_k$, which is a quadratic function of $\mathbf{x}$. The estimates of μ_i, π_i, Σ_i (where i = 1, 2,..., k) are similar to those for LDA except that the variance-covariance matrix should be calculated for each class.

6.3 Application of the Method

In this section, the LDA and QDA models are constructed based on the training set. The R function – "lda" and "qda" in the "MASS" package are used to build these models. We perform the LDA and QDA on the training set in order to predict potential PD patients. For the training set, the prior probability of the PD patients and non-PD patients are 0.24 and 0.76

respectively. Table 6.1 shows the mean of each feature in two groups (status=0 and status=1).

Table 6.1 Group Means of Each Feature in the Training Set

	F1	**F2**	**F3**	**F4**	**F5**
0	0.3826484	0.3161087	0.4693384	-0.4226056	-0.4907257
1	-0.3666264	-0.2695731	-0.2295525	0.1816655	0.2653796
	F6	**F7**	**F8**	**F9**	**F10**
0	-0.4275901	-0.44707	-0.4274885	-0.6340664	-0.5962047
1	0.1793565	0.196682	0.1793766	0.2562279	0.2406064
	F11	**F12**	**F13**	**F14**	**F15**
0	-0.6054963	-0.6115498	-0.6125951	-0.6055192	-0.2746026
1	0.253561	0.246001	0.2410501	0.2535492	0.1162346
	F16	**F17**	**F18**	**F19**	**F20**
0	0.526126	-0.4372229	-0.4479111	-0.8962023	-0.7089082
1	-0.2749772	0.244182	0.270581	0.3318493	0.2544097
	F21	**F22**			
0	-0.60200699	-0.8556873			
1	0.09608217	0.3227712			

6.4 Model evaluation

The LDA and QDA models fit to the 131 training observations and their performances are quantified by scoring the test set and computing for each patient the predicted probability of PD. Then using the cut-off value of 0.5 for predicted probability of Parkinson's disease, we derive correct classification rate, sensitivity, and specificity in the test set. Tables 6.2 and 6.3 show their prediction performances (LDA and QDA respectively) on the test set with 64 observations (cut-off value of 0.5). Figures 6.1 and 6.2 display the posterior probability of each observation of Parkinson's disease for LDA and QDA respectively.

Table 6.2 Actual Versus Predicted Parkinson's Disease in the Test Set (LDA)

	Predicted		
Actual	**0**	**1**	**Total**
0	12	5	17
1	4	43	47
Total	16	48	64

Table 6.3 Actual Versus Predicted Parkinson's Disease in the Test Set (QDA)

	Predicted		
Actual	**0**	**1**	**Total**
0	11	6	17
1	0	47	47
Total	11	53	64

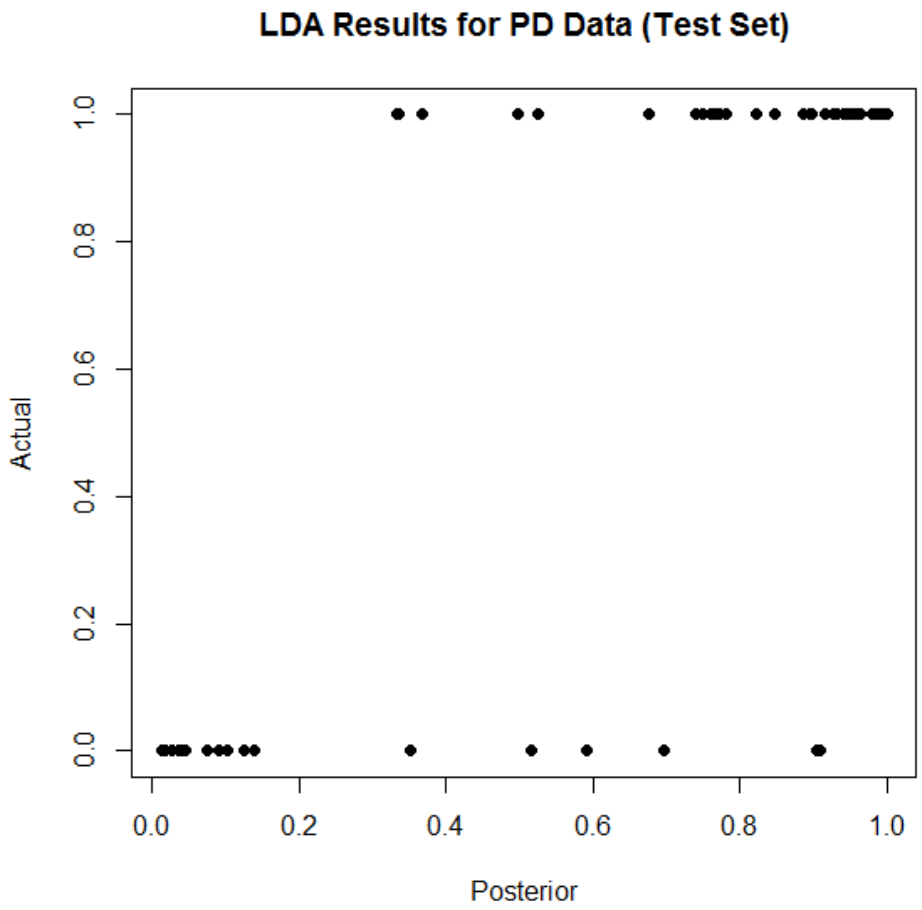


Figure 6.1 The Posterior Probability of Each Observation in the Test Set (LDA)

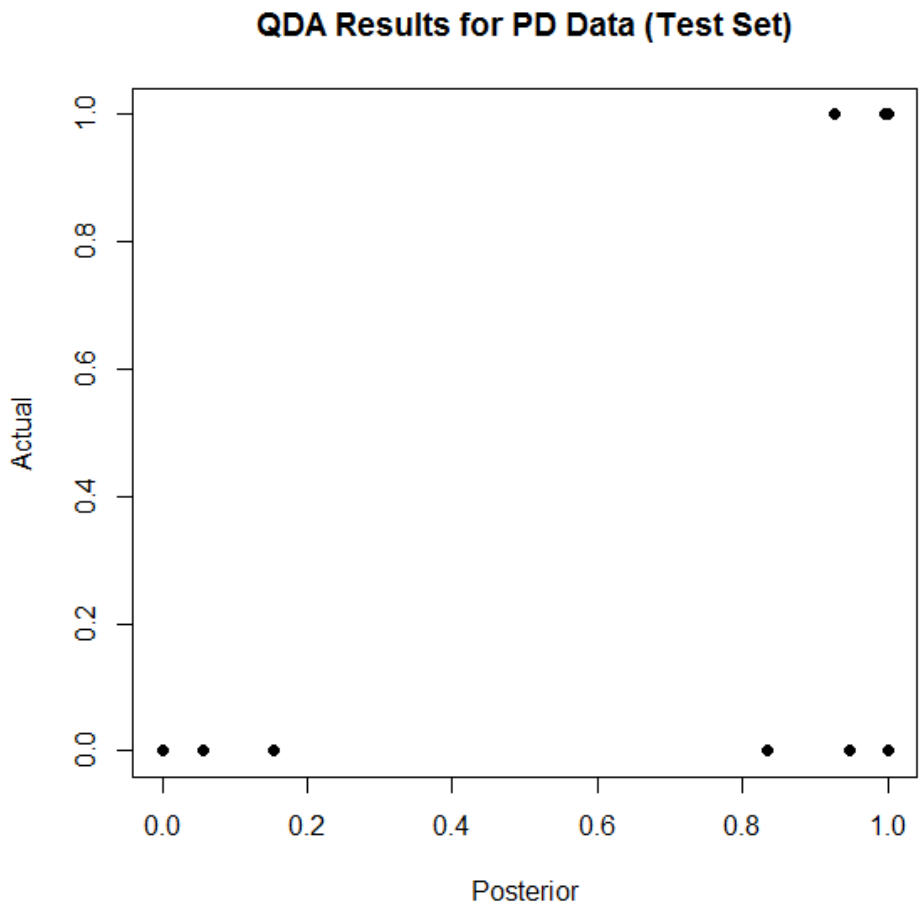


Figure 6.2 The Posterior Probability of Each Observation in the Test Set (QDA)

In summary, the LDA leads to a proportion of correctly predicted observations of 85.94%, with the sensitivity of 91.49% and the specificity of 70.59%. The QDA leads to a proportion of correctly predicted observations of 90.63% with the sensitivity of 100% and the specificity of 64.71%. Figure 6.3 shows the ROC curves of LDA and QDA.

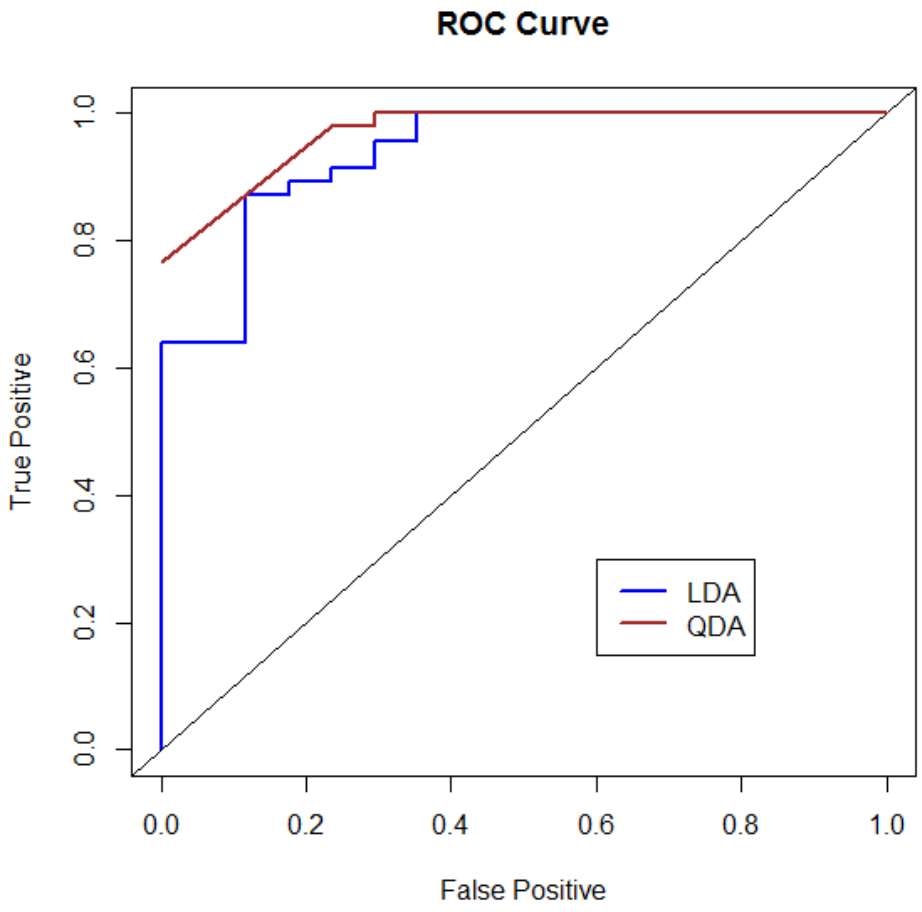


Figure 6.3 The ROC Curves of LDA and QDA

7 CLASSIFICATION TREE AND RANDOM FOREST

7.1 Chapter Overview

This chapter will review and apply Classification Tree and Random Forest to the PD data. The sections of this chapter are organized as follows. (i) Section 7.2 provides the review of the Classification Tree and Random Forest, (ii) Section 7.3 shows the application of the method, and (iii) Section 7.4 evaluates the performance of the models.

7.2 Review of the Method

The classification trees are used to predict the classes of observations or objects from their measurement(s) on explanatory variable(s). For the classification tree, response of each observation is a qualitative rather than a quantitative variable while the explanatory variables can be qualitative or quantitative.

To grow a classification tree, we start with a single node and then look for a binary split which gives the lowest misclassification error $E = 1 - \underset{k}{Max}(\hat{p}_{mk})$, Gini index $G = \sum_{k=1}^{K} \hat{p}_{mk} \times (1 - \hat{p}_{mk})$, or cross-entropy $D = -\sum_{k=1}^{K} \hat{p}_{mk} \log(\hat{p}_{mk})$, where $\hat{p}_{mk}$ represents the proportion of training observations in the m^{th} node that are from the k^{th} class. Then we can take each of the new nodes and repeat this process until it reaches a stopping criterion. Practically when building a classification tree, the Gini index and cross-entropy are preferred to measure the quality of a particular split, as both of them are more sensitive to changes in the node purity than the misclassification rate. However, the resulting tree will often be too large and generate good prediction on the training set, but usually be to over-fit for the data and give a poor prediction on the unseen test set. Thus, the pruning is typically necessary to avoid the over-fit problem by removing some nodes of the tree that provide little power to the classification. The reduced error pruning and cost complexity pruning are the two most popular algorithms. More details about how to prune a tree can be found in many academic

publications (e.g., James et al. 2013, Hastie et al. 2012, etc.).

The bagging (also known as bootstrap aggregating) tree is to fit many large trees (without pruning) by repeatedly resampling the training set data with replacement and predict the class of the observation by majority vote. Generally the bagging tree generates a smoother decision boundary by averaging the results of many trees. The random forest is another popular algorithm from the refinement of the bagging tree. Random forest tries to improve the performance of bagging trees by decorrelating the trees. At each tree split, the bagging tree treats all explanatory variables (predictors) as candidates to split the tree, while in random forest, a random sample of m variables is drawn from predictors, and only those m predictors are considered as candidates for splitting. The reason for doing this is that if one or a few variables are very strong predictors for the response variable, these variables are always selected in many bagging trees, causing these trees to be correlated. However, the random forest overcomes this problem by forcing each split to use only a subset of the predictors. Typically we set $m = \sqrt{p}$ or $m = \log(p)$, where p is the number of explanatory variables.

7.3 Application of the Method

In this section, the classification tree and random forest are constructed based on the training set. The R function - "rpart" in "rpart" package is used to build a classification tree. We use two different metrics to measure the "best" split, which are Gini Impurity and Information Gain (Entropy), and perform classification tree analysis on the training set in order to predict potential PD patients.

Table 7.1 shows the classification rules of Gini tree, and Figure 7.1 shows the change of R^2 and relative error in the full Gini tree due to the increase of split. We observe that split number =4 gives the largest R^2 and the smallest relative error. Figure 7.2 displays the plot of the Gini tree.

Table 7.1 The Classification Rules of Gini Tree

node	split	n	loss	yval	(yprob)		
1)	root	131	31	1	(0.23664122	0.76335878)	
2)	F22<-0.8051341	27	6	0	(0.77777778	0.22222222)	
4)	F16<0.8757867	17	0	0	(1.00000000	0.00000000)	*
5)	F16>=0.8757867	10	4	1	(0.40000000	0.60000000)	*
3)	F22>=-0.8051341	104	10	1	(0.09615385	0.90384615)	
6)	F11<-0.6470057	23	9	1	(0.39130435	0.60869565)	
12)	F1<-0.8756242	9	0	0	(1.00000000	0.00000000)	*
13)	F1>=-0.8756242	14	0	1	(0.00000000	1.00000000)	*
7)	F11>=-0.6470057	81	1	1	(0.01234568	0.98765432)	*

Where * denotes terminal node

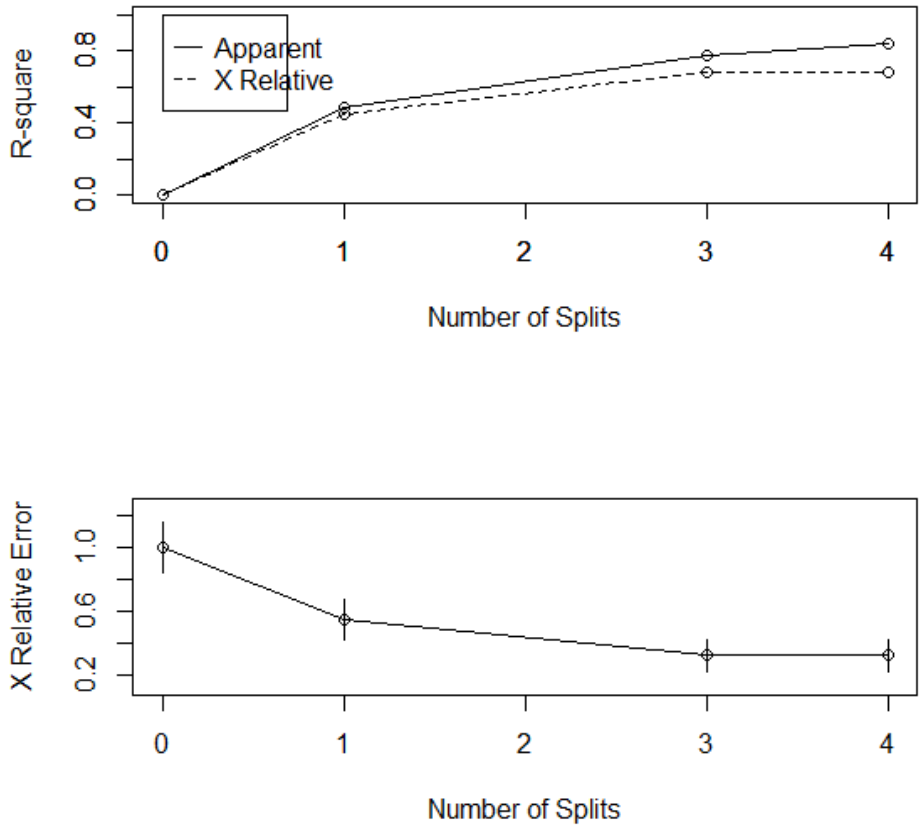


Figure 7.1 Tree Trace Plot of Gini Tree

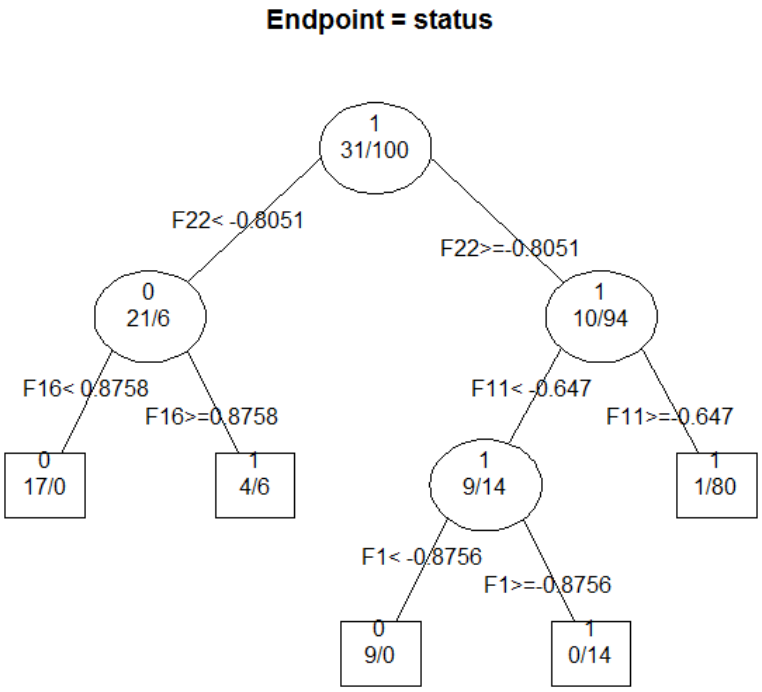


Figure 7.2 The Plot of Gini Tree

Table 7.2 shows the classification rules of the Entropy tree, and Figure 7.3 shows the change of r-square and relative error in the full Entropy tree due to the increase of split. We observe that split number =4 gives the largest r-square and the smallest relative error. Figure 7.4 displays the plot of the Entropy tree.

Table 7.2 The Classification Rules of Entropy Tree

node	**split**	**n**	**loss**	**yval**	**(yprob)**		
1)	root	131	31	1	(0.23664122	0.76335878)	
2)	F22<-0.8051341	27	6	0	(0.77777778	0.22222222)	
4)	F16<0.8757867	17	0	0	(1.00000000	0.00000000)	*
5)	F16>=0.8757867	10	4	1	(0.40000000	0.60000000)	*
3)	F22>=-0.8051341	104	10	1	(0.09615385	0.90384615)	
6)	F10<-0.4733814	29	10	1	(0.34482759	0.65517241)	
12)	F1<-0.8756242	10	1	0	(0.90000000	0.10000000)	*
13)	F1>=-0.8756242	19	1	1	(0.05263158	0.94736842)	*
7)	F10>=-0.4733814	75	0	1	(0.00000000	1.00000000)	*

Where * denotes terminal node

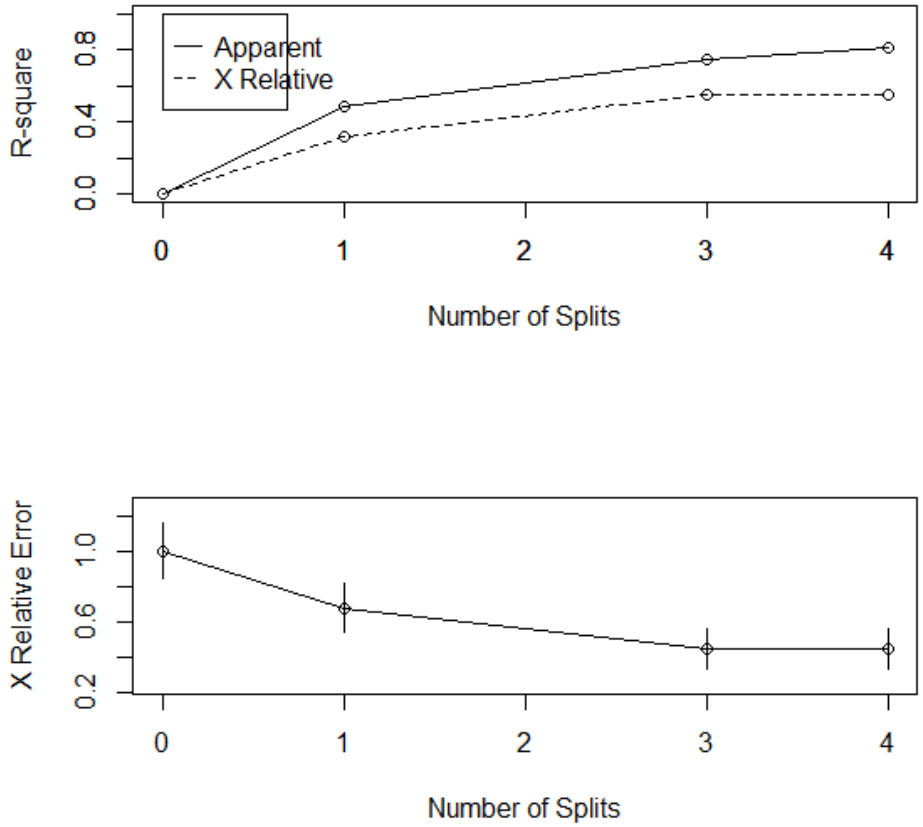


Figure 7.3 The Trace Plot of Entropy Tree

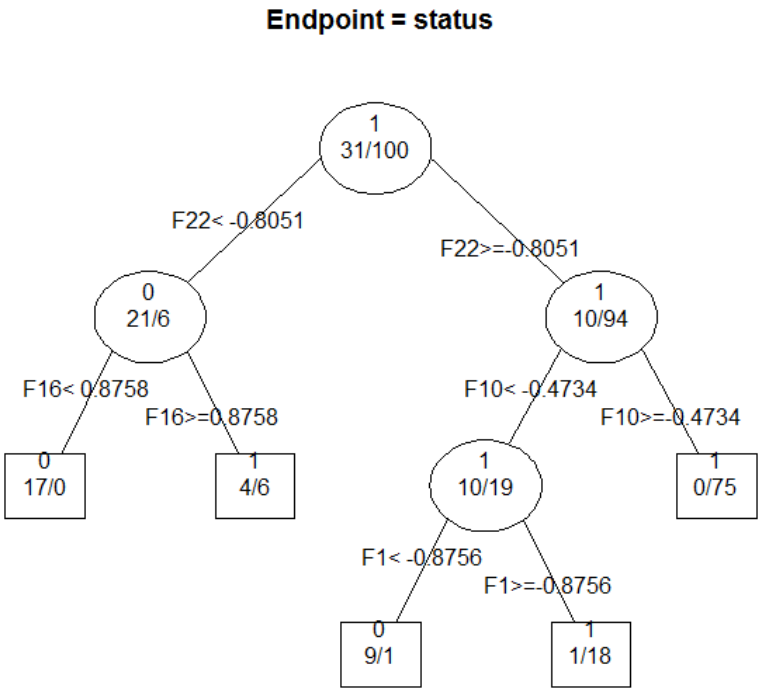


Figure 7.4 The Plot of Entropy Tree

As we have discussed in the previous section, random forests are an ensemble learning algorithm that operate by constructing many decision trees at training time and the predicted result of each observation is the mode of the classes output by individual trees. The R function – "randomForest" in "randomForest" package is used to build random forest. Figure 7.5 shows the prediction error rates when the number of decision trees are increasing and it is noticed that error continues to decrease, but finally it becomes constant as we increase number of classification trees.

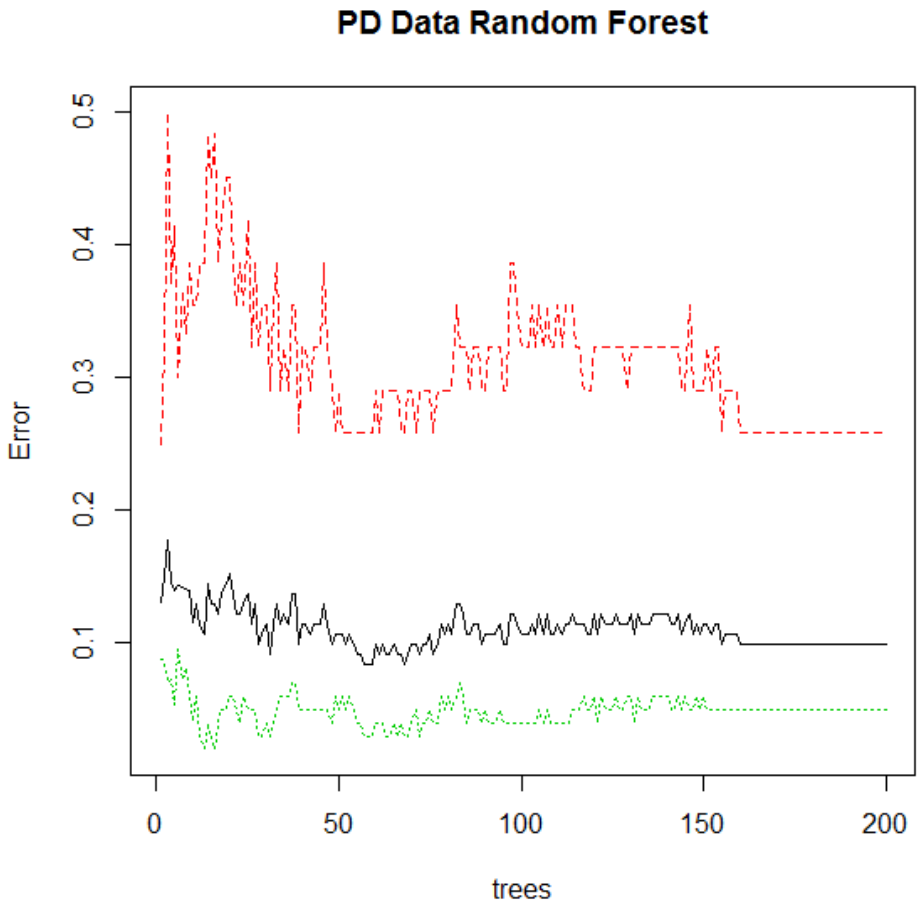


Figure 7.5 The Trace Plot of Random Forest (ntree=200)

Another important application of the random forest is to identify the importance of each feature by the decrease of overall accuracy or Gini Index when we are doing the classification analysis. Table 7.3 shows the importance of each factor and Figure 7.6 shows that in graph (The predictors are ordered top-to-bottom as most-to-least important). We observe that first five most important factors are F22, F19, F1, F18, and F2 by quantifying the decrease of Gini index among all trees.

Table 7.3 The Importance of Each Variable in Random Forest

Var	0	1	MeanDecreaseAccuracy	MeanDecreaseGini
F1	6.6488753	5.0000082	7.258872	2.776957
F2	5.1753964	2.378408	5.399501	2.543734
F3	3.1469658	2.1654111	3.376613	1.689876
F4	-2.3438043	2.6523531	1.356546	0.98642
F5	3.2612629	2.0855413	3.951474	1.185995
F6	3.3405569	2.5949376	3.790752	2.217774
F7	1.6823303	1.612651	2.632906	1.019866
F8	2.3977386	2.3356408	2.839247	1.347622
F9	2.1655812	4.1777227	4.501735	2.061103
F10	1.7472195	4.4371111	4.957006	1.151341
F11	-1.320507	5.1063803	4.53669	1.980359
F12	1.727472	3.9661194	4.645844	2.436768
F13	3.0110325	4.1549737	4.565424	2.183272
F14	0.5133293	5.0482592	5.158688	2.066224
F15	3.9634925	-0.1600499	2.989674	1.528073
F16	1.1994597	3.3966511	3.294367	1.323129
F17	-0.9408771	3.389013	2.307588	1.692024
F18	5.1237611	2.3893952	5.242597	2.56485
F19	6.3906007	7.0522474	7.879941	4.5328
F20	5.0118959	2.2594543	4.339337	2.302739
F21	4.0860417	0.9123728	3.549615	1.930101
F22	7.9656179	7.7836152	9.323918	5.459736

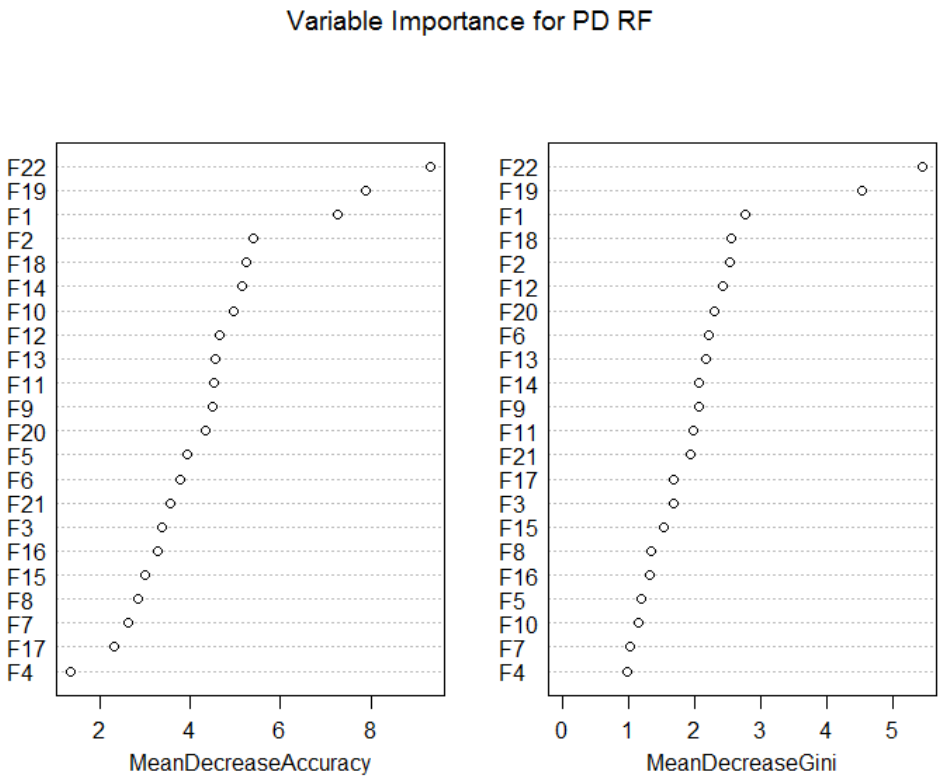


Figure 7.6 The Variable Importance Plot Using RF (ntree=200)

7.4 Model evaluation

The classification tree and random forest fit to the 131 training observations and their performances are quantified by scoring the test set and computing for each patient the predicted probability of PD. Then using the cut-off value of 0.5 for predicted probability of Parkinson's disease, we derive correct classification rate, sensitivity, and specificity in the test set. Tables 7.4, 7.5, and 7.6 show their prediction performances (classification tree - Gini, classification tree - Entropy, and random forest) on the test set (cut-off value of 0.5).

Table 7.4 Actual versus Predicted Parkinson's Disease in the Test Set (Gini Tree)

	Predicted		
Actual	**0**	**1**	**Total**
0	7	10	17
1	3	44	47
Total	10	54	64

Table 7.5 Actual versus Predicted Parkinson's Disease in the Test Set (Entropy Tree)

	Predicted		
Actual	**0**	**1**	**Total**
0	7	10	17
1	3	44	47
Total	10	54	64

Table 7.6 Actual versus Predicted Parkinson's Disease in the Test Set (Random Forest)

	Predicted		
Actual	**0**	**1**	**Total**
0	14	3	17
1	4	43	47
Total	18	46	64

According to the calculation, the classification tree (including Gini and Entropy) has the correct classification rate as 79.69%, the sensitivity as 93.62%, and the specificity as 41.18%. For random forest, we notice that the correct classification rate is 89.06%, the sensitivity is 91.49%, and the specificity is 82.35%. Figures 7.7 shows the ROC curves of classification tree (Gini tree and Entropy tree) and random forest.

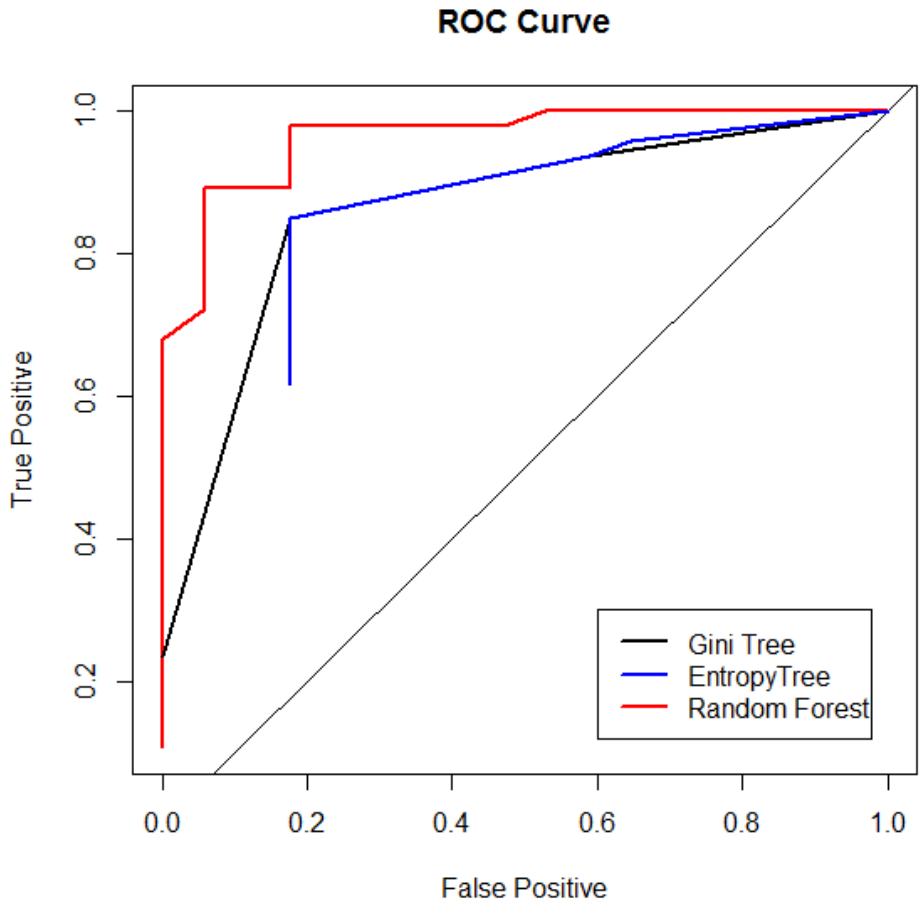


Figure 7.7 The ROC Curves of Classification Tree and Random Forest

8 SUPPORT VECTOR MACHINE

8.1 Chapter Overview

This chapter reviews and applies support vector machine to the PD data. The sections of this chapter are organized as follows. (i) Section 8.2 provides the review of the support vector machine, (ii) Section 8.3 shows the application of the method, and (iii) Section 8.4 evaluates the performance of the model.

8.2 Review of the Method

The support vector machine (SVM) is a non-probabilistic binary linear classifier and can be used for prediction analysis by constructing a hyperplane in a high dimensional space. More specifically, SVM is seeking to construct a $(p-1)$ dimensional hyperplane to separate the p-dimensional data points, where p is number of explanatory variable(s). There could be many candidate hyperplanes classifying the data, and usually we choose the hyperplane with the maximized distance to the nearest data point on each side.

For linear SVM, for any observation i, suppose $\mathbf{x}_i$ is a p-dimensional vector of explanatory variables and $y_i \in \{-1,1\}$ is the binary response. In the training dataset, we look for the maximum-margin hyperplane separating all data points with $y=1$ and $y=-1$. Any separating hyperplane in the p-dimensional space can be written as the set of points $\mathbf{x}$ satisfying $\mathbf{w}\bullet\mathbf{x}=b$, where $\bullet$ is the dot product and $\mathbf{w}$ is the normal vector to the hyperplane. If the data points are linearly separable, we could select two hyperplanes parallel to separating hyperplane that cut through the closest data points on either side (which means that no data points exist between these two hyperplanes) and try to maximize the distance between them. These two hyperplanes, usually called support hyperplanes, can be written as $\mathbf{w}\bullet\mathbf{x}=b+\delta$ and $\mathbf{w}\bullet\mathbf{x}=b-\delta$, which are equivalent to $\mathbf{w}\bullet\mathbf{x}=b+1$ and $\mathbf{w}\bullet\mathbf{x}=b-1$ by

scaling the function. Then, the distance between the these two hyperplanes and the separating hyperplans is $d_+ = \left(\|b+1\| - \|b\|\right)/\|\mathbf{w}\| = 1/\|\mathbf{w}\|$ and $d_+ = \left(\|b-1\| - \|b\|\right)/\|\mathbf{w}\| = 1/\|\mathbf{w}\|$, so the distance between these two support hyperplanes is $2/\|\mathbf{w}\|$. With above definition and formulation, for each data point i, we have constraints $\mathbf{w}\bullet\mathbf{x} - b \le -1 \quad \forall y_i = -1$ and $\mathbf{w}\bullet\mathbf{x} - b \ge +1 \quad \forall y_i = 1$, which can be rewritten as $y_i\left(\mathbf{w}\bullet\mathbf{x} - b\right) - 1 \ge 0 \quad \forall i$. Finally, we can get the optimization problem $\min\left(\|\mathbf{w}\|\right)$ with subject to $y_i\left(\mathbf{w}\bullet\mathbf{x} - b\right) - 1 \ge 0 \quad \forall i$. This optimization problem can be solved by standard quadratic programming techniques after some mathematical formula manipulations. The above linear SVM can still be applied to the non-separable case by introducing the concept of soft margin. Although the SVM is introduced as a linear classifier, the algorithm can be extended to be a non-linear classifier by applying the kernel trick, which aims to gain the linearly separation by mapping the data to a higher dimensional space. Some common kernels include polynomial kernel, radial kernel, sigmod kernel, etc.

8.3 Application of the Method

In this section, the SVM with different kernels are constructed based on the training set. There are two packages in R that can be used to build SVM - “e1071” and “svmpath". In this chapter, R function - “svm” in “e1071” package is employed to construct SVM on the training set in order to predict potential PD patients. The function “svm” give us the opportunity to apply different kernels to SVM and the kernels used in this study include “Linear Kernel”. “Polynomial Kernel” (with degree = 3), “Radial Basis Kernel”, and “Sigmoid Kernel”.

Moreover, cross validation is used in this section for model selection within each kernel. In general, the cross validation aims to estimate the accuracy of the performance of a learning model in practice, to select the appropriate level of flexibility of the model, and to ensure the

validity of the model. The goal of the cross validation is to define a dataset to “test” the model in the training phase to avoid problems like over fitting, so the idea is to split the training set into a set of data points to train with and a set of data points to validate on. The learning model is trained using the new training set and tested on the validation set in order to determine the model flexibility and which particular model to be chosen. We have to be aware that the validation set that was used as a part of training is different from the standalone test set, which is used to estimate how well the learning model works as a whole on unseen data. Since the cross validation is a mean of the resampling method, to reduce the variability, multiple rounds of cross validation are performed using different partition and the validation results are calculated by averaging over all rounds. Two widely used approaches are leave-one-out cross validation and k-fold cross validation, and in this chapter we use 10-fold cross validation to train the model.

8.4 Model evaluation

The model performance is quantified by scoring the test set and computing for each patient the predicted probability of PD. Then using the cut-off value of 0.5 for predicted probability of Parkinson’s disease, we derive the proportion of observations correctly classified as PD patients or correctly classified as non-PD patients by the SVMs (with various kernels), and the achieved sensitivity and specificity in the test set. Tables 8.1, 8.2, 8.3, and 8.4 provide the summary results (cut-off value of 0.5) of applying the models to the test set. Figures 8.1, 8.2, 8.3, and 8.4 display the posterior probability of each observation with Parkinson’s disease.

Table 8.1 Actual versus Predicted Parkinson's Disease in the Test Set (SVM - Linear Kernel)

	Predicted		
Actual	**0**	**1**	**Total**
0	8	9	17
1	0	47	47
Total	8	56	64

Table 8.2 Actual versus Predicted Parkinson's Disease in the Test Set (SVM - Polynomial Kernel)

	Predicted		
Actual	**0**	**1**	**Total**
0	9	8	17
1	1	46	47
Total	10	54	64

Table 8.3 Actual versus Predicted Parkinson's Disease in the Test Set (SVM - Radial Basis Kernel)

	Predicted		
Actual	**0**	**1**	**Total**
0	12	5	17
1	1	46	47
Total	13	51	64

Table 8.4 Actual versus Predicted Parkinson's Disease in the Test Set (SVM - Sigmoid Kernel)

	Predicted		
Actual	**0**	**1**	**Total**
0	0	17	17
1	0	47	47
Total	0	64	64

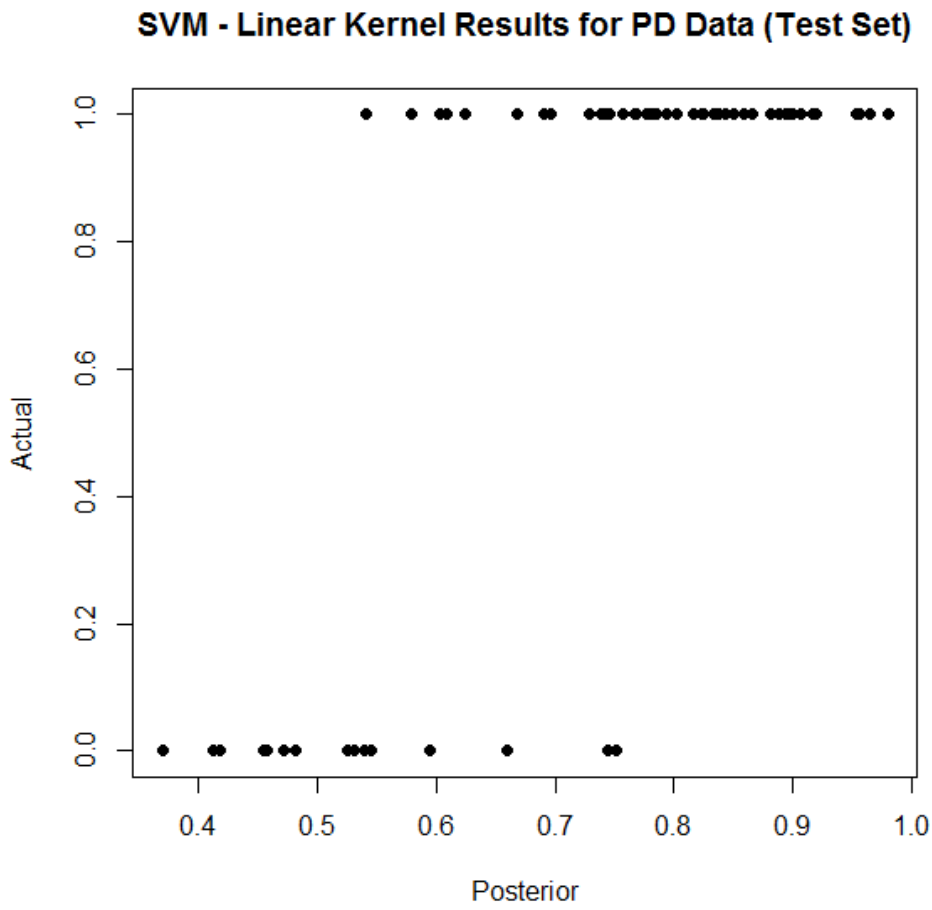


Figure 8.1 The Posterior Probability of Each Observation in the Test Set (SVM - Linear Kernel)

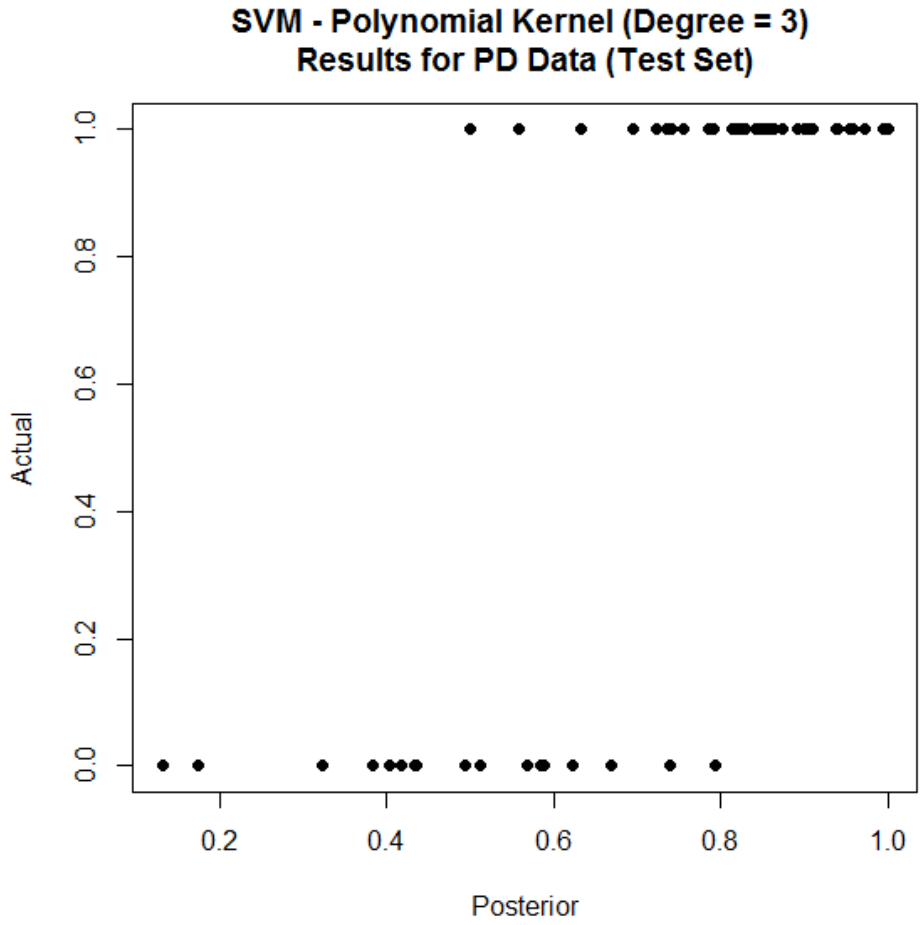


Figure 8.2 The Posterior Probability of Each Observation in the Test Set (SVM - Polynomial Kernel)

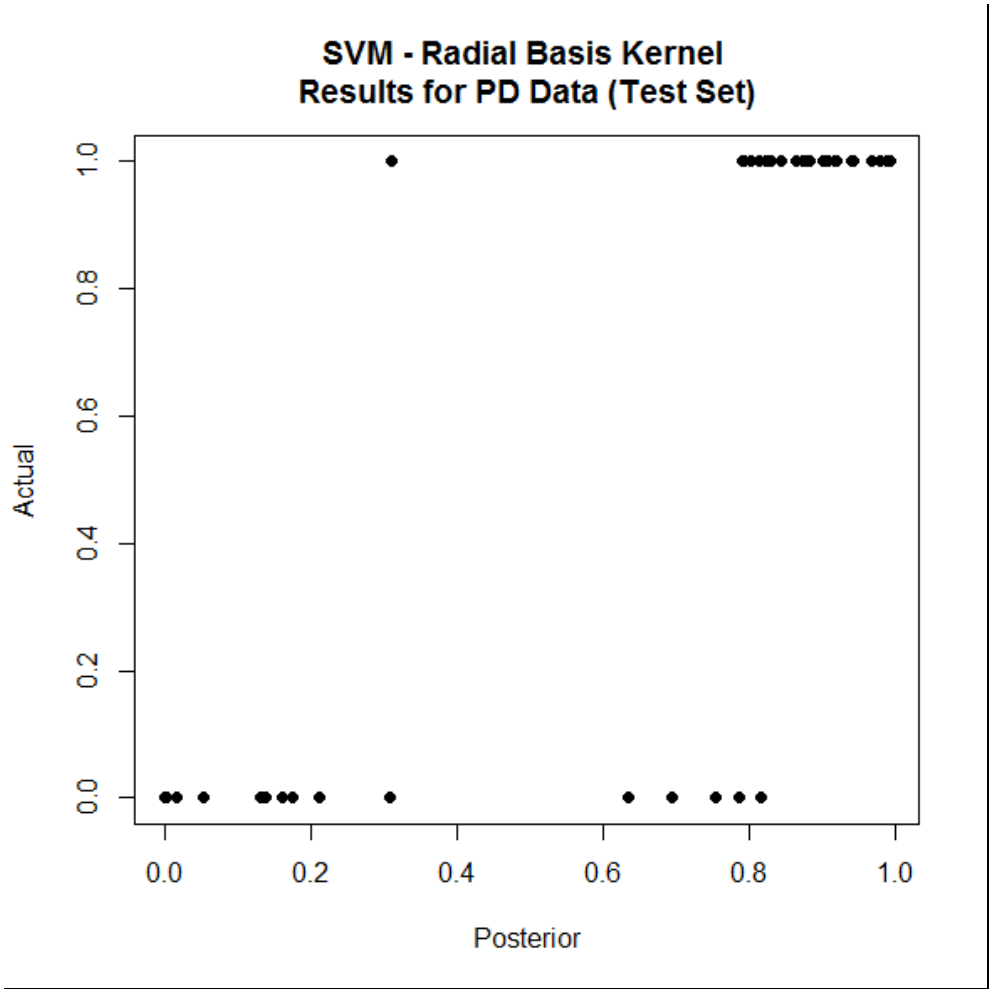


Figure 8.3 The Posterior Probability of Each Observation in the Test Set (SVM - Radial Basis Kernel)

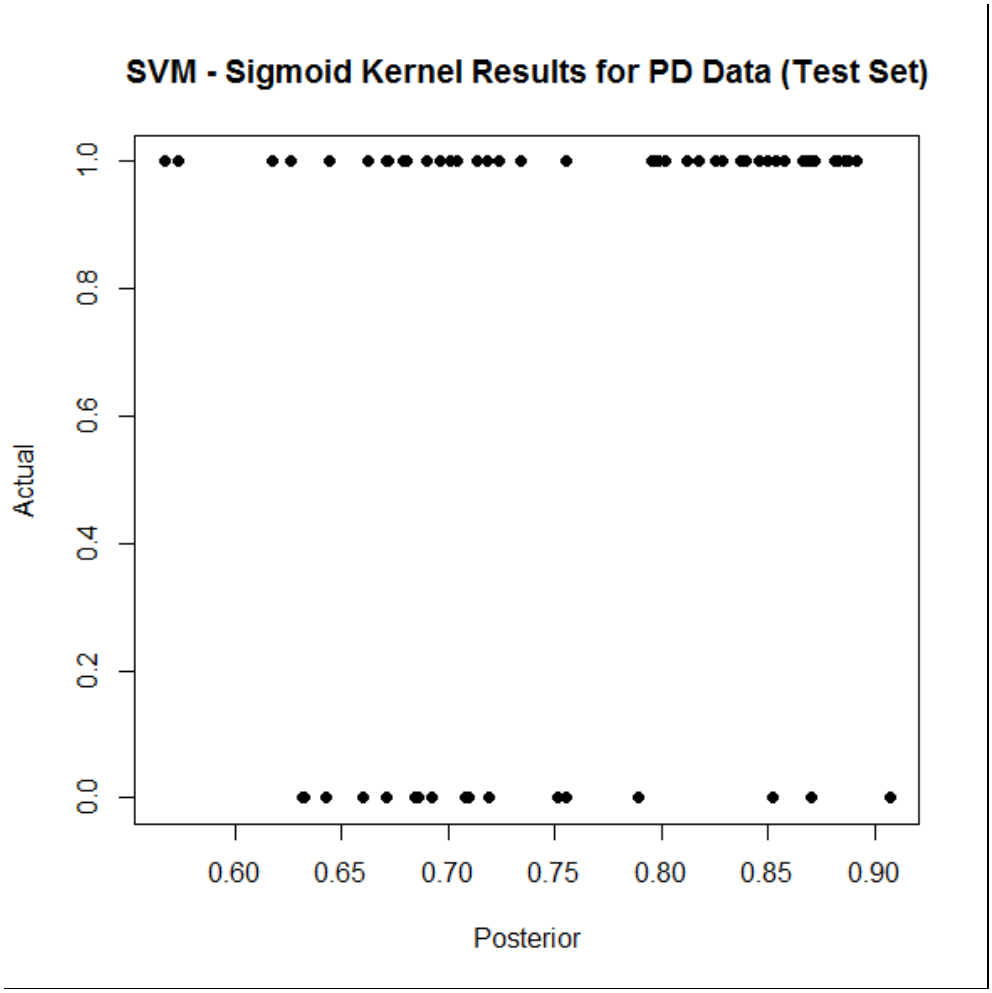


Figure 8.4 The Posterior Probability of Each Observation in the Test Set (SVM - Sigmoid Kernel)

The above confusion matrix shows that SVM with radial basis kernel gives a proportion of correctly predicted observations 90.63% and it is the best classifier among all SVMs, but SVM with Sigmoid kernel only gives a proportion of correctly predicted observations 73.44% and its prediction accuracy is the worst when comparing with other SVMs. Furthermore, according to the calculation, it is obvious that these SVMs with various kernels have different sensitivities and specificities. Figure 8.5 shows the ROC curves of SVMs with different kernels.

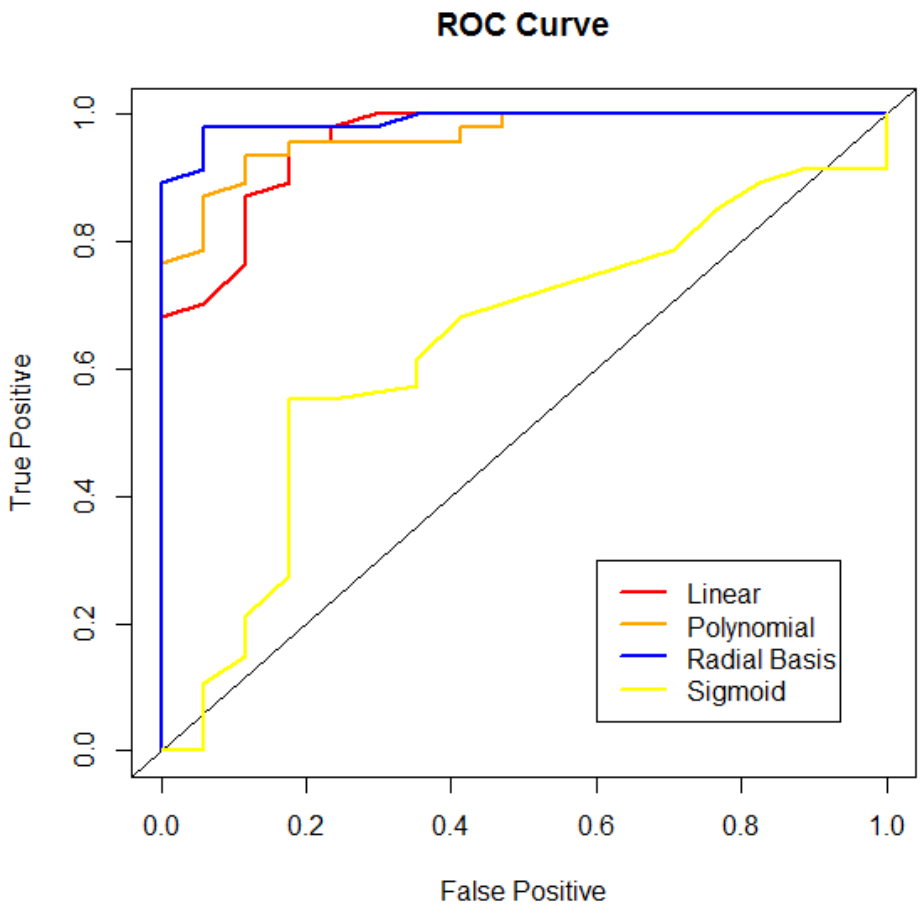


Figure 8.5 The ROC Curves of SVMs

9 ARTIFICIAL NEURAL NETWORK

9.1 Chapter Overview

This chapter reviews and applies Artificial Neural Network (ANN) to the PD data. The sections of this chapter are organized as follows. (i) Section 9.2 provides the review of the Artificial Neural Network, (ii) Section 9.3 shows the application of the method, and (iii) Section 9.4 evaluates the performance of the model.

9.2 Review of the Method

The artificial neural network (ANN) attempts to simulate the biological nervous systems, e.g., the human brain to process information. The network has a large amount of highly interconnected proceeding elements, which are called neurons (or processing elements and units), working in unison to solve specific problems. The network is connected with coefficients (weights), which constitute the neural structure and are organized in layers. The learning process of ANN typically involves adjustments to the synaptic connections between the neurons. In ANN, each neuron has weighted inputs, transfer function, and one output. The transfer function, the weights, and the structure of the network determine the behavior of an artificial neural network. The activation signal, which is constituted by the weighted sum of inputs, passes through the transfer function to produce a single output of neuron. The transfer function can be linear or non-linear. In order to obtain the desired output from the network, the input weights of each neuron need to be adjusted and optimized during the training process until the network reaches the desired prediction accuracy.

As in the above discussion, the learning process of ANN can be treated as a problem of updating the network architecture and connection weights so that the network can handle a particular task efficiently. The two best known ANN are self-organizing ANN and back-propagation ANN. More introductions to the modeling and learning of ANN are

provided by Hastie et al. (2012), Jain and Mao (1996), Dreiseitl and Ohno-Machado (2002), etc.

Last but not least, the ANN has a wide range of applications, including classification, prediction, clustering, etc., which implies that ANN is not just belonging to the domain of supervised learning, but also can be used as an unsupervised learning technique. However, in this study the ANN is purposely designed for supervised learning for potential PD patient prediction.

9.3 Application of the Method

In this section, the ANNs with different number of hidden layers are constructed based on the training set. R function - "nnet" in "nnet" package is employed to construct SVM on the training set in order to predict potential PD patients. When applying the function, we have the opportunity to set up the number of nodes in the hidden layer. We pick 5, 10, and 20 nodes to compare the model performance with various numbers of hidden layers. The input weights of each neuron in ANN are provided in Appendix A.

9.4 Model evaluation

The model performance is quantified by scoring the test set and computing for each patient the predicted probability of PD. Then using the cut-off value of 0.5 for predicted probability of Parkinson's disease, we derive the proportion of observations correctly classified as PD patients or correctly classified as non-PD patients by the ANNs (with various number of nodes in the hidden layer), and the achieved sensitivity and specificity in the test set. Tables 9.1, 9.2, and 9.3 provide the summary results (cut-off value of 0.5) of applying the models to the test set. Figures 9.1, 9.2, and 9.3 display the posterior probability of each observation with Parkinson's disease.

Table 9.1 Actual versus Predicted Parkinson's Disease in the Test Set (ANN with 5 Nodes in the Hidden Layer)

	Predicted		
Actual	**0**	**1**	**Total**
0	16	1	17
1	4	43	47
Total	20	44	64

Table 9.2 Actual versus Predicted Parkinson's Disease in the Test Set (ANN with 10 Nodes in the Hidden Layer)

	Predicted		
Actual	**0**	**1**	**Total**
0	17	0	17
1	2	45	47
Total	19	45	64

Table 9.3 Actual versus Predicted Parkinson's Disease in the Test Set (ANN with 20 Nodes in the Hidden Layer)

	Predicted		
Actual	**0**	**1**	**Total**
0	17	0	17
1	2	45	47
Total	19	45	64

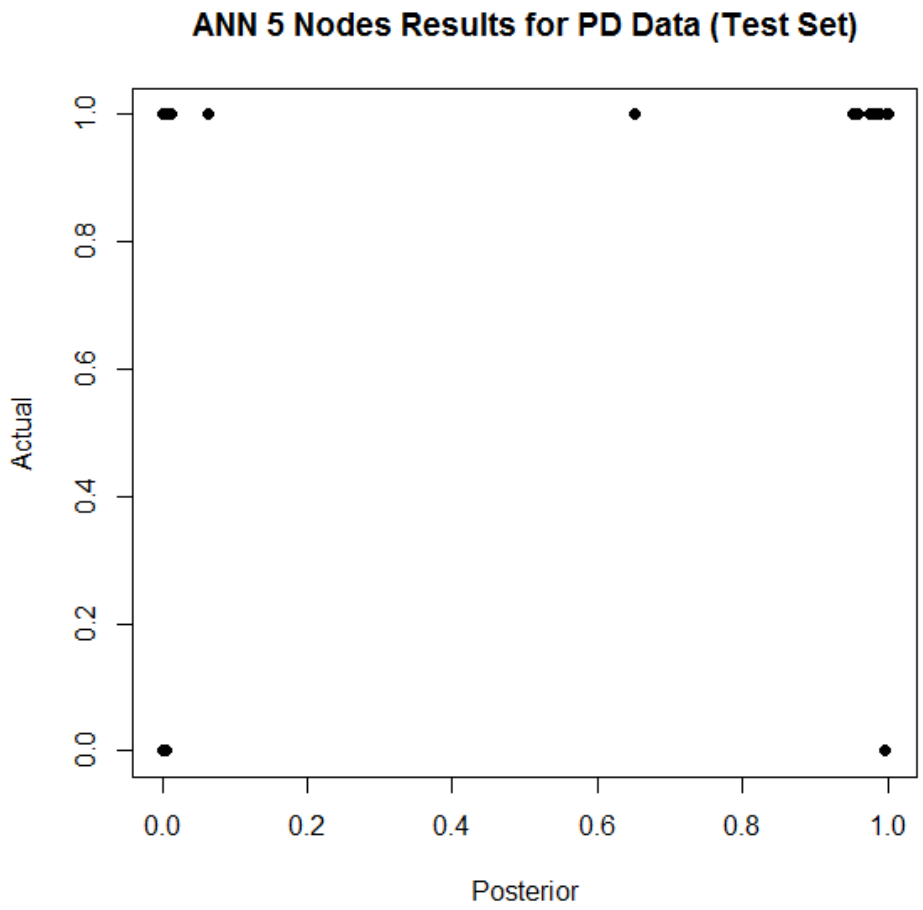


Figure 9.1 The Posterior Probability of Each Observation in the Test Set (ANN with 5 Nodes in the Hidden Layer)

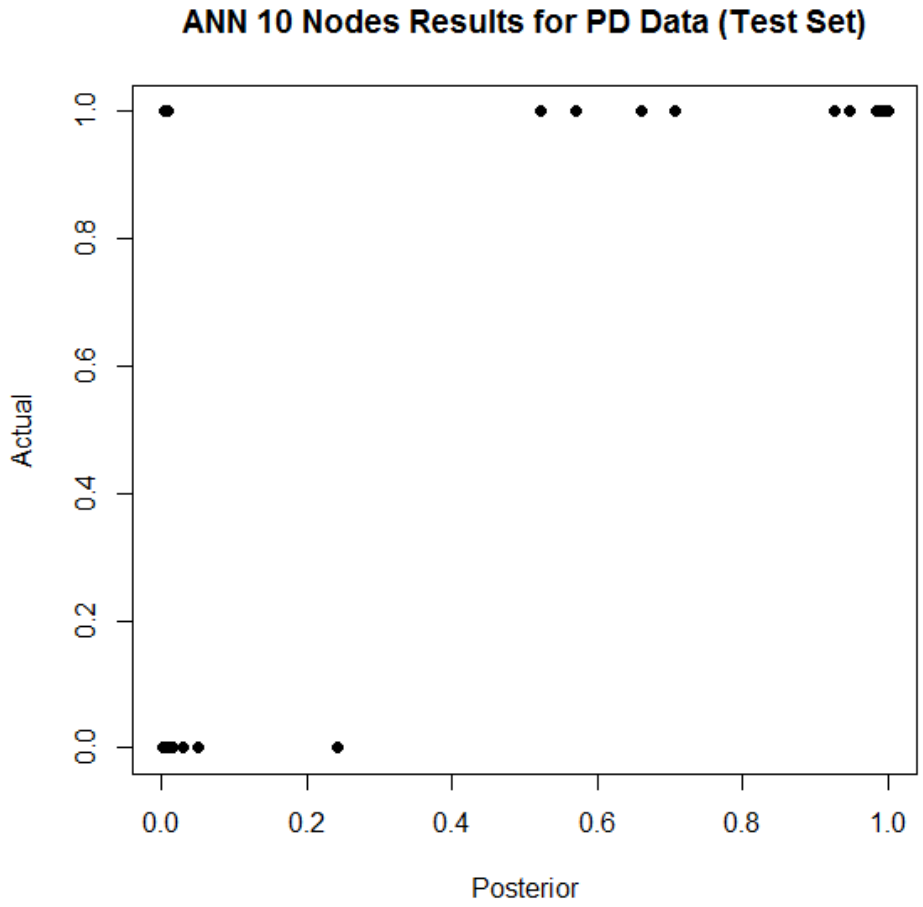


Figure 9.2 The Posterior Probability of Each Observation in the Test Set (ANN with 10 Nodes in the Hidden Layer)

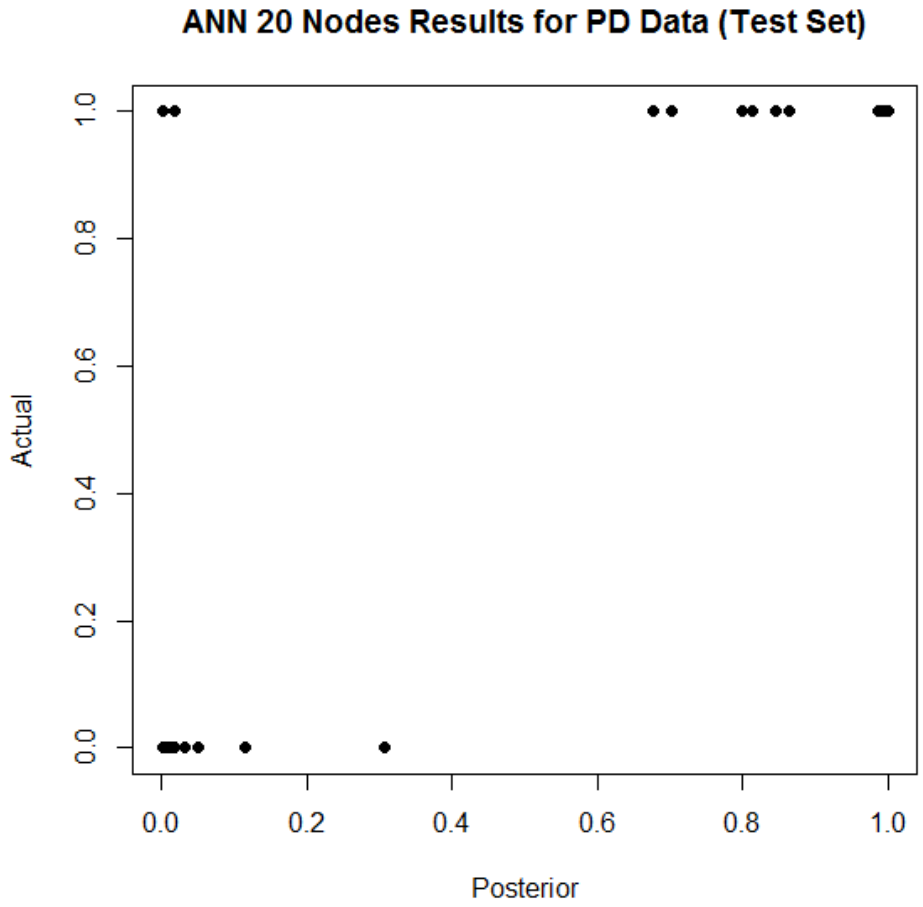


Figure 9.3 The Posterior Probability of Each Observation in the Test Set (ANN with 20 Nodes in the Hidden Layer)

Based on above analysis and calculation, the ANN with 5 nodes can predict the potential PD patients with an accuracy rate as 92.19%. At the same time, the model comes with the sensitivity of 91.49% and the specificity of 94.12%. On the other hand, the ANNs with 10 and 20 nodes in the hidden layer lead to a proportion of correctly predicted observations of 96.88%, and their sensitivities and specificities are 95.74% and 100% respectively. Figure 9.4 shows the ROC curves of ANNs with various number of nodes in the hidden layer.

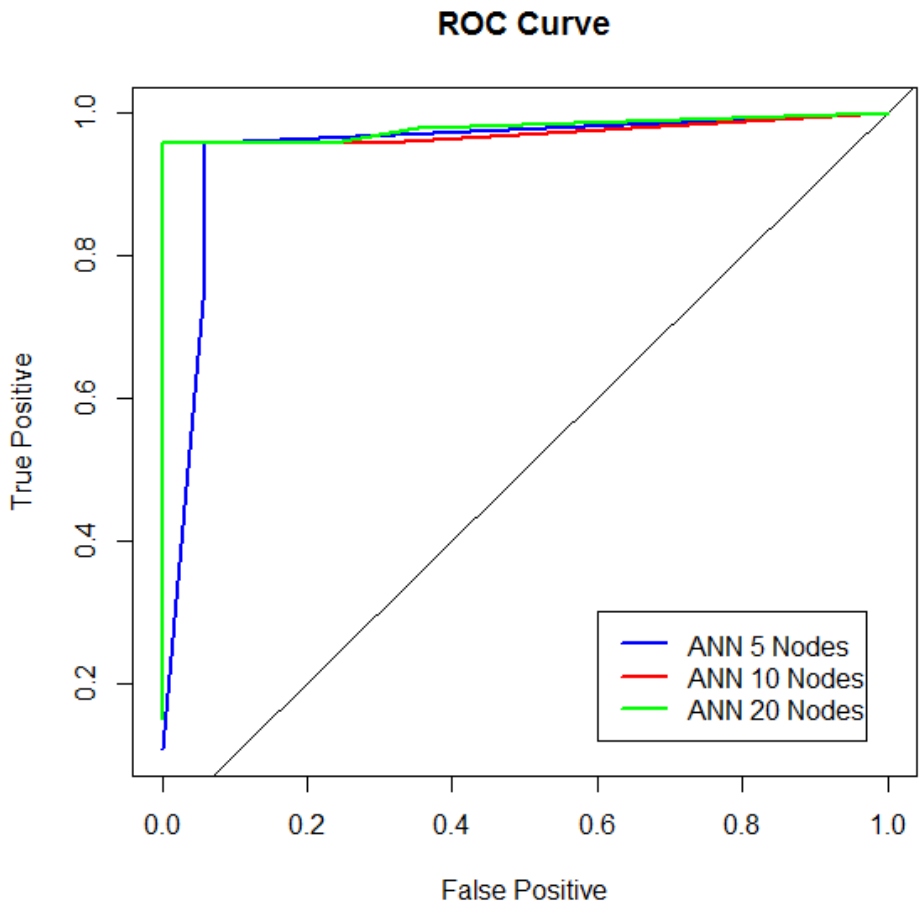


Figure 9.4 The ROC Curves of ANNs

10 DISCUSSIONS AND CONCLUSION

10.1 Chapter Overview

This chapter presents the discussions and conclusion of this study. This chapter is organized as follows. (i) Section 10.2 provides the summary results, (ii) Section 10.3 gives the recommendations, and (iii) Section 10.4 discusses the future work.

10.2 Summary Results

With the preceding analysis and discussion, various data mining methods (supervised learning algorithm) are applied to the training set for model construction. The performance of each model is evaluated against the test set by finding the correct classification rate, sensitivity, and specificity.

Table 10.1 gives the quantification of these three metrics for each model (with cut-off value 0.5). Figure 10.1 shows a comparison of different learning algorithms by overall correct classification rates. It is noted that the best model is ANN (10 or 20 nodes in the hidden layer) with the highest correct classification rate as 96.88%, while the SVM with sigmoid kernel is the worst classifier with the lowest correct classification rates as 73.44%.

However, the above misclassification rates for all models are calculated with a threshold value of 0.5, and in order to have a more comprehensive comparisons, ROC curves can be utilized to compare the performance of each model. As discussed in Section 4.4, a model with perfect discrimination has a ROC curve that passes through the upper left corner (0% false positive, 100% true positive), and the closer the ROC curve is to the upper left corner, the higher the overall accuracy of the model is. By comparing ROC curves for all models, we observe that ROC curves of SVM with radial basis kernel ANN with node=20 most close to the top left corner.

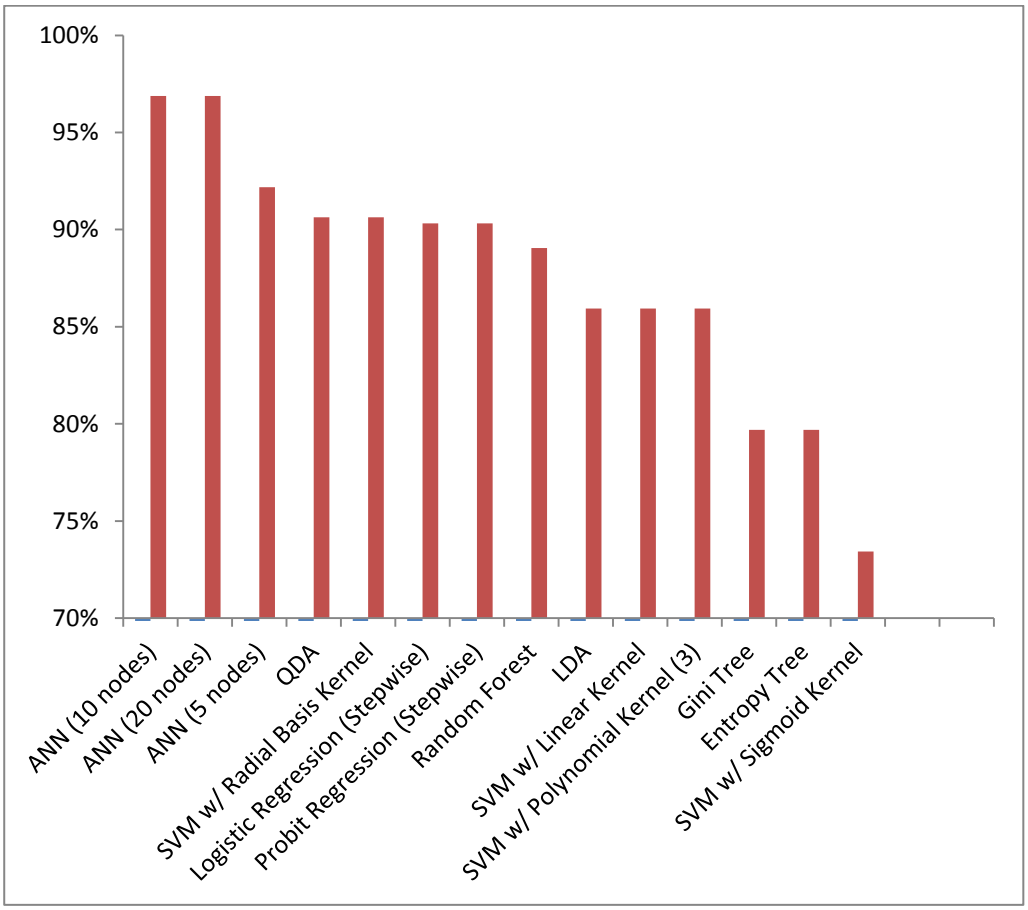


Figure 10.1 Comparison of Learning Algorithms by Overall Correct Classification Rate

Table 10.1 The Performance of Each Data Mining Model

Learning Algorithm	**Sensitivity Rate**	**Specificity Rate**	**Correct Classification Rate**
logistic Regression (Stepwise)	93.62%	82.35%	90.63%
probit Regression (Stepwise)	93.62%	82.35%	90.63%
LDA	91.49%	70.59%	85.94%
QDA	100%	64.71%	90.63%
Gini Tree	93.62%	41.18%	79.69%
Entropy Tree	93.62%	41.18%	79.69%
Random Forest	91.49%	82.35%	89.06%
SVM w/ Linear Kernel	100%	47.06%	85.94%
SVM w/ Polynomial Kernel (3)	97.84%	52.94%	85.94%
SVM w/ Radial Basis Kernel	97.84%	70.59%	90.63%
SVM w/ Sigmoid Kernel	100.00%	0.00%	73.44%
ANN (5 nodes)	91.49%	94.12%	92.19%
ANN (10 nodes)	95.74%	100.00%	96.88%
ANN (20 nodes)	95.74%	100.00%	96.88%

10.3 Recommendations

Several supervised learning methods are applied in this study for a possible suitable model to predict the potential PD patients. Based on the above analysis, we have the following recommendation. ANNs (with 20 nodes or 10 nodes) provide the lowest overall misclassification rates, and if the prediction accuracy is the only and major concern, ANN models are preferred. However, if considering the balance between the prediction accuracy and model interpretation, we recommend using GLMs including the logistic regression and probit regression models, which are able to reveal the most significant factors in the prediction of potential PD patients.

We notice that only 6 out of the 22 features are used to construct final regression model, which has a correct classification rate higher than 90%. The regressions imply that predictors F2, F3, F17, F18, F20, are F21 are the most important variables for the prediction of Parkinson's disease. The models formulized as follows:

Logistic Regression:

$$p = \frac{1}{1 + exp\left[-\begin{pmatrix} 5.65 + 4.63 \times F2 - 2.20 \times F3 + 0.73 \times F17 + 2.69 \times F18 + 5.86 \times F20 \\ -0.08 \times F21 + 5.09 \times F2 \times F18 + 11.10 \times F2 \times F20 - 5.28 \times F2 \times F21 \\ -1.38 \times F3 \times F17 - 2.46 \times F3 \times F18 + 1.59 \times F3 \times F21 \\ +1.51 \times F17 \times F20 + 1.96 \times F18 \times F20 - 2.23 \times F18 \times F21 \end{pmatrix} \right]},$$

where exp is an Exponential function and p denotes the probability of Parkinson's disease.

Probit Regression:

$$p = \Phi\begin{pmatrix} 3.20 + 2.62 \times F2 - 1.16 \times F3 + 0.45 \times F17 + 1.56 \times F18 + 3.32 \times F20 \\ -0.03 \times F21 + 2.88 \times F2 \times F18 + 6.35 \times F2 \times F20 - 2.97 \times F2 \times F21 \\ -0.76 \times F3 \times F17 - 1.38 \times F3 \times F18 + 0.92 \times F3 \times F21 \\ +0.86 \times F17 \times F20 + 1.11 \times F18 \times F20 - 1.26 \times F18 \times F21 \end{pmatrix},$$

where Φ is the Cumulative Distribution Function (CDF) of the standard normal distribution

and p denotes the probability of Parkinson's disease.

The model implementation implies that giving the concerns patients the measure of F2- maximum vocal fundamental frequency, F3 - minimum vocal fundamental frequency, F17 - recurrence period density entropy, F18 - detrended fluctuation analysis, F20 - nonlinear measure of fundamental frequency, and F21 - pitch period entropy. Once these measurements are captured, we can apply the formula above (logistic or probit regression models) to obtain the probability, p , of the patients with Parkinson's disease, then use the cut-off value of 0.5 to classify the patient into the group with Parkinson's disease if p is greater than 0.5. Otherwise we classify the patient into the group without Parkinson's disease.

10.4 Discussions

This section discusses potential concerns with GLM and the estimates of each model's discrimination.

The full logistic regression (with main effects and interactions) model in Chapter 4 includes 6 features, F2, F3, F17, F18, F20, and F22, and all possible two-way interactions among them, and the final recommended model is built on initial model with the variable selection scheme. Above initial features are preliminarily selected by using VIF. However, a reader might believe that if we include all 22 variables and two-way interactions in the full model, it can make the model much more flexible and expect a better regression model. Thus, we will construct a new logistic regression model with all 22 features and interaction terms. The variable selection scheme is applied to remove unnecessary features and interaction terms, and determine the discrimination ability of the new model.

As the original PD dataset has 22 features, if we include all 2-way interactions with original predictors in the initial GLM model, we will have 22+22*21/2=253 regressors. Since we only have 131 observations in the training set, it is impossible to find a single logistic regression model for prediction due to the restriction that the sample size needs to be larger

than the number of predictors (Lim et al, 2009). Thus, this time we proceed by adding a single 2-way interaction with the original 22 variables in the logistic regression model. Then, for each model, we use the stepwise variable selection method to remove all unnecessary predictors. Two hundred and thirty one (231=22*21/2) distinct logistic regression models have to be constructed and examined thoroughly and these models have the following pattern.

Model 1:
$Status \sim F1 + F2 + F3 + F4 + F5 + F6 + F7 + F8 + F9 + F10 + F11 + F12 +$
$F13 + F14 + F15 + F16 + F17 + F18 + F19 + F20 + F21 + F22 + F1 \times F2.$

Model 2:
$Status \sim F1 + F2 + F3 + F4 + F5 + F6 + F7 + F8 + F9 + F10 + F11 + F12 +$
$F13 + F14 + F15 + F16 + F17 + F18 + F19 + F20 + F21 + F22 + F1 \times F3.$

Model 3:
$Status \sim F1 + F2 + F3 + F4 + F5 + F6 + F7 + F8 + F9 + F10 + F11 + F12 +$
$F13 + F14 + F15 + F16 + F17 + F18 + F19 + F20 + F21 + F22 + F1 \times F4.$

……

Model 228:
$Status \sim F1 + F2 + F3 + F4 + F5 + F6 + F7 + F8 + F9 + F10 + F11 + F12 +$
$F13 + F14 + F15 + F16 + F17 + F18 + F19 + F20 + F21 + F22 + F19 \times F22.$

Model 229:
$Status \sim F1 + F2 + F3 + F4 + F5 + F6 + F7 + F8 + F9 + F10 + F11 + F12 +$
$F13 + F14 + F15 + F16 + F17 + F18 + F19 + F20 + F21 + F22 + F20 \times F21.$

Model 230:
$Status \sim F1 + F2 + F3 + F4 + F5 + F6 + F7 + F8 + F9 + F10 + F11 + F12 +$
$F13 + F14 + F15 + F16 + F17 + F18 + F19 + F20 + F21 + F22 + F20 \times F22.$

Model 231:
$Status \sim F1 + F2 + F3 + F4 + F5 + F6 + F7 + F8 + F9 + F10 + F11 + F12 +$
$F13 + F14 + F15 + F16 + F17 + F18 + F19 + F20 + F21 + F22 + F21 \times F22.$

All models above with two-way interactions are treated as full models, and unnecessary terms are removed by performing stepwise model selection, where AIC is used as a main criterion. After comparing all finalized 231 models, the following logistic regression model with the lowest AIC value of 60.71 is nominated as a recommended model for predicting potential Parkinson's disease:

$$Status \sim F7 + F8 + F10 + F14 + F16 + F17 + F19 + F20 + F14 \times F19.$$

Its corresponding full model is:

$$Status \sim F1 + F2 + F3 + F4 + F5 + F6 + F7 + F8 + F9 + F10 + F11 + F12 + F13 + F14 + F15 + F16 + F17 + F18 + F19 + F20 + F21 + F22 + F14 \times F19.$$

Table 10.2 provides the estimation of the final recommended logistic regression model. To evaluate its performance, we apply this logistic regression model with main effects and interaction terms, score the test set, and compute the predicted probability of PD for each patient. Table 10.3 shows the confusion matrix after applying the model to the test set (cut-off value of 0.5), and the accuracy of the above regression model is 92.19%, which is slightly higher than the previous recommended model with accuracy of 90.32% in Chapter 4.

Table 10.2 Final Model Estimates of Logistic Regression

	Estimate	**Std. Error**	**z value**	**Pr(>|z|)**
(Intercept)	11.63	3.20	3.64	0.00
F7	-35.07	10.02	-3.50	0.00
F8	23.57	7.11	3.31	0.00
F10	26.13	8.63	3.03	0.00
F14	-9.28	4.93	-1.88	0.06
F16	-3.75	1.78	-2.11	0.04
F17	-0.85	0.54	-1.58	0.11
F19	20.57	5.85	3.52	0.00
F20	1.56	0.71	2.20	0.03
F14:F19	19.69	5.92	3.32	0.00

Table 10.3 Actual versus Predicted Parkinson's Disease in the Test (Logistic Regression)

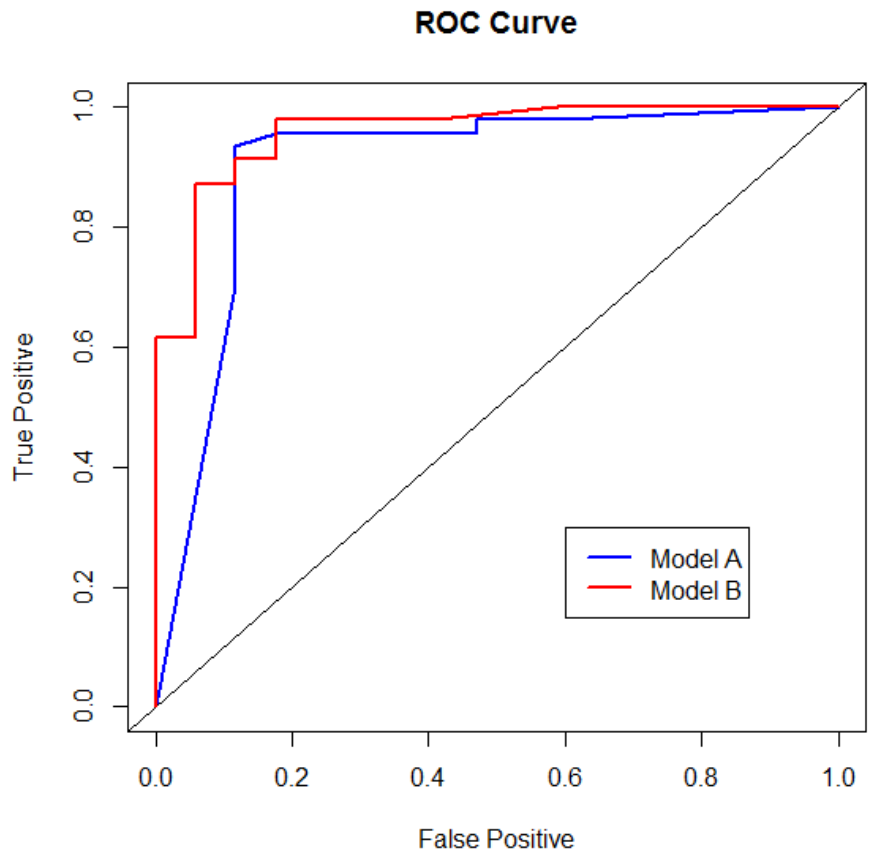
	Predicted		
Actual	**0**	**1**	**Total**
0	15	2	17
1	3	44	47
Total	18	46	64

The recent analysis might suggest using the above recommended model for the Parkinson's disease prediction due to the higher prediction accuracy. However, this prediction accuracy is based on the cut-off value of 0.5. To have a more comprehensive comparison of the above recommended model and the model recommended in Chapter 4, the ROC curves can be utilized. Figure 10.2 displays the ROC curves, where model A (referred the model with the lowest AIC among 231 models) is

$$Status \sim F7 + F8 + F10 + F14 + F16 + F17 + F19 + F20 + F14 \times F19$$

and model B (referred the model obtained by the VIF method) is

$$Status \sim F2 + F3 + F17 + F18 + F20 + F21 + F2 \times F18 + F2 \times F20 + F2 \times F21 + F3 \times F17 + F3 \times F18 + F3 \times F21 + F17 \times F20 + F18 \times F20 + F18 \times F21.$$



.

Figure 10.2 ROC Curves for the Comparison of Recommended Logistic Regression Models

By comparing the area under each ROC curve (AUC) in Figure 10.2, we notice that model B obtained by the VIF method is preferred in general. Thus, from the perspective of discrimination ability, the model recommended in Chapter 4 outperforms the model A mentioned above.

A similar analysis can be conducted for the probit regression discussed in Chapter 5. As in the previously noted analysis on the logistic regression, after performing two hundred and thirty one pribit regressions with stepwise variables selection, we get the best model with lowest AIC below.

$$Status \sim F7 + F8 + F10 + F14 + F16 + F17 + F19 + F20 + F14 \times F19.$$

Table 10.4 presents the estimation of the current recommended probit regression model and Table 10.5 displays the confusion matrix and indicates the model's accuracy is 92.19%. To compare the discrimination ability of the current model and previously recommended model in the Section 10.3, we draw the ROC curves for both models in Figure 10.3 and suggest that Model B is better than Model A in predicting the potential Parkinson's disease patients, as the area under ROC curve of Model B is larger than that of Model A, where model A (which is with the lowest AIC among 231 models) is

$$Status \sim F7 + F8 + F10 + F14 + F16 + F17 + F19 + F20 + F14 \times F19$$

and model B (which is obtained by the VIF method) is

$$\begin{aligned} Status \sim\ & F2 + F3 + F17 + F18 + F20 + F21 \\ & + F2 \times F18 + F2 \times F20 + F2 \times F21 + F3 \times F17 + F3 \times F18 \\ & + F3 \times F21 + F17 \times F20 + F18 \times F20 + F18 \times F21. \end{aligned}$$

Table 10.4 Final Model Estimates of Probit Regression

	Estimate	**Std. Error**	**z value**	**Pr(>|z|)**
(Intercept)	6.48	1.65	3.93	0.00
F7	-19.67	5.23	-3.76	0.00
F8	13.20	3.72	3.55	0.00
F10	15.02	4.61	3.26	0.00
F14	-5.59	2.69	-2.08	0.04
F16	-2.03	0.95	-2.14	0.03
F17	-0.49	0.29	-1.70	0.09
F19	11.38	3.05	3.73	0.00
F20	0.90	0.40	2.24	0.02
F14:F19	10.80	3.12	3.46	0.00

Table 10.5 Actual versus Predicted Parkinson's Disease in the Test (Probit Regression)

	Predicted		
Actual	**0**	**1**	**Total**
0	15	2	17
1	3	44	47
Total	18	46	64

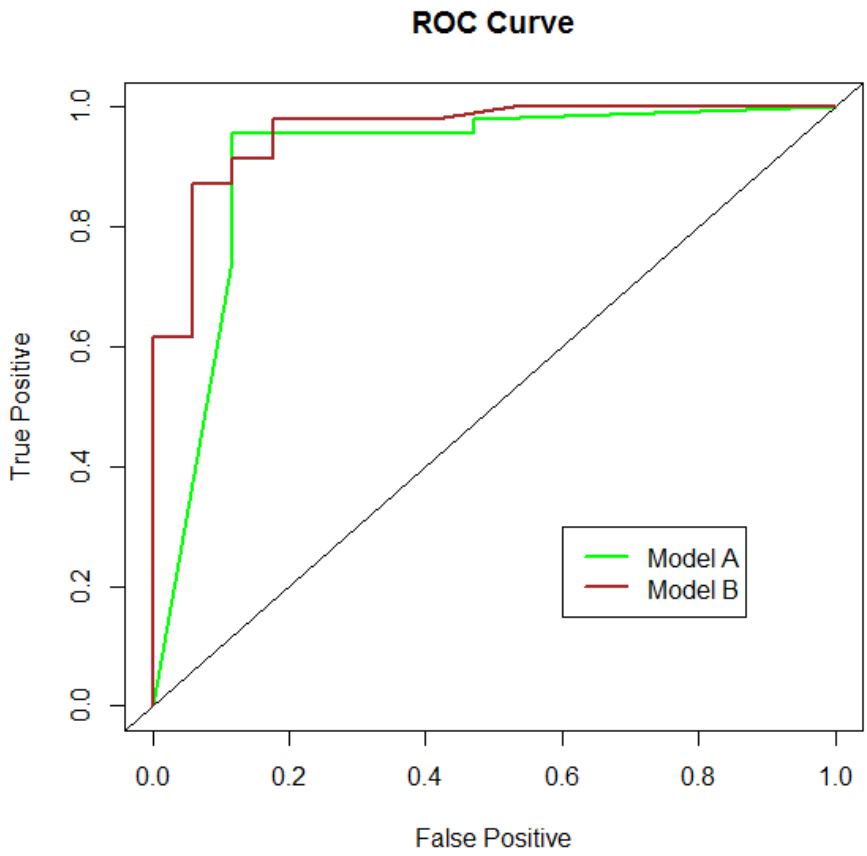


Figure 10.3 ROC Curves for the Comparison of Recommended Probit Regression Models

Another concern is how to provide an unbiased estimate of a model's discrimination, which is a measure of how well the classes in the dates are separated (Dreiseitl and Ohno-Machado, 2002). Throughout previous chapters, we employ a portion of the original data set as a test set that is not used in the model building process to calculate the correct classification rate, sensitivity, and specificity of each model. Table 10.1 displays the performance of each data mining model on the test set. We use these measures to compare and recommend models for potential Parkinson's disease prediction. However we have to keep in mind that all of these estimates are based on one specific split of the original data set into the training and test sets.

Due to the limit of time and space, we do not split the original data into the training and test sets multiple times, and repeat the model building and evaluation process for all predictive models based on each partition. To obtain a comprehensive evaluation and comparison of the above discussed data mining techniques, we can build and evaluate each model many times based on multiple partitions, and then utilize statistical tests to determine whether one model exceeds another one in discrimination ability. In this section, we demonstrate how to carefully assess the logistic regression and the Gini classification tree through multiple splits of original data set into training and test sets. For each split, the training set is used to build and estimate the logistic regression (main effects and interactions) model and the Gini classification tree, and the test set is used to assess models' accuracy, sensitivity, and specificity.

After we randomly split the original data into training and test sets, and build and evaluate both models for 200 times, Tables 10.6 and 10.7 show the statistics (mean and variance) of three evaluation metrics for both models and Figure 10.4 displays boxplots for accuracy, sensitivity, and specificity (the recommended logistic regression model in Chapter 4 and Gini classification tree in Chapter 7). We notice that, for the logistic regression model,

the average of correct classification rate is 85.25% with a variance of 0.17%, and for the Gini tree, the average correct classification rate as 83.39% with a variance of 0.33%. Furthermore, the paired Wilcoxon signed-rank test reveals that logistic regression significantly outperforms the Gini classification tree by comparing the correct classification rates of two models (with *p*-value= 0.00).

The above discussion suggests a better understanding of the prediction power of the logistic regression model and the Gini classification tree on the PD data set, and demonstrates how to compare models more comprehensively through multiple training and test sets splits, as well as the statistical test. The similar comparisons can be extended to all other data mining methods used in this thesis.

Table 10.6 The Mean and Variance of Accuracy, Sensitivity, and Specificity for Logistic Regression

	Accuracy	**Sensitivity**	**Specificity**
Mean	85.25%	90.52%	69.88%
Variance	0.17%	0.23%	1.38%

Table 10.7 The Mean and Variance of Accuracy, Sensitivity, and Specificity for Gini Classification Tree

	Accuracy	**Sensitivity**	**Specificity**
Mean	83.40%	91.24%	60.53%
Variance	0.33%	0.40%	2.96%

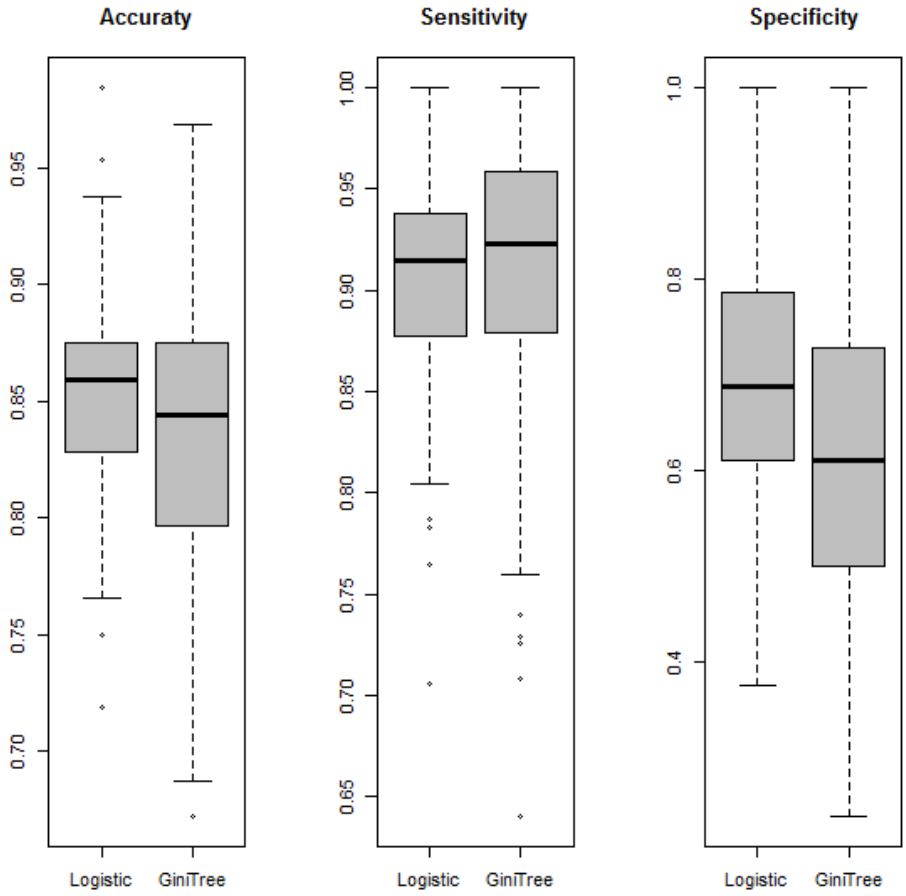


Figure 10.4 Boxplots for Accuracy, Sensitivity, and Specificity for Logistic Regression Model and Gini Classification Tree

10.5 Future Work

In the future applications, one might consider utilizing other supervised learning techniques to predict potential PD patients. For example, ensemble-based classifiers can be used to improve the prediction performance by using multiple learning algorithms (Rokach, 2010). Boosting Algorithm can turn weak learner into a strong learner, and examples of boosting algorithms are “AdaBoost”, “Boosting Tree”, “Gradient Boosting”, etc. The use of these advanced algorithms would improve the accuracy of the predictive model.

Moreover, we may consider applying the dimension reduction techniques to the dataset before conducting the data mining analysis. For instance, the Principle Component Analysis (PCA) is a well-known dimension reduction procedure. The statistical learning algorithms can be applied to the first few principle components for model building after an orthogonal linear transformation, which is more computationally efficient.

Another potential project is to use the cost-sensitive learning and make the study more practical by building the classifiers taking into account of the misclassification costs. The misclassification cost is not considered explicitly in this thesis. However different types of classification errors (false positive and false negative) often incur different costs. In many applications, correct classification of the rare class may have greater value than correct classification of the majority class.

REFERENCES

Agatonovic-Kustrin, S., & Beresford, R. (2000). Basic Concepts of Artificial Neural Network (ANN) Modeling and Its Application in Pharmaceutical Research. *Journal of Pharmaceutical and Biomedical Analysis*, *22*(5), 717–727. doi:10.1016/S0731-7085(99)00272-1.

Beuter, A., & Vasilakos, K. (1995). Tremor: Is Parkinson's disease a dynamical disease? *Chaos: An Interdisciplinary Journal of Nonlinear Science*, *5*(1), 35. doi:10.1063/1.166082.

Dreiseitl, S., & Ohno-Machado, L. (2002). Logistic Regression and Artificial Neural Network Classification Models: A Methodology Review. *Journal of Biomedical Informatics*, *35*(5-6), 352–359. doi:10.1016/S1532-0464(03)00034-0.

Hastie, T. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (2nd ed.). New York, NY: Springer.

Ho, A. K., Iansek, R., Marigliani, C., Bradshaw, J. L., & Gates, S. (1998). Speech Impairment in a Large Sample of Patients with Parkinson's Disease. *Behavioural Neurology*, (11), 131–137.

Jain, A. K., & Mao, J. (1996). Arificial Neural Networks: A Tutorial. IEEE. Retrieved from http://web.iitd.ac.in/~sumeet/Jain.pdf.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (Eds.). (2013). *An Introduction to Statistical Learning: with Applications in R*. New York: Springer.

Lim, N., Ahn, H., Moon, H., & Chen, J. J. (2009). Classification of High-Dimensional Data with Ensemble of Logistic Regression Models. *Journal of Biopharmaceutical Statistics*, *20*(1), 160–171. doi:10.1080/10543400903280639.

Little, M. A., McSharry, P. E., Roberts, S. J., Costello, D. A., & Moroz, I. M. (2007). Exploiting Nonlinear Recurrence and Fractal Scaling Properties for Voice Disorder Detection. *BioMedical Engineering OnLine*, *6*(1), 23. doi:10.1186/1475-925X-6-23.

Marsden, C. D. (1994). Parkinson's Disease. *Journal of Neurology, Neurosurgery & Psychiatry*, *57*(6), 672–681. doi:10.1136/jnnp.57.6.672.

Ramani, R. G., & Sivagami, G. (2011). Parkinson Disease Classification Using Data Mining Algorithms. *International Journal of Computer Applications*, *32*(9).

Rokach, L. (2010). Ensemble-based Classifiers. *Artificial Intelligence Review*, *33*(1-2), 1–39. doi:10.1007/s10462-009-9124-7.

Welling, M. (2014). Support Vector Machines. Retrieved from http://www.ics.uci.edu/~welling/classnotes/papers_class/SVM.pdf.

Wikipedia. (2014). Support Vector Machine. Retrieved from http://en.wikipedia.org/wiki/Support_vector_machine.

APPENDICES

Appendix A: The Input Weights of Each Neuron for ANN

The following part shows the input weights of each neuron for ANN with different number of nodes in the hidden layer.

- 5 nodes in the hidden layer

```
> summary(parkinsons.nn5)
a 22-5-1 network with 121 weights
options were - entropy fitting  decay=0.001
  b->h1  i1->h1  i2->h1  i3->h1  i4->h1  i5->h1  i6->h1  i7->h1  i8->h1  i
9->h1 i10->h1 i11->h1 i12->h1 i13->h1 i14->h1 i15->h1 i16->h1 i17->h1 i18->
h1
  -1.95   -3.63   -0.52    1.17    0.23    2.23   -1.49    2.19   -1.49
 0.62    0.47   -0.17    1.47    2.13   -0.18   -2.29    1.95    0.24    1.
94
i19->h1 i20->h1 i21->h1 i22->h1
   0.11    4.37   -2.48    1.62
  b->h2  i1->h2  i2->h2  i3->h2  i4->h2  i5->h2  i6->h2  i7->h2  i8->h2  i
9->h2 i10->h2 i11->h2 i12->h2 i13->h2 i14->h2 i15->h2 i16->h2 i17->h2 i18->
h2
  -2.92    0.75    0.38    0.52    0.72    0.87   -1.03    0.55   -1.03
 0.00    0.15    0.06    0.07    0.27    0.06    1.06    0.44    1.78    0.
03
i19->h2 i20->h2 i21->h2 i22->h2
  -1.57   -0.60   -0.68   -1.42
  b->h3  i1->h3  i2->h3  i3->h3  i4->h3  i5->h3  i6->h3  i7->h3  i8->h3  i
9->h3 i10->h3 i11->h3 i12->h3 i13->h3 i14->h3 i15->h3 i16->h3 i17->h3 i18->
h3
   0.03   -0.99   -1.20   -0.46    0.04   -0.23    1.16    0.42    1.16
 0.41    0.48    1.03    0.54   -0.50    1.03    0.48   -1.00    1.95    2.
73
i19->h3 i20->h3 i21->h3 i22->h3
   1.95    1.50   -0.43    1.97
  b->h4  i1->h4  i2->h4  i3->h4  i4->h4  i5->h4  i6->h4  i7->h4  i8->h4  i
9->h4 i10->h4 i11->h4 i12->h4 i13->h4 i14->h4 i15->h4 i16->h4 i17->h4 i18->
h4
   3.13   -3.99   -0.89    1.03   -0.34    0.35   -0.62   -3.64   -0.61
 0.39    0.53    0.19   -1.27    0.93    0.19    1.25   -1.87   -4.90   -0.
13
i19->h4 i20->h4 i21->h4 i22->h4
   1.66    5.91   -0.38   -0.29
  b->h5  i1->h5  i2->h5  i3->h5  i4->h5  i5->h5  i6->h5  i7->h5  i8->h5  i
9->h5 i10->h5 i11->h5 i12->h5 i13->h5 i14->h5 i15->h5 i16->h5 i17->h5 i18->
h5
   0.03   -0.99   -1.20   -0.46    0.04   -0.23    1.15    0.42    1.16
 0.41    0.48    1.03    0.54   -0.50    1.03    0.47   -0.99    1.95    2.
73
i19->h5 i20->h5 i21->h5 i22->h5
   1.95    1.50   -0.43    1.97
  b->o  h1->o  h2->o  h3->o  h4->o  h5->o
 -4.87 -13.89  -4.42   7.68  12.82   7.67
```

- 10 nodes in the hidden layer

```
> summary(parkinsons.nn10)
a 22-10-1 network with 241 weights
options were - entropy fitting  decay=0.001
  b->h1  i1->h1  i2->h1  i3->h1  i4->h1  i5->h1  i6->h1  i7->h1  i8->h1  i
9->h1 i10->h1 i11->h1 i12->h1 i13->h1 i14->h1 i15->h1 i16->h1 i17->h1 i18->
h1
  -0.06   -0.47   -0.35   -0.40    0.02    0.11    0.68    0.15    0.68
 0.45    0.42    0.78    0.39   -0.08    0.78    0.27   -0.80    1.10    1.
36
i19->h1 i20->h1 i21->h1 i22->h1
   1.18    0.65   -0.96    1.09
  b->h2  i1->h2  i2->h2  i3->h2  i4->h2  i5->h2  i6->h2  i7->h2  i8->h2  i
9->h2 i10->h2 i11->h2 i12->h2 i13->h2 i14->h2 i15->h2 i16->h2 i17->h2 i18->
h2
   0.63   -1.44   -0.28   -1.29   -0.79   -0.45    0.32   -0.28    0.31
 0.12    0.11   -0.29    0.23    0.68   -0.29   -0.31    1.02    2.14   -2.
86
i19->h2 i20->h2 i21->h2 i22->h2
   1.26   -0.49    0.57    0.99
  b->h3  i1->h3  i2->h3  i3->h3  i4->h3  i5->h3  i6->h3  i7->h3  i8->h3  i
9->h3 i10->h3 i11->h3 i12->h3 i13->h3 i14->h3 i15->h3 i16->h3 i17->h3 i18->
h3
   0.88    5.29    0.23    1.44   -0.22   -1.64    1.42   -0.18    1.40
-1.23   -0.98   -0.95   -1.23   -1.77   -0.95    0.72    0.55   -0.01   -0.
39
i19->h3 i20->h3 i21->h3 i22->h3
   0.82    0.36    1.38    0.02
  b->h4  i1->h4  i2->h4  i3->h4  i4->h4  i5->h4  i6->h4  i7->h4  i8->h4  i
9->h4 i10->h4 i11->h4 i12->h4 i13->h4 i14->h4 i15->h4 i16->h4 i17->h4 i18->
h4
   1.00   -2.26    0.36    0.00   -0.92   -0.38    0.37   -0.95    0.37
 0.24    0.59   -0.02    0.23   -0.01   -0.02    0.69    0.04   -2.76    0.
97
i19->h4 i20->h4 i21->h4 i22->h4
   1.58    0.51   -1.18    1.00
  b->h5  i1->h5  i2->h5  i3->h5  i4->h5  i5->h5  i6->h5  i7->h5  i8->h5  i
9->h5 i10->h5 i11->h5 i12->h5 i13->h5 i14->h5 i15->h5 i16->h5 i17->h5 i18->
h5
  -0.32   -0.39   -0.42   -0.10    0.11    0.31    0.65    0.15    0.65
 0.52    0.48    0.75    0.43    0.08    0.75    0.28   -0.93    0.93    1.
12
i19->h5 i20->h5 i21->h5 i22->h5
   0.95    0.02   -1.16    0.89
  b->h6  i1->h6  i2->h6  i3->h6  i4->h6  i5->h6  i6->h6  i7->h6  i8->h6  i
9->h6 i10->h6 i11->h6 i12->h6 i13->h6 i14->h6 i15->h6 i16->h6 i17->h6 i18->
h6
  -0.62   -2.23    1.55    1.88    0.29    1.60   -1.50    1.30   -1.51
 0.43   -0.02    0.34    0.93    1.39    0.33   -1.40    3.93    1.56    2.
21
i19->h6 i20->h6 i21->h6 i22->h6
   0.57    1.88   -0.89    1.33
  b->h7  i1->h7  i2->h7  i3->h7  i4->h7  i5->h7  i6->h7  i7->h7  i8->h7  i
9->h7 i10->h7 i11->h7 i12->h7 i13->h7 i14->h7 i15->h7 i16->h7 i17->h7 i18->
h7
  -2.35    2.15    0.17   -0.88    0.22   -0.13   -0.03    1.23   -0.03
-0.28   -0.40   -0.66    0.65    0.29   -0.66   -0.24   -0.93    2.27   -2.
11
i19->h7 i20->h7 i21->h7 i22->h7
  -0.52   -2.82    0.81   -0.04
  b->h8  i1->h8  i2->h8  i3->h8  i4->h8  i5->h8  i6->h8  i7->h8  i8->h8  i
9->h8 i10->h8 i11->h8 i12->h8 i13->h8 i14->h8 i15->h8 i16->h8 i17->h8 i18->
h8
  -1.50    0.50   -1.16   -2.96    0.63    0.63    0.09    0.25    0.09
-0.64   -0.69   -0.11   -0.49   -0.90   -0.11   -0.19    0.46    0.13   -1.
14
```

```
i19->h8 i20->h8 i21->h8 i22->h8
  -2.12   -2.61   -0.98   -1.37
  b->h9  i1->h9  i2->h9  i3->h9  i4->h9  i5->h9  i6->h9  i7->h9  i8->h9  i
9->h9 i10->h9 i11->h9 i12->h9 i13->h9 i14->h9 i15->h9 i16->h9 i17->h9 i18->
h9
   1.80   -2.07   -0.54   -0.63   -0.63   -0.22   -0.80   -0.77   -0.80
-0.09   -0.15   -0.30   -0.18    0.17   -0.30   -0.65   -0.54    0.34   -0.
25
i19->h9 i20->h9 i21->h9 i22->h9
   0.85    0.76   -0.49    0.41
  b->h10  i1->h10  i2->h10  i3->h10  i4->h10  i5->h10  i6->h10  i7->h10  i
8->h10  i9->h10 i10->h10 i11->h10 i12->h10 i13->h10 i14->h10 i15->h10 i16->
h10
   -0.31    -0.40    -0.42    -0.12     0.11     0.30     0.65     0.15
  0.65     0.51     0.47     0.75     0.43     0.07     0.75     0.28    -
0.92
i17->h10 i18->h10 i19->h10 i20->h10 i21->h10 i22->h10
    0.94     1.13     0.96     0.06    -1.15     0.90
  b->o  h1->o  h2->o  h3->o  h4->o  h5->o  h6->o  h7->o  h8->o  h9->o h10->
o
 -1.43   4.03   5.04   7.93   5.04   3.84  -8.87  -7.40  -8.20   4.28   3.
84
```

- 20 nodes in the hidden layer

```
> summary(parkinsons.nn20)
a 22-20-1 network with 481 weights
options were - entropy fitting  decay=0.001
  b->h1  i1->h1  i2->h1  i3->h1  i4->h1  i5->h1  i6->h1  i7->h1  i8->h1  i
9->h1 i10->h1 i11->h1 i12->h1 i13->h1 i14->h1 i15->h1 i16->h1 i17->h1 i18->
h1
   1.28   -1.16   -0.92   -0.62   -0.34   -0.07   -0.45   -0.41   -0.45
 0.02    0.04   -0.15   -0.07    0.23   -0.15   -0.25   -0.13    0.18   -0.
24
i19->h1 i20->h1 i21->h1 i22->h1
   0.35    0.66   -0.36    0.19
  b->h2  i1->h2  i2->h2  i3->h2  i4->h2  i5->h2  i6->h2  i7->h2  i8->h2  i
9->h2 i10->h2 i11->h2 i12->h2 i13->h2 i14->h2 i15->h2 i16->h2 i17->h2 i18->
h2
  -0.67   -0.14    0.27   -1.49    0.35    0.47   -0.31    0.25   -0.31
-0.44   -0.36   -0.60   -0.30   -0.09   -0.60   -0.14    0.99   -0.55   -1.
21
i19->h2 i20->h2 i21->h2 i22->h2
  -0.97   -0.15    0.39   -0.55
  b->h3  i1->h3  i2->h3  i3->h3  i4->h3  i5->h3  i6->h3  i7->h3  i8->h3  i
9->h3 i10->h3 i11->h3 i12->h3 i13->h3 i14->h3 i15->h3 i16->h3 i17->h3 i18->
h3
  -0.20   -0.56   -0.68   -0.39    0.11    0.26    0.72    0.07    0.72
 0.47    0.44    0.78    0.32   -0.06    0.78    0.40   -1.04    0.78    1.
32
i19->h3 i20->h3 i21->h3 i22->h3
   0.92    0.49   -1.02    0.80
  b->h4  i1->h4  i2->h4  i3->h4  i4->h4  i5->h4  i6->h4  i7->h4  i8->h4  i
9->h4 i10->h4 i11->h4 i12->h4 i13->h4 i14->h4 i15->h4 i16->h4 i17->h4 i18->
h4
  -1.35   -0.85   -0.20    0.87    0.38    0.61   -0.33    0.27   -0.33
 0.21    0.17    0.41    0.19    0.16    0.40   -0.19    1.14   -0.46    2.
43
i19->h4 i20->h4 i21->h4 i22->h4
  -0.10    1.80   -0.53   -0.01
  b->h5  i1->h5  i2->h5  i3->h5  i4->h5  i5->h5  i6->h5  i7->h5  i8->h5  i
9->h5 i10->h5 i11->h5 i12->h5 i13->h5 i14->h5 i15->h5 i16->h5 i17->h5 i18->
h5
```

```
  -0.94   -0.06   -0.54   -1.41    0.34    0.34    0.00    0.25    0.00
-0.51   -0.51   -0.33   -0.25   -0.53   -0.33   -0.14    0.11    0.03   -0.
64
i19->h5 i20->h5 i21->h5 i22->h5
  -1.14   -1.84   -0.44   -0.79
  b->h6  i1->h6  i2->h6  i3->h6  i4->h6  i5->h6  i6->h6  i7->h6  i8->h6  i
9->h6 i10->h6 i11->h6 i12->h6 i13->h6 i14->h6 i15->h6 i16->h6 i17->h6 i18->
h6
   0.39   -0.01   -0.39   -0.89   -0.21    0.06    0.16   -0.14    0.16
-0.41   -0.40   -0.84   -0.33    0.28   -0.83   -0.41    0.67    3.18   -2.
26
i19->h6 i20->h6 i21->h6 i22->h6
   0.95    0.23    0.34    0.83
  b->h7  i1->h7  i2->h7  i3->h7  i4->h7  i5->h7  i6->h7  i7->h7  i8->h7  i
9->h7 i10->h7 i11->h7 i12->h7 i13->h7 i14->h7 i15->h7 i16->h7 i17->h7 i18->
h7
   1.77   -1.63   -0.18    0.71   -0.18    0.09    0.06   -0.90    0.06
 0.26    0.47    0.38   -0.38    0.00    0.38    0.30    0.44   -1.76    1.
76
i19->h7 i20->h7 i21->h7 i22->h7
   0.04    1.82   -0.76   -0.18
  b->h8  i1->h8  i2->h8  i3->h8  i4->h8  i5->h8  i6->h8  i7->h8  i8->h8  i
9->h8 i10->h8 i11->h8 i12->h8 i13->h8 i14->h8 i15->h8 i16->h8 i17->h8 i18->
h8
   1.29   -1.17   -0.92   -0.62   -0.34   -0.07   -0.46   -0.41   -0.46
 0.02    0.04   -0.15   -0.07    0.23   -0.15   -0.25   -0.14    0.18   -0.
24
i19->h8 i20->h8 i21->h8 i22->h8
   0.35    0.66   -0.36    0.19
  b->h9  i1->h9  i2->h9  i3->h9  i4->h9  i5->h9  i6->h9  i7->h9  i8->h9  i
9->h9 i10->h9 i11->h9 i12->h9 i13->h9 i14->h9 i15->h9 i16->h9 i17->h9 i18->
h9
   1.29   -1.16   -0.92   -0.62   -0.34   -0.07   -0.45   -0.41   -0.45
 0.02    0.04   -0.15   -0.07    0.23   -0.15   -0.25   -0.14    0.18   -0.
24
i19->h9 i20->h9 i21->h9 i22->h9
   0.35    0.66   -0.37    0.19
  b->h10  i1->h10  i2->h10  i3->h10  i4->h10  i5->h10  i6->h10  i7->h10  i
8->h10  i9->h10 i10->h10 i11->h10 i12->h10 i13->h10 i14->h10 i15->h10 i16->
h10
    1.48    -0.74    -0.24    -1.39    -0.02    -0.51     1.25     0.08
  1.26    -0.44    -0.28    -0.25    -0.47    -0.87    -0.25    -0.29    -
0.46
i17->h10 i18->h10 i19->h10 i20->h10 i21->h10 i22->h10
   -3.23     1.33     1.50     0.03    -0.12     1.34
  b->h11  i1->h11  i2->h11  i3->h11  i4->h11  i5->h11  i6->h11  i7->h11  i
8->h11  i9->h11 i10->h11 i11->h11 i12->h11 i13->h11 i14->h11 i15->h11 i16->
h11
    0.43    -2.21    -0.03     0.13    -0.47    -0.07     0.59    -0.52
  0.59     0.13     0.47     0.08    -0.06    -0.13     0.08     0.46
0.79
i17->h11 i18->h11 i19->h11 i20->h11 i21->h11 i22->h11
   -3.40     0.54     1.01     1.06    -1.44     0.93
  b->h12  i1->h12  i2->h12  i3->h12  i4->h12  i5->h12  i6->h12  i7->h12  i
8->h12  i9->h12 i10->h12 i11->h12 i12->h12 i13->h12 i14->h12 i15->h12 i16->
h12
   -1.89    -2.48    -0.52    -0.57    -0.02     0.78    -0.96     0.36
 -0.96     0.15     0.03     0.14     0.32     0.30     0.14    -0.57
0.60
i17->h12 i18->h12 i19->h12 i20->h12 i21->h12 i22->h12
    0.73     0.56     0.37     1.16    -1.14     0.52
  b->h13  i1->h13  i2->h13  i3->h13  i4->h13  i5->h13  i6->h13  i7->h13  i
8->h13  i9->h13 i10->h13 i11->h13 i12->h13 i13->h13 i14->h13 i15->h13 i16->
h13
   -0.84    -1.31    -0.02     0.84     0.20     0.67    -0.93     0.76
 -0.93     0.27     0.11     0.07     0.52     0.89     0.07    -0.75
1.94
i17->h13 i18->h13 i19->h13 i20->h13 i21->h13 i22->h13
    0.78     1.35     0.72     1.98    -1.39     1.21
```

```
  b->h14  i1->h14  i2->h14  i3->h14  i4->h14  i5->h14  i6->h14  i7->h14  i
8->h14  i9->h14 i10->h14 i11->h14 i12->h14 i13->h14 i14->h14 i15->h14 i16->
h14
   -0.56    -0.90    -0.72    -0.87     1.19     0.85     0.18     0.67
  0.18    -0.07    -0.17     0.39    -0.28    -0.43     0.39    -0.23    -
0.52
i17->h14 i18->h14 i19->h14 i20->h14 i21->h14 i22->h14
   -1.71     2.47     0.14     1.95     0.43     0.06
  b->h15  i1->h15  i2->h15  i3->h15  i4->h15  i5->h15  i6->h15  i7->h15  i
8->h15  i9->h15 i10->h15 i11->h15 i12->h15 i13->h15 i14->h15 i15->h15 i16->
h15
    1.00     0.04     0.59     1.55    -0.37    -0.36    -0.02    -0.25
 -0.02     0.52     0.53     0.32     0.26     0.55     0.32     0.15    -
0.15
i17->h15 i18->h15 i19->h15 i20->h15 i21->h15 i22->h15
   -0.03     0.71     1.23     1.91     0.49     0.86
  b->h16  i1->h16  i2->h16  i3->h16  i4->h16  i5->h16  i6->h16  i7->h16  i
8->h16  i9->h16 i10->h16 i11->h16 i12->h16 i13->h16 i14->h16 i15->h16 i16->
h16
   -0.66    -0.13     0.27    -1.48     0.35     0.46    -0.31     0.25
 -0.31    -0.43    -0.36    -0.60    -0.30    -0.09    -0.60    -0.14
0.99
i17->h16 i18->h16 i19->h16 i20->h16 i21->h16 i22->h16
   -0.55    -1.20    -0.97    -0.14     0.39    -0.54
  b->h17  i1->h17  i2->h17  i3->h17  i4->h17  i5->h17  i6->h17  i7->h17  i
8->h17  i9->h17 i10->h17 i11->h17 i12->h17 i13->h17 i14->h17 i15->h17 i16->
h17
   -0.21    -3.45    -0.05    -1.06     0.07     1.13    -0.85     0.27
 -0.85     0.78     0.64     0.73     0.83     0.92     0.73    -0.44    -
0.22
i17->h17 i18->h17 i19->h17 i20->h17 i21->h17 i22->h17
    0.38     0.56    -0.26    -0.31    -1.19     0.10
  b->h18  i1->h18  i2->h18  i3->h18  i4->h18  i5->h18  i6->h18  i7->h18  i
8->h18  i9->h18 i10->h18 i11->h18 i12->h18 i13->h18 i14->h18 i15->h18 i16->
h18
   -1.02     1.00     0.81     0.55     0.31     0.08     0.37     0.37
  0.37    -0.01    -0.02     0.13     0.07    -0.17     0.13     0.21
0.10
i17->h18 i18->h18 i19->h18 i20->h18 i21->h18 i22->h18
   -0.15     0.22    -0.30    -0.52     0.34    -0.17
  b->h19  i1->h19  i2->h19  i3->h19  i4->h19  i5->h19  i6->h19  i7->h19  i
8->h19  i9->h19 i10->h19 i11->h19 i12->h19 i13->h19 i14->h19 i15->h19 i16->
h19
   -0.21    -0.56    -0.68    -0.40     0.11     0.26     0.72     0.07
  0.72     0.47     0.44     0.78     0.32    -0.06     0.78     0.40    -
1.04
i17->h19 i18->h19 i19->h19 i20->h19 i21->h19 i22->h19
    0.78     1.32     0.92     0.49    -1.02     0.80
  b->h20  i1->h20  i2->h20  i3->h20  i4->h20  i5->h20  i6->h20  i7->h20  i
8->h20  i9->h20 i10->h20 i11->h20 i12->h20 i13->h20 i14->h20 i15->h20 i16->
h20
   -0.09     0.14     0.06    -0.43     0.32     0.46    -0.10     0.54
 -0.10    -0.30    -0.35    -0.09     0.00    -0.39    -0.09    -0.22    -
0.31
i17->h20 i18->h20 i19->h20 i20->h20 i21->h20 i22->h20
    0.74     0.97    -0.60    -1.86    -0.51    -0.35
  b->o  h1->o  h2->o  h3->o  h4->o  h5->o  h6->o  h7->o  h8->o  h9->o h10->
o h11->o h12->o h13->o h14->o h15->o h16->o h17->o h18->o h19->o h20->o
  0.14   2.33  -3.30   3.85  -3.50  -3.62   4.95   4.74   2.34   2.33   5.
67   5.17  -4.50  -4.48  -4.19   3.94  -3.28  -4.71  -1.76   3.85  -3.00
```