AN APPLICATION OF BAYESIAN CLASSIFICATION METHODS IN DIABETES DATA

CHENG WEI KUANG

BACHELOR OF SCIENCE (HONS)

STATISTICAL COMPUTING AND

OPERATIONS RESEARCH

FACULTY OF SCIENCE

UNIVERSITI TUNKU ABDUL RAHMAN

OCTOBER 2017

AN APPLICATION OF BAYESIAN CLASSIFICATION METHODS IN DIABETES DATA

By

CHENG WEI KUANG

A project report submitted to the Department of Physical and Mathematical Science
Faculty of Science
Universiti Tunku Abdul Rahman
in partial fulfilment of the requirements for the degree of
Bachelor of Science (Hons) Statistical Computing and Operations Research

October 2017

ABSTRACT

AN APPLICATION OF BAYESIAN CLASSIFICATION METHODS IN DIABETES DATA

Cheng Wei Kuang

Diabetes is one of the leading causes of death that has been causing viral around the world, and undiagnosed diabetes could cause some serious health problems, or even increasing the risk of death. To illustrate the problem, a diabetes dataset with six variables is obtained from R package “locfit”, and Bayesian Classification is applied to the dataset to minimise the number of misclassification of the types of diabetes diagnosed on the patients. Therefore, this study attempts to obtain the most suitable classification technique by comparing three different Bayesian Classification techniques that being used in predicting three types of diabetes. The three techniques included in the study are Naïve Bayes Classifier, Gaussian Mixture Model and Gaussian Process Classifier. The three classification techniques were compared in terms of accuracy, percentage of underestimation and computation time. To plot and illustrate the classification results, R programming language was used in this study. The result of the study shows that Gaussian Mixture Model is the most suitable classification method for the diabetes dataset, as it achieved an accuracy of 91%. Nevertheless, Gaussian Process Classifier and Naïve Bayes Classifier were also effective in producing 90% accuracy and 86% accuracy respectively.

ACKNOWLEDGEMENT

Firstly, I would like to thanks my university supervisor, Mr. Lee Chee Nian for all the guidance throughout my research period. He has been leading me through everything carefully, from deciding a project title to writing a good report. When I faced some problems in the study and I could not solve them by myself, he always tried his best to solve my problems. Sometimes, he assured me that I did a good job when I feel that I did not.

Next, I would like to thanks my family for all the supports they gave me, especially my parents who gave birth to me and made me who I am today. All of them are constantly feeding me with energy and power, especially when I am feeling down. Most importantly, they are one of the sources of my motivation to complete my study and report of the project.

Besides that, I would like to show my gratitude to all my coursemates, including some seniors and juniors, for supporting me throughout my three years of studies in the university. They make me realised that I am not walking down this path alone, as some of them are also struggling on their research at the same time. Also, it is our joy to share the experiences in our own study, and sometimes exchanging some useful information with the others.

Lastly, I would like to say thanks to my university and all the related staffs for making all these possible. Without them, I would not even have the chance to carry out this study and complete this report.

DECLARATION

I hereby declare that the project report is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at UTAR or other institutions.

CHENG WEI KUANG

APPROVAL SHEET

This project report entitled **"AN APPLICATION OF BAYESIAN CLASSIFICATION METHODS IN DIABETES DATA"** was prepared by CHENG WEI KUANG and submitted as partial fulfilment of the requirements for the degree of Bachelor of Science (Hons) Statistical Computing and Operations Research at Universiti Tunku Abdul Rahman.

Approved by:

____

(Mr. Lee Chee Nian) Date:………………..
Supervisor
Department of Physical and Mathematical Science
Faculty of Science
Universiti Tunku Abdul Rahman

FACULTY OF SCIENCE

UNIVERSITI TUNKU ABDUL RAHMAN

Date: ________________

PERMISSION SHEET

It is hereby certified that **CHENG WEI KUANG** (ID No: **14ADB00785**) has completed this final year project entitled "AN APPLICATION OF BAYESIAN CLASSIFICATION METHODS IN DIABETES DATA" under the supervision of Mr. Lee Chee Nian from the Department of Physical and Mathematical Science, Faculty of Science.

I hereby give permission to the University to upload the softcopy of my final year project in pdf format into the UTAR Institutional Repository, which may be made accessible to the UTAR community and public.

Yours truly,

____

(CHENG WEI KUANG)

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

a-MCI	Amnesic Mild Cognitive Impairment
BIC	Bayesian Information Criterion
EM	Expectation-Maximisation
E-step	Expectation-step
GMM	Gaussian Mixture Model
GP	Gaussian Process
GPC	Gaussian Process Classification
GP-LR	Gaussian Process Logistic Regression
GPR	Gaussian Process Regression
k-NN	k-Nearest Neighbour
M-step	Maximisation-step
MLE	Maximum Likelihood Estimation
MBHAC	Model-Based Hierarchical Agglomerative Clustering
NBC	Naïve Bayes Classification
pdf	Probability Density Function
SVM	Support Vector Machine
UCI	University of California, Irvine

CHAPTER 1

INTRODUCTION

1.1 Introduction to Diabetes

Diabetes mellitus, or in short diabetes, is a kind of lifelong disease that occurs when the amount of blood glucose is too high in the body (Diabetes UK, 2017a). Glucose is consumed in order to obtain energy, but the energy can only be obtained by breaking down the glucose molecules. This process can only be done with the aid of insulin. Therefore, when the insulin is insufficient or malfunctioned, the large amount of glucose in the body cannot be broken down. In this case, the level of blood glucose will only keep increasing, causing diabetes. At the same time, diabetes can cause many serious health problems, such as stroke, kidney failure and heart attack, which act as complications that worsen a person's body condition and increasing the risk of death (World Health Organization, 2016).

In year 2014, there were around 422 million of people with diabetes around the world, which was almost four times the number of people with diabetes in 1980 (World Health Organization, 2016). This showed that diabetes was becoming a more serious problem over time. In addition, there were 1.5 million number of deaths caused by diabetes in year 2012, while another 2.2 million number of deaths occurred on people with high blood glucose level (World Health Organization, 2016). In year 2015, diabetes was the sixth leading cause of death around the world (World Health Organization, 2017).

It is found that the proportions of undiagnosed and untreated diabetes in various countries, such as Scotland, Thailand and England, were ranged between 24% and 62% (Gakidou, et al., 2011). It is important for a diabetic patient to be diagnosed as early as possible, as diabetes can worsen the person's health condition if it is left undiagnosed for a long period. The facilities of blood glucose testing should be made available in more places so that people are able to detect and identify whether they are in a risk of getting diabetes (World Health Organization, 2016). Misdiagnosed diabetes is almost as serious as undiagnosed diabetes, as the diagnosed patients are unable to use the right way to take care of their bodies.

With the recent increase in the trend of big data, data mining techniques have been widely employed to various industries. In medical field, classification techniques are often applied to detect the presence of certain diseases by analysing other medical data.

In the context of multivariate analysis, discrimination and classification are two closely related fields, which are responsible in separating a dataset into two or more predefined groups and categorising a new data into one of these groups (Johnson and Wichern, 2007). These techniques are useful in predicting the current stages of the diabetes based on other information of the patients. As the techniques are not perfect, they may result in some misclassifications. Therefore, many researchers are trying to come up with different classification rules for different datasets.

Furthermore, the increase in the popularity of Bayes' Theorem has promoted its application in the field of classification. Some of the methods have utilized the Bayes' Theorem in the algorithms of classification, and these methods are known as Bayesian classification methods or Bayesian classifier. These methods not only focus on assigning a new data into one group, but also compute the probability that the data belongs to each of the groups, which gives more information to the researchers.

It can be often seen that the improvement and development in technologies helped in overcoming some limitations that has troubled the scientists over decades. Therefore, by utilizing the potential of data mining techniques, this study set off to find out a way to reduce the number of undiagnosed and misdiagnosed diabetes patients around the world.

1.2 Classification of Diabetes

There are few ways to classify diabetes. The most popular way is by aetiological types, which classify diabetes according to the causes. Another way would be by the advancing stages of diabetes, which can be shared among all the aetiological types of diabetes.

1.2.1 Aetiological Types of Diabetes

Based on the variety causes of the occurrence, diabetes can be categorised into many different aetiological types. Two of the most common types of diabetes are known as Type 1 diabetes and Type 2 diabetes. Other types of diabetes are considered as very rare as compared to the two stated, and therefore none will be discussed in this session.

Type 1 diabetes occurs when the autoimmune system in the body is activated to destroy the beta cells in the pancreas (American Diabetes Association, 2014), which is responsible in producing insulin to aid in the process of breaking down of glucose. With the destruction of beta cells, the production of insulin will be slowed down or even halted, and this in turns affects the speed of glucose breakdown. This type of diabetes is responsible for around 5-10% of all types of diabetes (American Diabetes Association, 2014).

Type 2 diabetes happens when the insulin resistance of the body cells are high, and therefore they hardly respond to insulin. In normal cases, the body cells should be able to respond to insulin, by absorbing glucose from the bloodstream to be broken down (NIDDK, 2009). When the body cells fail to respond to insulin, less glucose will be absorbed, and therefore the blood glucose level will remain high. This case is also known as relative insulin deficiency, where a higher amount of insulin is required to do a task that could have been done with a small amount of insulin. There are many reasons behind the high insulin resistance, with most cases related to obesity. Type 2 diabetes accounts for roughly 90-95% of diabetes, which is much more common than Type 1 diabetes (American Diabetes Association, 2014).

1.2.2 Stages of Diabetes

Regardless of the cause, diabetes can also be divided into four different stages based on the abnormality of the metabolism of carbohydrate (Fajans, 1973). These four stages are used to determine whether the diabetes is in an early stage or an advanced stage, so that appropriate actions can be taken accordingly.

The earliest stage is known as prediabetes, where the blood glucose level is higher than normal and lower than the blood glucose level of diabetic patients (Diabetes UK, 2017b). Prediabetes is unable to be detected through glucose tolerance test, as it does not show any abnormality (Fajans, 1973). However, those estimated to have prediabetes will have a higher risk of diagnosed with diabetes in future, although this can be avoided by controlling their lifestyles (Diabetes UK, 2017b).

Following prediabetes stage is the subclinical diabetes stage, where the glucose tolerance level is abnormal only when the person is pregnant or under stress, while remaining normal in other conditions (Fajans, 1973).

The next stage is known as chemical diabetes or latent diabetes, where the blood glucose level is abnormal when tested, but no symptom is exhibited (Fajans, 1973). In this case, the existence of diabetes may not be obvious in daily life, but it can be easily detected by running the glucose tolerance test.

The most advanced stage among all the stages is known as overt diabetes or frank diabetes. In this stage, the symptoms of diabetes are obvious, and the blood glucose level is at an extremely high level. In other words, the person with overt diabetes has fasting hyperglycemia (Fajans, 1973).

1.3 Objectives

The objectives of carrying out this study are as follows:

i) To determine the most suitable classification technique that can be used on diabetes data, by studying on three of the Bayesian classifiers.

ii) To evaluate the performance of each technique in terms of number of misclassification, number of underestimations and computation time for each classification process.

iii) To discuss the advantages and limitations of the three Bayesian classifiers.

1.4 Significance of the Study

This study is done in hoping for the number of misclassified diabetes can be reduced in future, so that there will be a lower number of patients and death by diabetes. In addition, this study can also help in increasing the awareness of public towards diabetes, so that cares and attentions can be given to diabetes in earlier stages to prevent it from worsening. Furthermore, this study promotes the use of different stages of diabetes in classification, which could have been used together with the aetiological types of diabetes, but is far less common as compared to them.

1.5 Limitation of the Study

In this study, only three Bayesian classification methods are included due to time constraints, while there are other possible classification methods that could be studied.

Besides that, the dataset used in this study is considered as a secondary data, which is obtained from another research. Therefore, it is not possible to justify the accuracy and the degree of bias of the data collected, as the full information of the data is not available.

1.6 Report Outline

In this Chapter, an introduction to the diabetes has been given, and the objectives of the study have been mentioned. Next, in Chapter 2, some of the other literature works in related fields are discussed. In Chapter 3, the method and tools used in this study are fully explained, along with the overall algorithm of the data analysis and the algorithms of each classification technique. The results of the analysis and study are then discussed in detail in Chapter 4 with the aid of some illustrations. Chapter 5 is used to conclude the whole study.

CHAPTER 2

LITERATURE REVIEW

In this chapter, some of the previous works done by other researchers are discussed to support this study. Section 2.1 explains about the use of classification methods on some medical datasets, while Section 2.2 and Section 2.3 explain the use of some Bayesian classification methods that may perform better than other non-Bayesian methods. Section 2.2 focuses on the medical datasets mentioned in Section 2.1, while Section 2.3 focuses on researches in other fields.

2.1 Classification Methods in Medical Data

As mentioned previously, classification methods are popular in medical data, as they have the ability to detect whether the disease is present in an individual or not. In this case, the dataset used contains a categorical variable that is used to determine the presence or the stage of the disease, while other variables help in distinguishing the characteristics of the classes. The latter is often referred as the independent variables, while the categorical variable is known as the dependent variable.

For instance, Deepthi, Ravikumar and Nair (2016) applied various classification methods on an Arrhythmia dataset, which is obtainable from the Machine Learning Repository of University of California, Irvine (UCI). This dataset contains 452 instances (samples) and 279 different attributes (variables), where one of the attributes represents the classes of arrhythmia, and others give some information regarding the individuals, including ages, heights, weights and other biological information. In this case, the classes are represented by 16 different class codes, where code “01” representing normal individual and code “02” to code “16” representing different classes of arrhythmia (Guvenir, Acar and Muderrisoglu, 1998).

In another study on classification of Alzheimer’s disease, 116 samples were collected, where 27 of the participants were diagnosed with probable Alzheimer’s disease, 50 of them with Amnesic Mild Cognitive Impairment (a-MCI), and the rest were free from both diseases (Challis, et al., 2015). In addition, the ages, genders and numbers of years of formal education of each of the participants were also collected (Challis, et al., 2015). The data of all the participants were then combined into one dataset, with one additional variable indicating the presence of disease in each individual.

For the case of classification of diabetes, one of the most popular dataset used is the Pima Indian Diabetes dataset, which is also obtainable from the UCI Machine Learning Repository. In this dataset, a binary class variable is used as the indicator for the presence of diabetes, where value ‘1’ represents “tested positive for diabetes” and value ‘0’ for the opposite (National Institute of Diabetes and Digestive and Kidney Diseases, 1990).

In many real cases, the data collected might be incomplete due to certain reasons, such as human error, corrupted file and non-respondent in survey.

However, a proper classification algorithm requires a complete set of data without any missing value. Therefore, data cleaning is done to account for the lack of information.

A quick approach can be taken by removing all instances with missing value in any of the variables (Agrawal and Dewangan, 2015), but the drawback would be the reduced number of instances, which is unfavourable for small datasets, as a smaller sample size leads to a lower consistency and accuracy.

To avoid this problem, instead of removing any of the samples, it is suggested that each of the missing values can be replaced by the mean value of all the other available values of the same attribute (Deepthi, Ravikumar and Nair, 2016). In this case, the mean value acted as an estimator to the missing value, which should not be too far from the actual value. This approach not only prevents the sample size from reducing, but also allows the full utilization of all the other values.

To find out whether classification methods are suitable to be applied in medical data, some measures of performances can be used. The most popular measure is the accuracy, which indicates the percentage of samples that are correctly classified. In other words, accuracy can be interpreted as the chance of correctly diagnosing the presence of the disease in a patient.

For the Arrhythmia dataset, an accuracy of 91.11% was achieved by Majority Voting, which ensembled five different classification methods and was applied on many subsets of the whole dataset (Deepthi, Ravikumar and Nair, 2016). For the Alzheimer's disease dataset, Support Vector Machine (SVM) had managed to distinguish the normal individuals from those with a-MCI with an accuracy of 81%, while Gaussian Process Logistic Regression (GP-LR) classifier had reached an accuracy of 97% in distinguishing a-MCI patients from Alzheimer's disease patients (Challis, et al., 2015). For the case of Pima Indian Diabetes dataset, an improved version of J48 Decision Tree algorithm had reached an accuracy of 99.87%, which has almost approaching perfect, considering the large sample size used (Kaur and Chhabra, 2014).

2.2 Bayesian Classification Methods in Medical Data

The main difference between Bayesian classification methods and other non-Bayesian methods is the implementation of Bayes' Theorem, which allows the computation of class membership probabilities in most cases. It was also often argued that to obtain a high accuracy, Bayesian techniques require a relatively small sample size as compared to non-Bayesian techniques.

One of the most commonly used Bayesian classification method is the Naïve Bayes Classification (NBC), which is favoured by many researchers because of the simplicity in computation. A few researchers had attempted to apply NBC in the Pima Indian Diabetes dataset. NBC managed to classify the samples correctly with an accuracy of around 75%, which was not the best, but only a few percents behind most of the other methods (Agrawal and Dewangan, 2015). The only exception was the improved J48 Decision Tree algorithm, which had greatly outperformed all the other methods (Kaur and Chhabra, 2014).

NBC was applied in the Arrhythmia dataset, along with four other non-Bayesian classification methods. In addition, three different ensemble techniques were applied on each of the basic classification methods, and Majority Voting was applied by ensembling all the five basic classifiers (Deepthi, Ravikumar and Nair, 2016). For each of the methods, the classification model was built by using 90% of the samples in dataset, while the other 10% was only used to evaluate the performance of the methods (Deepthi, Ravikumar and Nair, 2016). Among the basic classifiers, NBC had achieved an accuracy of 80%, which was only 4.44% behind the highest accuracy achieved by k-Nearest Neighbour (k-NN) (Deepthi, Ravikumar and Nair, 2016). The ensemble techniques had managed to increase the accuracy of NBC by around 4% to 9%, making it the best or the second best classifier among all the single-ensembled classifiers (Deepthi, Ravikumar and Nair, 2016).

For the Alzheimer's disease dataset, GP-LR model was applied in the binary classification problems, with the aid of Expectation Propagation algorithm in solving the intractable integrals (Challis, et al., 2015). For the performance evaluation, 10 samples from each of the classes were used, while the rest of the samples were used to build the classification models (Challis, et al., 2015). As mentioned previously, the classification problem was divided into two binary classification processes, and it was found that the accuracies obtained by GP-

LR model was almost the same as the one obtained by SVM in both parts of the experiments (Challis, et al., 2015).

2.3 Bayesian Classification Methods in Other Fields

Other than the medical field, some of the Bayesian classification techniques are often applied in many other fields. One of the examples is Gaussian Mixture Model (GMM), which was applied in the classification of images obtained from remote sensing, and the result was compared to those obtained by k-NN and Random Forest (Lagrange, Fauvel and Grizonnet, 2016).

In this experiment, two different datasets were used, namely the Aisa dataset and Potsdam dataset. For Aisa dataset, the image consisted of 361,971 pixels, and each pixel represented one of the 16 types of landscapes, including reed, maize and sunflower (Lagrange, Fauvel and Grizonnet, 2016). For Potsdam dataset, the image was divided into 24 tiles, where each tile contained 6000 × 6000 pixels, and each pixel came from one of the six classes, including building, car and tree (Lagrange, Fauvel and Grizonnet, 2016).

Since the dimensions of the datasets were too high, instead of using the whole dataset for the computation of results, a smaller number of samples from each class were used, and the experiment was carried out for multiple times by using different sample sizes. For each of the experiment, the datasets were divided into two equal sets, where one of the sets was used to formulate the classification rules, while the other set was used for performance evaluation (Lagrange, Fauvel and Grizonnet, 2016). The results were then computed for both of the datasets, by using the three classification methods mentioned above and three additional GMM classifiers with different feature selection techniques applied (Lagrange, Fauvel and Grizonnet, 2016).

For the Aisa dataset, the performance of Random Forest was as good as the performances of GMM with feature selections, while outperforming all the other methods (Lagrange, Fauvel and Grizonnet, 2016). It can be observed that the performance of Random Forest improved as the sample size became larger, while the performance of GMM was consistent in different sample sizes. For the Potsdam dataset, the performance of Random Forest in small sample size was also similar to those of GMM with feature selections, and had outperformed them in a larger sample size (Lagrange, Fauvel and Grizonnet, 2016). These results further supported the argument that Bayesian classification methods can perform well in small sample sizes.

Besides that, Gaussian Process (GP) had been extended into crowdsourcing problem to be used for classification of annotator data, with the aid of Variational Bayes inference to solve the intractable integrals (Besler, et al., 2016). The performance of GP classifier was then compared to other state-of-art crowdsourcing methods, such as Raykar and Rodrigues (Besler, et al., 2016). A dataset with many sentences was obtained, and the goal of the classification problem was to determine whether the sentence has a positive sentiment or a negative sentiment (Besler, et al., 2016). From the dataset, 946 sentences were used to build the classification models, and 5428 sentences were used to evaluate the performance of each classification methods (Besler, et al., 2016). From the results, it can be seen that GP classifier had slightly outperformed all the other methods (Besler, et al., 2016).

CHAPTER 3

METHODOLOGY

The main objective of this project is to determine the most suitable classification technique that can be used on a diabetes dataset. In order to achieve this, each of the steps involved in this study should be analysed in detail, including the algorithm of each of the classification techniques studied. The whole process of obtaining the results for this project will be explained in detail in this chapter.

Section 3.1 introduces about the tools and dataset used in the study. Section 3.2 explicates the overall procedures of dealing with the dataset to obtain the results. Section 3.3 postulates Bayes' Theorem, which is required in each of the following subsections. Section 3.4 introduces about the algorithm of NBC. Section 3.5 refines the algorithm of GMM classification, and Section 3.6 broaches the algorithm for Gaussian Process Classification (GPC).

3.1 Tools and Dataset

The computation and analysis of the results is done with the aid of R programming language, which is also equipped with RStudio as the Integrated Development Environment in this study. This is also used in plotting and illustrating the classification results for the dataset used.

In order to illustrate the case of diabetes patients, a diabetes dataset is obtained from a past research done by Reaven and Miller (1979). This dataset contains 145 samples of non-obese adults, and each of them belongs to one of the three classes, namely “Normal”, “Chemical” and “Overt” (Reaven and Miller, 1979). The “Normal” class implies that the subject is in a normal condition and does not have diabetes. The “Chemical” class indicates that the subject has chemical diabetes, while “Overt” class indicates overt diabetes.

In the original dataset, there are a total of five independent variables and one class variable. The first two variables are the relative weight and the fasting plasma glucose, and were labelled as “rw” and “fpg” respectively in the dataset. The next two variables were labelled as “ga” and “ina”, which represent the area under the curve of plasma glucose curve and plasma insulin curve respectively, after the Oral Glucose Tolerance Test was carried out on a patient for three hours (Reaven and Miller, 1979). The other variable is the steady state plasma glucose, which was labelled as “sspg” in the dataset (Reaven and Miller, 1979).

The dataset mentioned above is obtainable from an R package named as “locfit”, with the dataset named as “chemdiab” in the package (Loader, 2013). In addition, some other R packages are used for the algorithms for each of the classification methods. Package “e1071” is used for the algorithm of NBC (Meyer, et al., 2017), package “mclust” for the algorithm of GMM (Scrucca, et al., 2016), and packages “vbmp” and “kernlab” for the algorithm of GPC (Lama and Girolami, 2016; Karatzoglou, et al., 2004).

3.2 Overall Algorithm

The whole process of analysis of data is divided into two major parts. The first part is focused on the formulation of the classification rule based on the method of classification used. This is also known as model training or model building. The second part is carried out for the purpose of performance evaluation, where the classes of diabetes of a set of sample are predicted based on the model formed in the first part. The result of prediction is then compared with the actual classes of the samples, and the performance is evaluated based on how well the classification methods predict the classes.

For this purpose, the dataset is separated into two different sets, where one of the sets is known as training set, and the other set is known as test set. Training set, as the name suggests, is used in the first part of the analysis process mentioned above, while test set is used for the other part. In this case, the 145 samples in the original dataset are randomly separated into a training set and a test set in a ratio of 6:4.

For the process of separation of the dataset, simple random sampling is used. However, the training set should consist of sufficient samples from all the three classes, so that a good estimation can be done for each of the classes. Therefore, after the sampling is done, the ratio of training set to train set in each of the classes is also computed, and the resulting sets are appropriate as long as these ratios are not too far from 6:4. A scatterplot matrix is also plotted for both sets of samples for a better illustration of the distributions.

After the training set and test set are ready, the process of model training is carried out on the training set of data. After this, the models formed are then applied on the test set of data to predict the classes for each of the samples, by first assuming no information on the classes of these samples. These can be done by using any classification methods. In this study, three different classification methods are selected, namely NBC, GMM classifier and GPC. The algorithms of the model training and prediction for each of the methods are further explained in the following sections.

The results for the prediction obtained from each of the methods are then compared with the actual classes of the samples. As suggested in the literature, the most popular methods for the measure of performance is the accuracy, which represents the percentage of correct classifications. The accuracy is defined as follows:

$$\text{Accuracy} = \frac{\text{Total number of samples that are correctly classified}}{\text{Total number of samples in the test set}}. \quad (3.1)$$

Besides that, a confusion matrix is also included in each set of result obtained. This matrix is a good way to illustrate both the predicted classes and the actual classes in one picture. Table 3.1 shows an example of confusion matrix for a three-class classification problem, where each of the nine alphabets represent a positive integer. From the matrix, the number of samples that are misclassified from each of the classes can be seen clearly.

Table 3.1: Layout for Confusion Matrix.

	Predicted Class		
Original Class	Chemical	Normal	Overt
Chemical	A	B	C
Normal	D	E	F
Overt	G	H	I

To determine the most suitable classification method for this diabetes dataset, the main measure of performance used here is the accuracy. In addition, after the results are obtained from all the three classification methods, the strengths and weaknesses of each method are also analysed, and the most suitable method is chosen after considering all the information obtained from the analysis.

3.3 Bayes' Theorem

Bayes' Theorem may introduce here beforehand the algorithms of the three classification methods, as it is the key concept for each of the Bayesian classification techniques. The general Bayes' Theorem equation (Bolstad, 2007) as follows:

$$f(\theta \mid x_1, x_2, ..., x_k) = \frac{f(x_1, x_2, ..., x_k \mid \theta) f(\theta)}{f(x_1, x_2, ..., x_k)} . \tag{3.2}$$

From the equation, k represents the total number of variables in the dataset, which should be five in the diabetes dataset used in this study. The parameter θ is usually replaced by the class variable, as its probability is the point of interest in a classification problem. If there is p number of possible classes in the dataset, then θ can take p number of discrete values from θ_1 to θ_p. As observable from this equation, there are four different probabilities that form the equation in every variations of Bayes' Theorem.

$f(\theta)$ is known as the prior probability, which represents the prior belief on the class marginal probabilities based on some background information that is already available (Bolstad, 2007). The likelihood is represented by $f(x_1, x_2, ..., x_k \mid \theta)$ in equation (3.2). This probability calculates the likelihood of obtaining the sample values, given that the sample belongs to a particular class (Bolstad, 2007).

The joint probability $f(x_1, x_2, ..., x_k)$ is known as marginal or evidence. It represents the marginal probability of obtaining the sample values, regardless of the classes. By using the law of total probability (Bolstad, 2007), this probability can be obtained by applying the equation below:

$$f(x_1, x_2, ..., x_k) = \sum_{i=1}^{p} f(x_1, x_2, ..., x_k \mid \theta_i) f(\theta_i). \tag{3.3}$$

As the marginal probability of the sample involves both the prior probabilities and the likelihoods, the computation of marginal probability is usually the last step before obtaining the posterior probability.

The posterior probability is the probability that a sample belongs to a particular distribution with parameter θ, after considering the values of each variable in the sample. The posterior probability is represented by $f(\theta \mid x_1, x_2, ..., x_k)$, and is the key used to predict the class probabilities of a sample in a classification problem (Bolstad, 2007).

3.4 Naïve Bayes Classification

As one of the most popular classification method, NBC focuses on classifying a dataset with a simple algorithm. This is achieved by assuming that all the independent variables in the dataset are also independent with each other. By assuming this, the values of all the covariance and correlation between each variable are equal to zero, which makes the computation easier.

As a Bayesian classification method, Bayes' Theorem is used as the key concept of NBC, which is represented by the equation below:

$$f(C \mid x_1, x_2, ..., x_5) = \frac{f(x_1, x_2, ..., x_5 \mid C) f(C)}{f(x_1, x_2, ..., x_5)} . \qquad (3.4)$$

In this equation, the variable C represents the class variable, which takes discrete values C_1, C_2 and C_3 in this case of diabetes data. As a step of standardisation for the whole report, C_1 is used to represent the "Chemical" class, while C_2 and C_3 are used to represent "Normal" class and "Overt" class respectively. Similarly, the five variables in the dataset, which are "rw", "fpg", "ga", "ina" and "sspg", are represented by x_1, x_2, x_3, x_4 and x_5 respectively.

By adding the independent assumption into the problem, the likelihood function can be simplified by using the multiplicative rule. This leads to a new likelihood function as follows:

$$f(x_1, x_2, ..., x_5 \mid C) = \prod_{i=1}^{5} f(x_i \mid C). \tag{3.5}$$

Given a particular class, the likelihood function of a variable is formed by using all samples in the training set that belongs to that class. Since all the variables involved in the diabetes dataset (except the class variable) is continuous, the likelihood function should be represented by a probability density function (pdf) of a continuous distribution. In this study, Gaussian distribution (also known as Normal distribution) is chosen to carry out this task, as it is often the resulting distribution from a large random sample, according to the Central Limit Theorem. Besides that, the two main parameters for Gaussian distribution are the mean and variance, which makes the parameter estimation much easier than most of the other distributions. For this purpose, the training set is further separated according to their classes, and the sample mean, $\bar{x}$ and sample variance, s^2 for each variable in each class is computed.

Then, the likelihood function for each variable in each class can be represented by the conditional pdf below:

$$f(x_{ij} \mid C_j) = \frac{1}{s_{ij}\sqrt{2\pi}} \times \exp\left[\frac{-(x_{ij} - \bar{x}_{ij})^2}{2s_{ij}^2}\right]. \tag{3.6}$$

In this classification problem, the prior probability is obtained from the training set of data, as the training set represents the information on hand. The proportion of each class in the training set is used as the prior probability of that class.

It can be found that the marginal probability of a sample, $f(x_1, x_2, ..., x_k)$ is constant when computing the posterior probabilities for all the possible classes. Therefore, this probability is often omitted from the calculation and replaced by a constant value *d*. By combining equation (3.4) and equation (3.5), the posterior probability can be further simplified as follows:

$$f(C \mid x_1, x_2, ..., x_5) = d \times f(C)\prod_{i=1}^{5} f(x_i \mid C). \tag{3.7}$$

To classify a sample into one of the three classes, the posterior probabilities for all the three classes are computed by using the values of the five variables from the sample. Then, the sample is classified into the class with the highest posterior probability.

3.5 Gaussian Mixture Model Classification

GMM is a distribution formed by combining a number of Gaussian distributions into one distribution. By applying the concepts of GMM into classification, its algorithm is very similar to the algorithm of the NBC. The only difference is in terms of the likelihood function, where a GMM is used for each class in this case. The algorithms in obtaining all the other probabilities and classifying the test set samples are the same as in Section 3.4. Therefore, this section focuses on the algorithm in forming the GMM for the likelihood function.

Since the variables are not assumed to be independent with each other, it is more convenient to use the multivariate version of Gaussian distribution instead of the univariate version. The pdf of a GMM (Scrucca, et al., 2016) as follows:

$$f(\mathbf{x}) = \sum_{g=1}^{G} \pi_g f_N(\mathbf{x}; \boldsymbol{\mu}_\mathbf{g}, \boldsymbol{\Sigma}_\mathbf{g}) \,. \tag{3.8}$$

In equation (3.8), the variable G represents the total number of components in the model, or in other words, the number of multivariate Gaussian distribution functions. The $f_N(\mathbf{x})$ in the equation represents the pdf of a Gaussian distribution, with a mean vector $\boldsymbol{\mu}_\mathbf{g}$ and a covariance matrix $\boldsymbol{\Sigma}_\mathbf{g}$ as the parameters. Each of the components comes with a variable π_g, which represents the proportion of the Gaussian distribution in the GMM, and is known as mixing weight (Scrucca, et al., 2016). If the value of G is known, then the number of parameters to be estimated is equals to three times *G*, as each of the components has three parameters to be estimated, which are π_g, $\boldsymbol{\mu}_\mathbf{g}$ and $\boldsymbol{\Sigma}_\mathbf{g}$.

Some of the special models of GMM are introduced in Section 3.5.1. To obtain the estimated values of *G* and all the parameters, the first step involved is known as Model-Based Hierarchical Agglomerative Clustering (MBHAC), which is further discussed in Section 3.5.2. Then, in Section 3.5.3, the best model is selected, along with a value for *G*. The algorithm for parameter estimation is then discussed in Section 3.5.4.

3.5.1 Model-Based Clustering

The concept of model-based clustering is introduced here, as it is always associated with the concept of GMM, and its function is also supported by the R package, which is the “mclust” (Scrucca, et al., 2016) package. The main difference between each model is in the parameterisation of the covariance matrix, which causes some changes in the volume, shape and orientation of the contour of the components. In this package, 14 different models are supported, and the list of models available is shown in Table 3.2.

Table 3.2: List of models of GMM.

Model Name	$\mathbf{\Sigma_g}$	Volume	Shape	Distribution / Orientation
EII	$\lambda\mathbf{I}$	Equal	Equal	Spherical
VII	$\lambda_g\mathbf{I}$	Vary	Equal	Spherical
EEI	$\lambda\mathbf{A}$	Equal	Equal	Coordinate axes
VEI	$\lambda_g\mathbf{A}$	Vary	Equal	Coordinate axes
EVI	$\lambda\mathbf{A_g}$	Equal	Vary	Coordinate axes
VVI	$\lambda_g\mathbf{A_g}$	Vary	Vary	Coordinate axes
EEE	$\lambda\mathbf{DAD}^\mathbf{T}$	Equal	Equal	Equal
VEE	$\lambda_g\mathbf{DAD}^\mathbf{T}$	Vary	Equal	Equal
EVE	$\lambda\mathbf{DA_gD^T}$	Equal	Vary	Equal
VVE	$\lambda_g\mathbf{DA_gD^T}$	Vary	Vary	Equal
EEV	$\lambda\mathbf{D_gAD_g^T}$	Equal	Equal	Vary
VEV	$\lambda_g\mathbf{D_gAD_g^T}$	Vary	Equal	Vary
EVV	$\lambda\mathbf{D_gA_gD_g^T}$	Equal	Vary	Vary
VVV	$\lambda_g\mathbf{D_gA_gD_g^T}$	Vary	Vary	Vary

The parameterisation of the covariance matrix is done with the eigen-decomposition of the matrix (Scrucca, et al., 2016). Without applying any modification, the covariance matrix of each component can be decomposed as follows:

$$\mathbf{\Sigma_g} = \lambda_g \mathbf{D_g A_g D_g^T} . \tag{3.9}$$

In equation (3.9), λ_g is a constant that affects the volume of the contour. $\mathbf{A_g}$ is a diagonal matrix with determinant equals to one, and is responsible in controlling the shape of the contour. $\mathbf{D_g}$ is an orthogonal matrix, where the columns represent the eigenvectors, and affects the orientation of the contour (Scrucca, et al., 2016).

As observable from Table 3.2, equation (3.9) represents the covariance matrix for model “VVV”, while other covariance matrices are formed by setting one or more parameters to be equal for all the components. When λ_g is equal for every component, all the components will have a same volume. Setting $\mathbf{A_g}$ to be equal for every component causes the shape to be equal for every component. For the case of equal $\mathbf{D_g}$, the orientation of every contour will be the same.

Another possible case is that all the $\mathbf{D_g}$ matrices are identity matrices (represented by **I** in Table 3.2), where all the diagonal elements are equals to one, and all the off-diagonal elements are zero. In this case, all the contours will be aligned to the coordinate axes. Furthermore, if all the $\mathbf{A_g}$ matrices are also identity matrices, the contours will become spherical.

Many would prefer the model “VVV”, as the information exhibited is more complete as compared to other models. In some cases, other models are also considered, as they represent a parsimonious form of the “VVV” model. These models save a lot of computation time in large datasets, and sometimes perform even better than the “VVV” model.

3.5.2 Model-Based Hierarchical Agglomerative Clustering

In this section, the concept of MBHAC is introduced, as this step serves as an important step in the model selection and parameter estimation. The main purpose of MBHAC is to form a number of clusters from the dataset with the highest likelihood possible.

As the word “agglomerative” suggests, the clustering process is done with a “bottom-up” approach, which starts by putting each of the samples into a multivariate-normally-distributed cluster of only one sample. Then, two of the clusters are selected to combine with each other, and the selection is based on the changes in the value of the likelihood. It can be shown that the value of likelihood decreases whenever two clusters are merged into one. Therefore, from all the existing clusters, the two clusters that would cause the smallest drop in the likelihood value are merged (Scrucca, et al., 2016). This process is repeated until the required number of clusters is obtained.

From all the existing clusters, the mean vector $\boldsymbol{\mu}_{\mathbf{g}}$ and the covariance matrix $\boldsymbol{\Sigma}_{\mathbf{g}}$ can be obtained by using all the samples in that cluster. The mixing weight π_g can be estimated by the proportion of samples that belongs to that cluster. At this point, a GMM is already formed as all the parameters are obtained.

However, it is not guaranteed that the estimated values of the parameters here represent the Maximum Likelihood Estimation (MLE) of the real parameters.

3.5.3 Model Selection

This section introduces the metric used to select the best model to represent the GMM for the dataset. In model selection, the two main criteria that are considered are the number of component (*G*) and the model for covariance matrices, which have been discussed in Section 3.5.1.

For each of the covariance matrix models and the value of *G*, the value of Bayesian Information Criterion (BIC) (Schwarz, 1978) is computed as follows:

$$BIC_{M,G} = 2l_{M,G}(\mathbf{x}) - v\log(n). \tag{3.10}$$

From equation (3.10), *n* represents the number of samples in the dataset, and *v* represents the number of parameters that are estimated in the model. The model of covariance matrix and the number of component are represented by *M* and *G* respectively. $l_{M,G}(\mathbf{x})$ represents the log-likelihood of the model, which is calculated by using the model obtained with MBHAC.

After the value of BIC is obtained from all the possible models, the model with the highest BIC value is selected as the best model to represent the dataset. However, only the training set of data is used to form the model here, and the best model for training set does not necessary represent the best model for the test set of data. In the case of overfitting, other models with slightly lower BIC values are also considered.

3.5.4 Parameter Estimation

After a model is selected, the last step is to obtain a better estimate for all the parameters in all the components of the GMM. The traditional way of doing this is the MLE, which is attempted by many researchers but with no success, as the solution does not have a closed form for the case of GMM. Therefore, the method used here is the Expectation-Maximisation (EM) algorithm, which is a numerical method used in maximising objective functions in the form of likelihood functions (McLachlan and Krishnan, 2008). By using this method, it can be shown that the value of the likelihood increases monotonously in every iteration.

The EM algorithm consists of three main steps, namely the initialisation step, the Expectation-step (E-step) and the Maximisation-step (M-step) (McLachlan and Krishnan, 2008). The initialisation step is used to initialise all the required parameters in the model, which is done by using MBHAC (Scrucca, et al., 2016).

The next step is the E-step, which is used to predict the soft membership of each of the samples. The soft membership here implies that each sample has a probability of falling into one of the components, and the E-step helps to obtain all these probabilities. This is one of the advantages over the hard membership, which assign each sample to one and only one component.

For simplicity purpose, a latent variable $q_i^{(g)}$ is introduced where:

$$q_i^{(g)} = P(z_i = g \mid \mathbf{x_i}) . \tag{3.11}$$

Equation (3.11) defined $q_i^{(g)}$ to be equal to the probability that the *i*-th sample belongs to the *g*-th component. In order to obtain the value of $q_i^{(g)}$, the following equation is used (McLachlan and Krishnan, 2008):

$$P(z_i = g \mid \mathbf{x_i}) = \frac{P(\mathbf{x_i} \mid z_i = g) P(z_i = g)}{P(\mathbf{x_i})} . \tag{3.12}$$

It can be seen that equation (3.12) is another variation of the Bayes' Theorem, which utilizes the posterior probabilities as the soft membership of the components. In this case, the prior probability of the *g*-th component, $P(z_i = g)$ is equals to the mixing weight of the component, π_g. The likelihood function of the component, $P(\mathbf{x_i} \mid z_i = g)$ is represented by a multivariate normal distribution with a mean vector $\boldsymbol{\mu}_\mathbf{g}$ and a covariance matrix $\boldsymbol{\Sigma}_\mathbf{g}$ as the parameters. The marginal probability $P(\mathbf{x_i})$, again, is omitted from the calculation and dealt with the same way as in NBC. Obtaining the value of $q_i^{(g)}$ for all the components and all the samples concludes the E-step (McLachlan and Krishnan, 2008).

The M-step is used to update all the parameters with new values based on the $q_i^{(g)}$ obtained in E-step. Each of the parameters is estimated by using the concept of weighted average (McLachlan and Krishnan, 2008). The mixing weight π_g, the mean vector $\boldsymbol{\mu}_\mathbf{g}$ and the covariance matrix $\boldsymbol{\Sigma}_\mathbf{g}$ can be estimated by using the following equations (McLachlan and Krishnan, 2008):

$$\pi_g = \frac{1}{n}\sum_{i=1}^{n} q_i^{(g)}\ , \tag{3.13}$$

$$\boldsymbol{\mu}_\mathbf{g} = \frac{\sum_{i=1}^{n} q_i^{(g)} \mathbf{x}_\mathbf{i}}{\sum_{i=1}^{n} q_i^{(g)}}\, , \tag{3.14}$$

$$\boldsymbol{\Sigma}_\mathbf{g} = \frac{\sum_{i=1}^{n} q_i^{(g)} (\mathbf{x}_\mathbf{i} - \boldsymbol{\mu}_\mathbf{i})(\mathbf{x}_\mathbf{i} - \boldsymbol{\mu}_\mathbf{i})^T}{\sum_{i=1}^{n} q_i^{(g)}}\, . \tag{3.15}$$

After obtaining the new estimates for all the parameters, these new estimates are then used to replace the original estimates. For the likelihood of the GMM to converge, the E-step and M-step of the EM algorithm is repeatedly performed by iteratively update the parameters with the new estimates (McLachlan and Krishnan, 2008). The likelihood is said to have achieved the maximum when the changes in the value of the parameters between iterations are very low (McLachlan and Krishnan, 2008).

As mentioned earlier in Section 3.5, a GMM is used to represent the likelihood function for a class in the classification problem. Therefore, for a three-class classification problem, three GMM is required. The posterior probability for the classes is then computed with the same algorithm as in NBC.

3.6 Gaussian Process Classification

According to Rasmussen and Williams (2006), GP is defined as “a collection of random variables, in which any finite subset of these variables has a joint Gaussian distribution”. In this study, GP is used to represent a distribution over a function, which is then applied in the context of regression and classification.

Suppose that a function $f(\mathbf{x})$ has a GP distribution, the notation for the distribution of the function is as follows:

$$f(\mathbf{x}) \sim GP(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')) \,. \tag{3.16}$$

From equation (3.16), $m(\mathbf{x})$ represents the mean function of $f(\mathbf{x})$, while $k(\mathbf{x}, \mathbf{x}')$ represents the covariance function or kernel function of $f(\mathbf{x})$, with **x** and **x’** represent two input vectors for the function. In most of the cases, the mean function $m(\mathbf{x})$ is assumed zero for the simplicity of the notation, as it does not affect the results (Rasmussen and Williams, 2006). There is no particular standard or rule used to determine the type of covariance function to be used in a model, and therefore some common covariance functions are usually used. Table 3.3 summarises some of the covariance functions that are provided in the packages used in this study (Lama and Girolami, 2016; Karatzoglou, et al., 2004).

Table 3.3: Some commonly used covariance functions.

Name of the Covariance Function	Equation of the Covariance Function
Linear / Dot-product kernel	$\mathbf{x}^{\mathbf{T}}\mathbf{x}'+c$
Polynomial kernel	$(\alpha\mathbf{x}^{\mathbf{T}}\mathbf{x}'+c)^{d}$
Gaussian kernel	$\exp\left(-\frac{\lvert\mathbf{x}-\mathbf{x}'\rvert^{2}}{2\sigma^{2}}\right)$
Laplacian kernel	$\exp\left(-\frac{\lvert\mathbf{x}-\mathbf{x}'\rvert}{\sigma}\right)$
Cauchy kernel	$\frac{1}{1+\frac{\lvert\mathbf{x}-\mathbf{x}'\rvert^{2}}{\sigma^{2}}}$

It is good to take note that the concept of GPC is extended from Gaussian Process Regression (GPR). Therefore, the concept of GPR is introduced in Section 3.6.1, while the concept of GPC is introduced in Section 3.6.2.

3.6.1 Gaussian Process Regression

In the case of a linear regression, the objective is to predict the value of a function $f(\mathbf{x})$ of a new input vector $\mathbf{x}$. The function $f(\mathbf{x})$ can be written in the form as follows:

$$f(\mathbf{x})=\mathbf{x}^{\mathbf{T}}\mathbf{w}\,. \tag{3.17}$$

As a function of variables, the vector $\mathbf{x}$ is represented by unknown variables, which is from x_1 to x_5 in the case of the diabetes data. The vector $\mathbf{w}$ represents the vector of regression parameters or coefficients, which can take any real number value. In some cases, mapping the input vectors into a higher-dimensional vector space may results in a better model. In this case, the equation can be rewritten as follows:

$$f(\mathbf{x}) = \phi(\mathbf{x})^T \mathbf{w} . \tag{3.18}$$

In equation (3.18), $\phi(\mathbf{x})$ is the higher-dimensional vector transformed from the original vector **x**. Also, the length of the vector **w** is adjusted to be equal to the length of $\phi(\mathbf{x})$. Since the dataset is subjected to noise, the residual variable ε is also introduced into the model, which results to the following model:

$$y = f(\mathbf{x}) + \varepsilon . \tag{3.19}$$

By applying the concept of Bayesian regression into the model in equation (3.18), a prior distribution is set on the vector **w**, and a zero-mean Gaussian distribution is chosen to perform the function. The notation of the distribution of the vector *w* is shown below:

$$\mathbf{w} \sim N(\mathbf{0}, \mathbf{\Sigma_p}) . \tag{3.20}$$

However, a problem faced by the model in equation (3.18) is that the length and elements of the vector $\phi(\mathbf{x})$ is unknown. While obtaining this information is possible, the algorithm takes a large amount of time. Therefore, the use of GP is suggested, where the prior distribution is set on the whole $f(\mathbf{x})$ instead of the vector **w**. From equation (3.18) and equation (3.20), it can be shown that the distribution of $f(\mathbf{x})$ is as follows:

$$f(\mathbf{x}) \sim GP(0, \phi(\mathbf{x})^T \mathbf{\Sigma_p} \phi(\mathbf{x'})) . \qquad (3.21)$$

Again, the vector $\phi(\mathbf{x})$ is unknown in equation (3.21), and this is where the kernel trick comes in. The covariance function in equation (3.21) is replaced by one of the commonly used covariance functions, which only depends on the original input vector **x**, and possibly some other tuning parameters, as observable from Table 3.3. This step helps to save a lot of computation time from obtaining the vector $\phi(\mathbf{x})$ while obtaining the same results.

For simplicity of notation, f_i is used to represent the value of $f(\mathbf{x_i})$, which is the output for the *i*-th input vector. Furthermore, a vector $\mathbf{f}$ is used to represent the collection of values of f_i from the training set of data, while a vector $\mathbf{f*}$ is used for the test set (Rasmussen and Williams, 2006). According to the definition of GP above, both of the vectors should follow a multivariate Gaussian distribution. Putting both vectors together results in the following distribution:

$$\begin{bmatrix} \mathbf{f} \\ \mathbf{f*} \end{bmatrix} \sim N\left(0, \begin{bmatrix} \mathbf{K}(\mathbf{X},\mathbf{X}) & \mathbf{K}(\mathbf{X},\mathbf{X*}) \\ \mathbf{K}(\mathbf{X*},\mathbf{X}) & \mathbf{K}(\mathbf{X*},\mathbf{X*}) \end{bmatrix} \right) . \qquad (3.22)$$

In the covariance matrix in equation (3.22), the matrix $\mathbf{K}(\mathbf{X},\mathbf{X*})$ represents the collection of the values of covariance functions between the samples in the training set and the test set, and the same applies to $\mathbf{K}(\mathbf{X},\mathbf{X})$ and $\mathbf{K}(\mathbf{X*},\mathbf{X*})$. In addition, the noises in the training set of data can be considered by replacing the vector **f** with a vector **y**, and by adding the variance of the training set into each diagonal element in the covariance matrix in equation (3.22).

Then, from equation (3.22), the predictive distribution of all the samples in the test set can be obtained by taking the posterior distribution of $\mathbf{f*}$ given $\mathbf{f}$, which can be proven to be as follows:

$$\begin{aligned}\mathbf{f*} \mid \mathbf{X},\mathbf{X*},\mathbf{f} \sim N(&\mathbf{K}(\mathbf{X*},\mathbf{X})\mathbf{K}(\mathbf{X},\mathbf{X})^{-1}\mathbf{f},\\ &\mathbf{K}(\mathbf{X*},\mathbf{X*})-\mathbf{K}(\mathbf{X*},\mathbf{X})\mathbf{K}(\mathbf{X},\mathbf{X})^{-1}\mathbf{K}(\mathbf{X},\mathbf{X*})).\end{aligned} \tag{3.23}$$

From the distribution shown in equation (3.23), the values of the mean are then used as the predicted values of these samples, and this concludes the algorithm for GPR. The similar concept is then used in GPC.

3.6.2 Gaussian Process Classification

GPC uses a concept similar to the logistic regression, which transforms the output of a linear regression function into class probabilities through a logistic function (Rasmussen and Williams, 2006). In the case of GPC, the linear regression function is the GP discussed in Section 3.6.1.

As an extension to the GPR, all the variables used here are defined in the same way as in Section 3.6.1, with the exception of variable *y*, which takes categorical value instead of continuous value, and is used to represent the class membership of the sample. Two new vectors, **y** and **y*** are also introduced, which represent a collection of values of y_i from the training set and test set respectively. Also, the function $f(\mathbf{x})$ is defined in the same way as in equation (3.18), equation (3.19) and equation (3.21).

The first step in GPC involves the Bayes' Theorem, which is stated as follows:

$$P(\mathbf{f} \mid \mathbf{X}, \mathbf{y}) = \frac{P(\mathbf{y} \mid \mathbf{f})P(\mathbf{f} \mid \mathbf{X})}{P(\mathbf{y} \mid \mathbf{X})}. \tag{3.24}$$

As in equation (3.21), a GP distribution is used as a prior distribution for $f(\mathbf{x})$, and therefore the prior distribution $P(\mathbf{f} \mid \mathbf{X})$ of the vector **f** is multivariate Gaussian, with a mean vector of zero and a covariance matrix of $\mathbf{K}(\mathbf{X}, \mathbf{X})$. For the likelihood function $P(\mathbf{y} \mid \mathbf{f})$, any sigmoid function can be used to transform the output into class probability, and the two most commonly used functions are the logistic function and the probit function (Rasmussen and Williams, 2006). The computation of the marginal probability $P(\mathbf{y} \mid \mathbf{X})$ is omitted with a similar reason as in the previous sections.

After obtaining the posterior probability in equation (3.24), the next step is to obtain the predictive distribution for the vector **f***. It can be shown that the predictive distribution (Rasmussen and Williams, 2006) can be obtained by using the following equation:

$$P(\mathbf{f}^* \mid \mathbf{X}, \mathbf{y}, \mathbf{X}^*) = \int P(\mathbf{f}^* \mid \mathbf{X}, \mathbf{X}^*, \mathbf{f})P(\mathbf{f} \mid \mathbf{X}, \mathbf{y})d\mathbf{f}\ . \tag{3.25}$$

In equation (3.25), the probability $P(\mathbf{f} \mid \mathbf{X}, \mathbf{y})$ is obtained from the previous step, while $P(\mathbf{f^*} \mid \mathbf{X}, \mathbf{X^*}, \mathbf{f})$ is obtained from the distribution mentioned in equation (3.23). However, when both of the probabilities are substituted into equation (3.25), it can be seen that the integral is intractable because of the sigmoid function (Rasmussen and Williams, 2006). Therefore, some analytical approximation methods can be used to obtain an approximated solution to the integral (Rasmussen and Williams, 2006). In this study, Laplace Approximation method and Variational Bayes method are used.

The last step of GPC is to obtain the class probabilities for the test set of data, which is represented by the vector **y***. This is done by obtaining the posterior probabilities for each of the possible values of **y***, and classifying each sample into the class having the highest posterior probability (Rasmussen and Williams, 2006). The equation used to obtain these probabilities is similar to the predictive distribution above, and is shown in the equation below:

$$P(\mathbf{y^*} \mid \mathbf{X}, \mathbf{y}, \mathbf{X^*}) = \int P(\mathbf{y^*} \mid \mathbf{f^*}) P(\mathbf{f^*} \mid \mathbf{X}, \mathbf{y}, \mathbf{X^*}) d\mathbf{f}^* . \qquad (3.26)$$

In equation (3.26), the probability $P(\mathbf{f^*} \mid \mathbf{X}, \mathbf{y}, \mathbf{X^*})$ is obtained from equation (3.25), while $P(\mathbf{y^*} \mid \mathbf{f^*})$ is obtained by using the same sigmoid function as in equation (3.24). The existence of sigmoid function in equation (3.26) causes the integral to be intractable in some cases, and the solution to this is the same as in equation (3.25). Solving this integral should results in numerical solutions, and the classification of the samples in the test set can be done.

CHAPTER 4

RESULTS AND DISCUSSION

After studying all the three classification methods, the algorithms were applied on the diabetes dataset. In this chapter, the overall algorithm of the study is discussed again, with the aid of the outputs and the results obtained from the R. The results and performances of the classification methods were then evaluated and compared with each other, in order to determine the most suitable classification method for this dataset.

Section 4.1 focuses on the selection of samples for the training set and the test set. Section 4.2 explains the result obtained by using NBC, while Section 4.3 and Section 4.4 explain the results obtained by GMM classification and GPC respectively. In Section 4.5, comparison was being done on the three classification methods, and the most suitable classification method was selected.

4.1 Selection of Training Set and Test Set

As mentioned in the previous chapter, the first step of the analysis was separating the original dataset into two different sets in a ratio of 6:4. Therefore, simple random sampling was carried out to select 87 samples out of the original 145 samples, and these samples were placed in the training set. The other 58 samples were placed in the test set for the performance evaluation of the classifiers in the later sections.

In order to represent the original dataset well, the training set should contain samples from a broad range of values, up to an extent similar to the original dataset. Figure 4.1 shows a scatterplot matrix illustrating the distribution of samples from both sets of data, with the red triangles representing training set samples and the black circles representing test set samples.

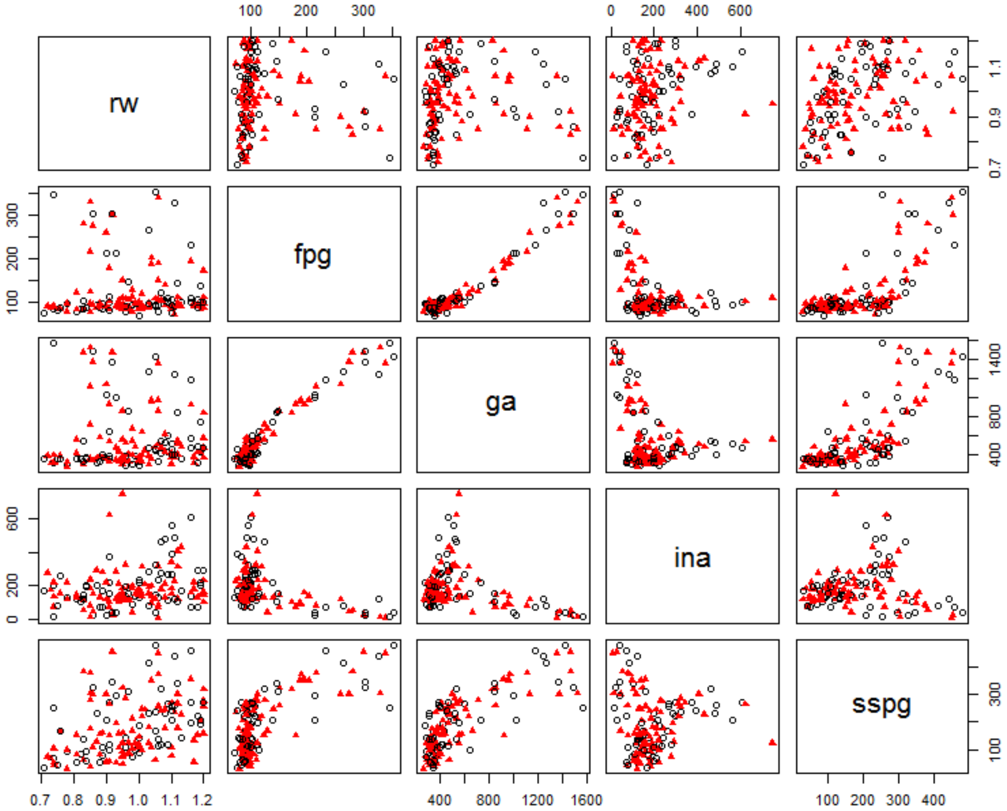

Figure 4.1: Scatterplot matrix of the original data.

It can be seen from Figure 4.1 that the samples in both sets were able to spread throughout the plot instead of cluttering together at some points, and therefore the two sets should be good enough to be used. Besides that, the training set should also consisted of sufficient samples from each of the diabetes classes, and therefore the proportion of samples of each class in both sets of sample was computed and shown in Table 4.1. The proportions in the original dataset were also included in the table.

Table 4.1: Number and proportion of samples of each class in each set.

Set of Samples	Number (and Proportion) of Samples of Each Class in Each Set			
	Chemical	Normal	Overt	Total
Training Set	21 (0.2414)	46 (0.5287)	20 (0.2299)	87 (1.0000)
Test Set	15 (0.2586)	30 (0.5172)	13 (0.2241)	58 (1.0000)
Original Set	36 (0.2483)	76 (0.5241)	33 (0.2276)	145 (1.0000)

By comparing the proportions in the three sets of samples, it can be seen that both the training set and the test set contain sufficient number of samples from each class to represent the original dataset. Therefore, these training and test sets were used for the analysis of each classification methods in the following sections.

Figure 4.2 and Figure 4.3 illustrated the distribution of samples of each class in the training set and the test set respectively. In these scatterplot matrices, the “Normal” samples, “Chemical” diabetes patients and “Overt” diabetes patients were denoted by black triangles, red circles and green squares respectively. As an act of standardisation, the same symbols were also being used to represent the three classes in all of the plots in the following sections.

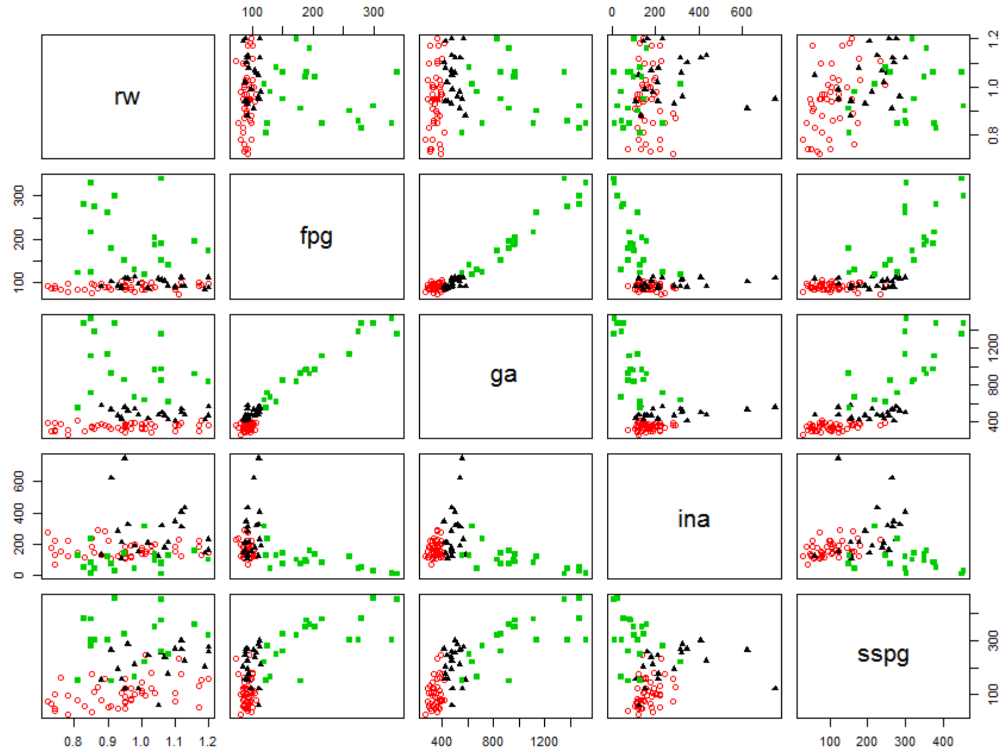

Figure 4.2: Scatterplot matrix of the training set of data.

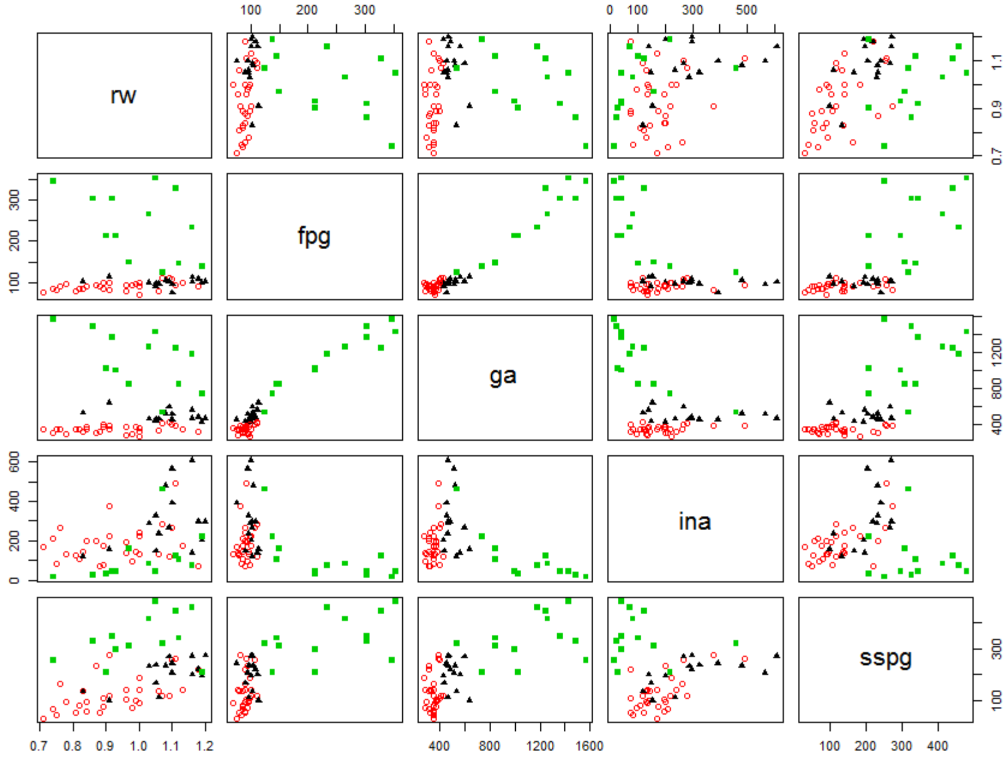

Figure 4.3: Scatterplot matrix of the original test set of data.

4.2 Results of Naïve Bayes Classification

As mentioned in the previous chapter, NBC assumes that the variables in the dataset are independent with each other, which seems unreasonable as observed from the scatterplot matrices in the previous section. In addition, each variable in each class is assumed to follow a Gaussian distribution in the algorithm of NBC, while the actual distribution is not verified. Therefore, Shapiro-Wilk Normality Test (Shapiro and Wilk, 1965) was performed to test whether the variables were normally distributed. Table 4.2 showed the p-value of each variable in each of the classes, which implied that the variable was normally distributed if the value is greater than 0.01.

Table 4.2: The p-values of the Shapiro-Wilk Normality Test.

	p-values		
Variables	Chemical	Normal	Overt
rw	0.16010	0.09874	0.43620
fpg	0.22650	0.8231	0.01167
ga	0.08189	0.4234	0.11430
ina	9.629×10^{-4}	3.706×10^{-6}	3.847×10^{-5}
sspg	0.07716	2.933×10^{-4}	0.51630

From Table 4.2, we can see that the variable “ina” was not normally distributed in every class, while variable “sspg” was not normally distributed in “Normal” class. Therefore, it can be concluded that the assumptions made in the algorithm of NBC had been violated. In spite of that, NBC was still applied on this dataset with the reason that the algorithm is much simpler than all the other methods, and the efficiency of this method will be evaluated accordingly later in this section.

The prior probability $f(C)$ was first obtained from the training set of data. As mentioned in the algorithm, the prior probability is equals to the proportions of each class in the training set, as shown in the first row of Table 4.1. To form the likelihood function, the sample mean and standard deviation of each of the variables in each class were computed. The model was then formed by using equation (3.7), and the posterior probabilities for each sample in the test set was computed by substituting the values of the five variables into the equation. All the test set samples were then classified according to the highest posterior probabilities, and the resulting classes were compared with the original classes.

The results for the NBC were illustrated with a confusion matrix, as shown in Table 4.3. The accuracy obtained by NBC is 86.2069%, which was equivalent to eight classification errors. The results were much better than expected, considering that the independence assumption was heavily violated. In addition, Figure 4.4 showed a scatterplot matrix for the test set of data, with the classes labelled based on the results of NBC.

Table 4.3: Confusion matrix for the results of NBC.

	Resulting Class of NBC		
Original Class	Chemical	Normal	Overt
Chemical	13	1	1
Normal	5	25	0
Overt	1	0	12

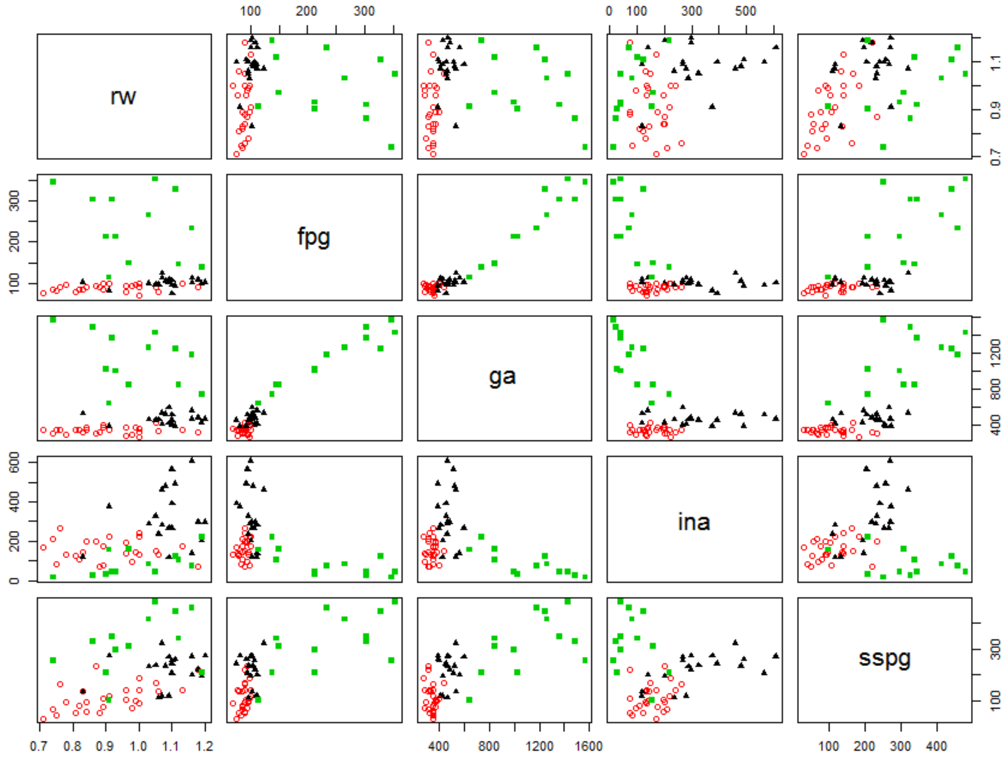

Figure 4.4: Scatterplot matrix of the results of NBC.

4.3 Results of Gaussian Mixture Model Classification

The process of model fitting for GMM classification focuses on the likelihood function, where a GMM was used for every class. The challenging part in this step was to choose the best model to represent the dataset, and at the same time avoiding the overfitting of the training set. In most cases, a large number of

GMM components led to overfitting, while a small number results in underfitting.

Based on the algorithm discussed and some trial-and-error, the best model was obtained, where the samples in both the training set and test set can be classified with a low classification error. With the aid of some built-in-functions in the “mclust” package, the components of the GMM models were illustrated in Figure 4.5 by using the scatterplot matrix.

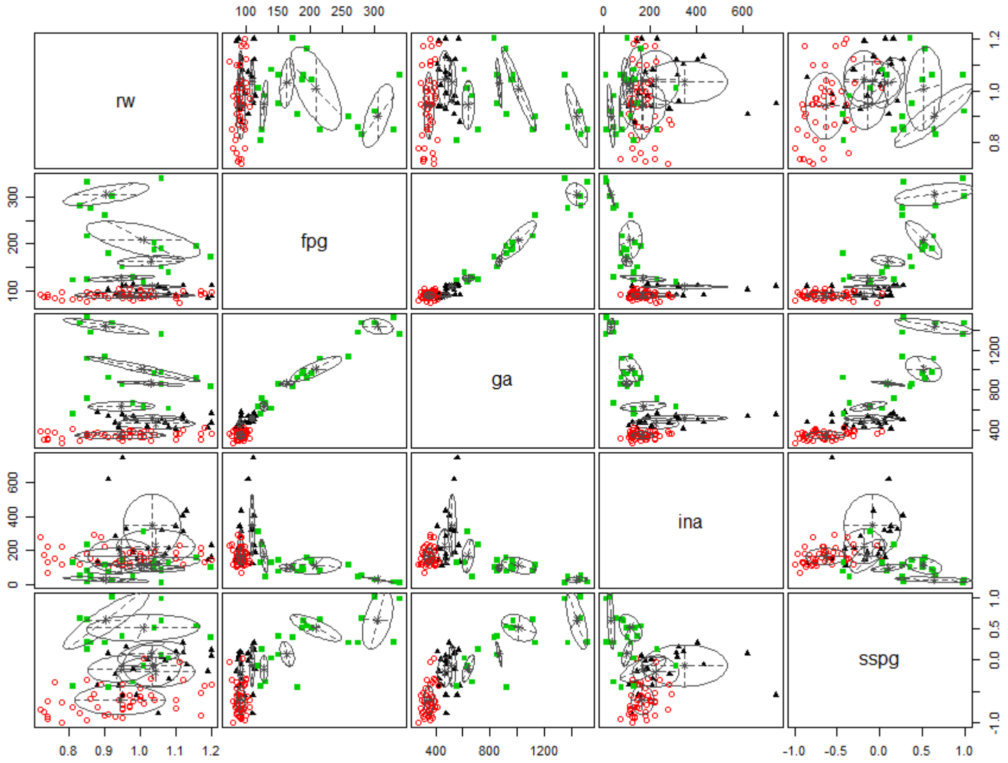

Figure 4.5: Scatterplot matrix of the training set illustrating the GMM clusters.

While it may be difficult to determine the classes of the clusters from Figure 4.5, the text output of R was able to explain the model fitted in an organised way. Table 4.4 summarised the output by stating the model and the number of components used to fit the model in each of the classes.

Table 4.4: GMM fitted based on the training set.

Class	Model	Number of Components, *G*
Chemical	EVI	2
Normal	XXI	1
Overt	VEV	4

Although there are a large number of “Normal” samples in the training set, the way the samples were distributed allow them to be well represented by one component. On the other hand, the distribution of “Chemical” samples was slightly dispersed from an ellipsoidal cluster, and therefore two components were being used for a better representation. For “Overt” samples, since the values for some of the variables have a broader range, four components were used, where each of the components takes up a part of the range. The “EVI” and “VEV” models have been explained in Table 3.2, while “XXI” model was used for a single-component GMM aligned to the coordinate axes (Scrucca, et al., 2016).

The posterior probabilities were then computed by using the GMM as the likelihood function, and by using the same prior probability as in NBC. The confusion matrix and the scatterplot matrix for the results of GMM classification showed in Table 4.5 and Figure 4.6 respectively. With only five misclassified samples, this classification method has achieved an accuracy of 91.3793%.

Table 4.5: Confusion matrix for the results of GMM classification.

	Resulting Class of GMM		
Original Class	Chemical	Normal	Overt
Chemical	15	0	0
Normal	3	27	0
Overt	2	0	11

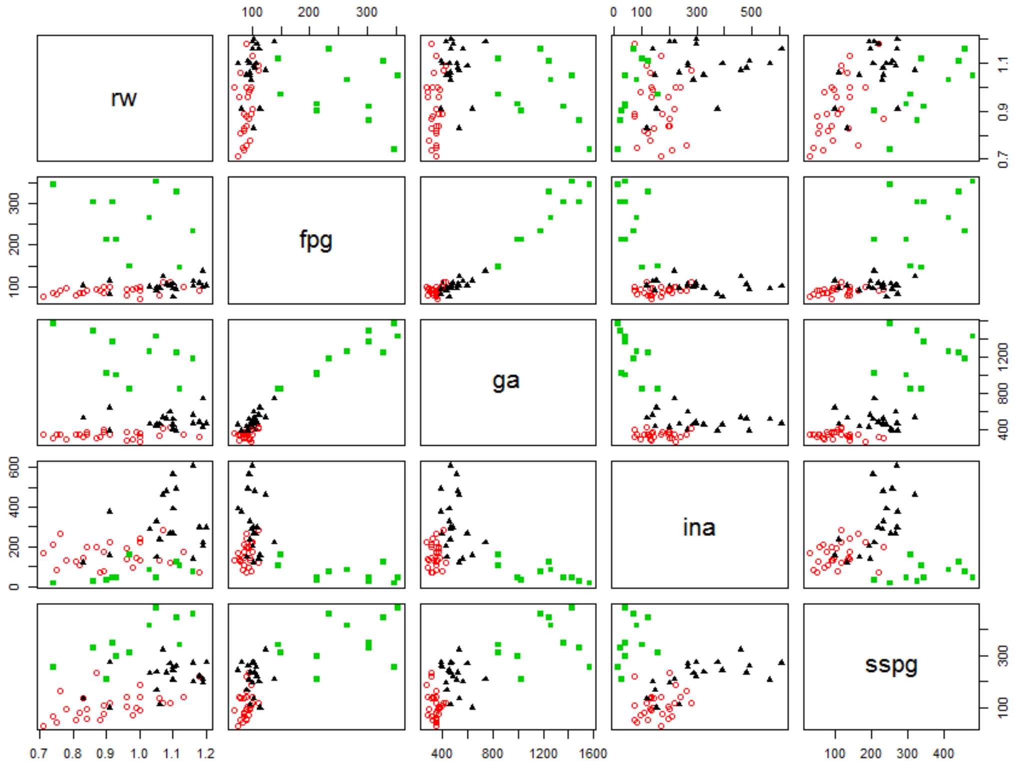

Figure 4.6: Scatterplot matrix of the results of GMM classification.

4.4 Results of Gaussian Process Classification

Unlike the other two methods, GPC involves a more complex algorithm by mapping the input vector into a higher-dimensional vector space, making the process difficult to be illustrated physically. One of the properties of GPC that significantly affects the classification results is the choice of covariance function for the GP model. Since there was no algorithm in determining the best covariance function, the trial-and-error approach was used by applying all the built-in covariance functions in the “vbmp” and “kernlab” packages. By using this approach, the covariance function selected is the Cauchy kernel, which performed the classification with the lowest classification error as compared to the others. The model training and prediction of test set samples were performed according to the algorithm of GPC, with the use of Variational Bayes inference to solve the integrals in equation (3.25) and equation (3.26).

Table 4.6 and Figure 4.7 summarised the results of the classification in the form of confusion matrix and scatterplot matrix respectively. The accuracy achieved by this classification method is 89.6552%, which was equivalent to six misclassifications.

Table 4.6: Confusion matrix for the results of GPC.

	Resulting Class of GPC		
Original Class	Chemical	Normal	Overt
Chemical	12	3	0
Normal	2	28	0
Overt	1	0	12

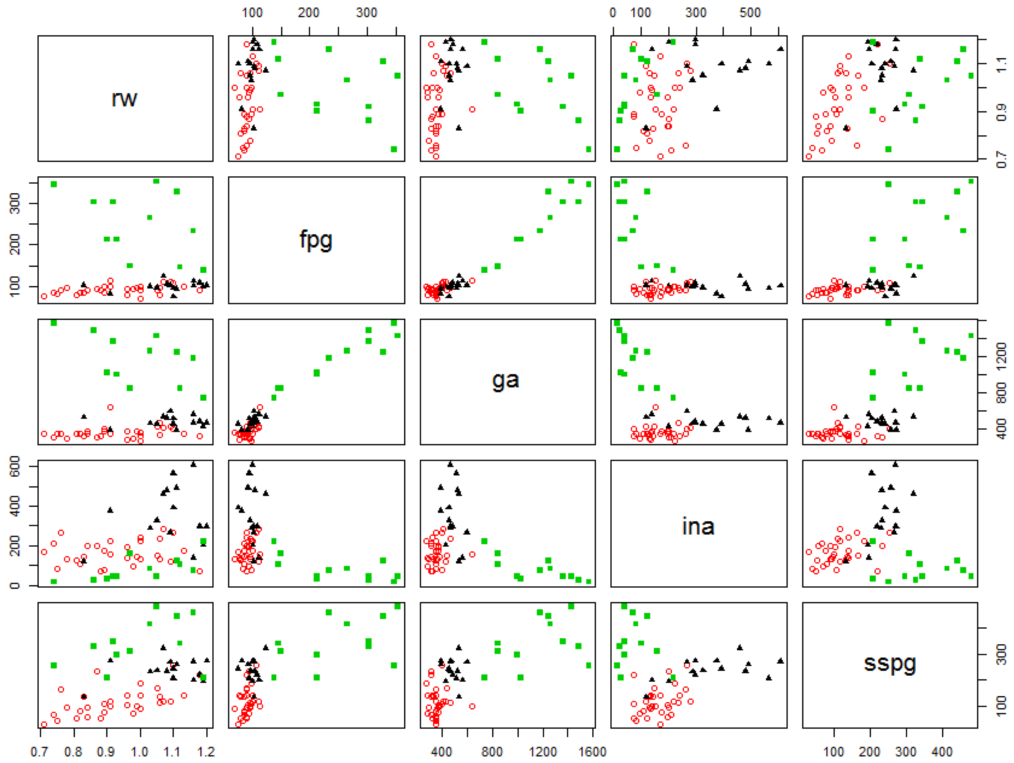

Figure 4.7: Scatterplot matrix of the results of GPC.

4.5 Comparing the Results of the Classification Methods

To determine the best classification method among the three methods, the main measure used is the accuracy. As stated previously, GMM classification gave the highest accuracy of 91.3793%, which was slightly higher than GPC's 89.6552% and NBC's 86.2069%. In terms of percentage, the difference in the accuracy may seem small, but every single percent of difference would be equivalent to a huge increase in misclassified samples in a larger dataset, and this was the reason accuracy was the main criteria evaluated.

In the context of diabetes, it is known that "Overt" diabetes is more serious than "Chemical" diabetes, while "Normal" samples indicate that the person is not diagnosed with diabetes. Among all the cases of misclassifications, classifying a patient into a less serious class than its actual class may be harmful to the patient, because the patient would believe that he or she remained in a safe condition, however the reality denied. The other direction of misclassification might not cause a serious impact, as it might help the patients to be alerted of their conditions.

By referring to the results of the three classifiers, the percentage of misclassification that “underestimated” the patients’ condition was computed. Out of the eight misclassifications, NBC had underestimated only two of them and overestimated the rest. GMM classification resulted in two underestimated samples out of five misclassifications, while GPC had two out of six. In this case, GMM did a better job than GPC, but slightly behind NBC in terms of percentages. However, since both NBC and GMM resulted in two underestimated samples, GMM still did a better job by considering the overall accuracy.

Another point of interest is on the computational time for the classifications, which is affected by the complexity of the algorithms. The use of straightforward algorithm in NBC allowed the classification to be completed in no time. GMM classification involves EM algorithm, which uses iterative approach to get the converged estimates of the parameters. Despite involving an iterative algorithm, GMM classification took less than one second to complete its task. On the other hand, GPC involves the Variational Bayes method, which involves a more complicated iterative algorithm than EM algorithm. On average, the number of iterations used in a single run of GPC ranged between nine and 15 iterations, with each iteration taking up around 2.3

seconds, and the time taken is expected to increase in a larger dataset. This placed GPC in a more disadvantageous position compared to NBC and GMM.

Table 4.7 summarised the results of all the three classification methods, which includes the criteria discussed in this section.

Table 4.7: Summary of the results of the three classification methods.

	Classification Method		
Criteria	NBC	GMM	GPC
Accuracy (Number of misclassification)	86.2069% (8)	91.3793% (5)	89.6552% (6)
Percentage of underestimated misclassifications (Number of underestimations)	25% (2)	40% (2)	66.6667% (4)
Time taken for the classification process	< 1 second	< 1 second	20 – 40 seconds

Besides those mentioned above, each of the classification methods has their own strengths and weaknesses. Although NBC is computational-wise convenient, the results are affected by the assumptions on independence and distribution, while the same assumptions are not being applied on GMM and GPC. This can be seen in the case of the diabetes dataset, where variables "fpg" and "ga" are highly correlated with each other, and variable "ina" is not normally distributed. In spite of these, the performance of NBC was considered to be as good as the other two methods, and NBC is expected to be performing better in other datasets where the assumptions are fulfilled.

GMM is considered as a powerful tool, as it can be used to represent almost any distributions by using multiple Gaussian distributions. However, this characteristic also leads to the risk of overfitting the dataset, and therefore a careful selection of the model is acquired. In some cases, the best classification model could have been formed by GMM. Identifying the model is difficult as it lies in between many other models that either overfit or underfit the dataset. Therefore, the time consuming part of GMM is not in the algorithm itself, but lies in the stage of model selection.

For the case of GPC, the use of covariance functions involves mapping the input vectors into a higher-dimensional feature space. This process allows most researchers to identify the latent structures or features, which lies hidden within the dataset and not visible through the original variables. However, this is highly dependent on the choice of the covariance function, which also affects the results of classification. As there is countless number of choices for the covariance function, only some of the popular choices are being considered.

To conclude the analysis of the results, the most suitable classification method that can efficiently classify the samples in the diabetes dataset is the GMM classification, which is followed by GPC and NBC. The main reason lies in the way that the dataset is distributed, where in this case GMM is more favoured than the other methods.

CHAPTER 5

CONCLUSIONS

5.1 Concluding Remarks

The motivation for this study came from the high number of misdiagnosed diabetes, which is one of the leading causes of death globally. This study set off to search for the most suitable classification method to be applied on a diabetes dataset, which is used to represent a real-world case. In order to achieve this, three different Bayesian classification methods have been studied, namely NBC, GMM and GPC. For each classification method, a classification model was trained by using 60% of the samples in the dataset, and the efficiency of the model was evaluated with the remaining samples.

Although the algorithm of NBC involves some assumptions on the independency and the distribution, the simplicity of algorithm allowed the whole classification process to be completed with the speed of light. Without fulfilling all the assumptions, NBC still managed to achieve an accuracy of 86%. On the other hand, GMM classification involves a more complicated algorithm than NBC, but the difference is insignificant in terms of time taken. With some efforts in the model selection, GMM has achieved an accuracy of 91%. For the case of GPC, mapping the vector of variables into a higher-dimensional space and implementing numerical methods to solve intractable integrals costed a large amount of time in the process. In exchange for that, GPC managed to reach a high accuracy of 90%.

Based on the accuracy and some other criteria, the most suitable classification method for the diabetes dataset is GMM, which is followed by GPC and NBC. This study has been done in hope to contribute to the field of diabetes, so that the number of misdiagnosed diabetes can be reduced. It is important to take note that the most suitable classification method for every dataset is different, and a brand new analysis should be done on the dataset before deciding a classification method to be applied.

5.2 Recommendations for Future Studies

As mentioned previously, the same classification methods that do well on a dataset might not be doing as well on other dataset. Therefore, future studies can focus on applying these classification methods to some other datasets, especially those with a larger sample size and a higher number of variables. A throughout study can be considered to obtain a classification method that can do well on different datasets. Also, many other popular classification methods can be studied, which includes Bayesian and non-Bayesian classification methods. Furthermore, since the trend for data mining is currently in rise, it is a good idea to extend the classification methods to explore the field of data mining.

REFERENCES

Agrawal, P. and Dewangan, A. K., 2015. A Brief Survey on the Techniques Used for the Diagnosis of Diabetes-Mellitus. *International Research Journal of Engineering and Technology (IRJET),* 02(03), pp. 1039-1043. [online] Available at <https://www.irjet.net/archives/V2/i3/Irjet-v2i3152.pdf> [Accessed 16 June 2017].

American Diabetes Association, 2014. Diagnosis and Classification of Diabetes Mellitus. *Diabetes Care,* 37(Supplement 1), pp. S81-S90. [online] Available at <http://care.diabetesjournals.org/content/diacare/37/Supplement_1/S81.full.pdf> [Accessed 23 June 2017].

Besler, E., Ruiz, P., Molina, R. and Katsaggelos, A. K., 2016. Classification of Multiple Annotator Data Using Variational Gaussian Process Inference. *IEEE,* Volume 24, pp. 2025-2029. [online] Available at <http://www.eurasip.org/Proceedings/Eusipco/Eusipco2016/papers/1570256378.pdf> [Accessed 20 February 2017].

Bolstad, W. M., 2007. *Introduction to Bayesian Statistics.* 2nd ed. New Zealand: John Wiley & Sons, Inc.

Challis, E., Hurley, P., Serra, L., Bozzali, M., Oliver, S. and Cercignani, M., 2015. Gaussian process classification of Alzheimer's disease and mild cognitive impairment from resting-state fMRI. *NeuroImage,* 112, pp. 232-243.

Deepthi, S., Ravikumar, A. and Nair, R. V., 2016. Evaluation of Classification Techniques for Arrhythmia Screening of Astronauts. *Procedia Technology,* 24, pp. 1232-1239. [online] Available at <http://www.sciencedirect.com/science/article/pii/S2212017316301888> [Accessed 19 February 2017].

Diabetes UK, 2017a. *Diabetes: the basics - Diabetes UK.* [online] Available at: <https://www.diabetes.org.uk/Diabetes-the-basics/> [Accessed 20 June 2017].

Diabetes UK, 2017b. *Prediabetes or Borderline Diabetes.* [online] Available at: <http://www.diabetes.co.uk/pre-diabetes.html> [Accessed 23 June 2017].

Fajans, S. S., 1973. The Definition of Chemical Diabetes. *Metabolism,* 22(2), pp. 211-217.

Gakidou, E., Mallinger, L., Abbott-Klafter, J., Guerrero, R., Villalpando, S., Ridaura, R. L., Aekplakorn, W., Naghavi, M., Stephen, L., Lozano, R. and Murray, C. J., 2011. Management of diabetes and associated cardiovascular risk factors in seven countries: a comparison of data from national health examination surveys. *Bulletin of the World Health Organization*, 89(3), pp. 172-183. [online] Available at <http://www.scielosp.org/pdf/bwho/v89n3/08.pdf> [Accessed 24 June 2017].

Guvenir, H. A., Acar, B. and Muderrisoglu, H., 1998. Arrhythmia Data Set. *UCI Machine Learning Repository*. [online] Available at: <https://archive.ics.uci.edu/ml/datasets/Arrhythmia> [Accessed 28 June 2017].

Johnson, R. A. and Wichern, D. W., 2007. *Applied Multivariate Statistical Analysis*. 6th ed. New Jersey: Pearson Prentice Hall.

Karatzoglou, A., Smola, A., Hornik, K. and Zeileis, A., 2004. kernlab - An S4 Package for Kernel Methods in R. *Journal of Statistical Software*, 11(9), pp. 1-20.

Kaur, G. and Chhabra, A., 2014. Improved J48 Classification Algorithm for the Prediction of Diabetes. *International Journal of Computer Applications*, 98(22), pp. 13-17.

Lagrange, A., Fauvel, M. and Grizonnet, M., 2016. Large-scale feature selection with Gaussian mixture models for the classification of high dimensional remote sensing images. *Transactions On Computational Imaging*, Volume Special Issue For Computational Imaging For Earth Sciences, pp. 1-13. [online] Available at <https://hal.archives-ouvertes.fr/hal-01382500v2/document> [Accessed 19 February 2017].

Lama, N. and Girolami, M., 2016. vbmp: Variational Bayesian Multinomial Probit Regression. R package version 1.42.0. [online] Available at <http://bioinformatics.oxfordjournals.org/cgi/content/short/btm535v1> [Accessed 12 February 2017].

Loader, C., 2013. locfit: Local Regression, Likelihood and Density Estimation. R package version 1.5-9.1. [online] Available at <https://CRAN.R-project.org/package=locfit> [Accessed 27 June 2017].

McLachlan, G. J. and Krishnan, T., 2008. *The EM Algorithm and Extensions*. 2nd ed. New Jersey: John Wiley & Sons, Inc..

Meyer, D., Dimitriadou, E., Hornik, K., Weingessel, A. and Leisch, F., 2017. e1071: Misc Functions of the Department of Statistics, Probability Theory Group (Formerly: E1071), TU Wien. R package version 1.6-8. [online] Available at <https://CRAN.R-project.org/package=e1071> [Accessed 12 February 2017].

National Institute of Diabetes and Digestive and Kidney Diseases, 1990. Pima Indians Diabetes Data Set. *UCI Machine Learning Repository.* [online] Available at: <https://archive.ics.uci.edu/ml/datasets/Pima+Indians+Diabetes> [Accessed 28 June 2017].

NIDDK, 2009. *Prediabetes & Insulin Resistance.* [online] Available at: <https://www.niddk.nih.gov/health-information/diabetes/overview/what-is-diabetes/prediabetes-insulin-resistance> [Accessed 23 June 2017].

Rasmussen, C. E. and Williams, C. K. I., 2006. *Gaussian Process for Machine Learning.* Cambridge: MIT Press.

Reaven, G. M. and Miller, R. G., 1979. An Attempt to Define the Nature of Chemical Diabetes Using a Multidimensional Analysis. *Diabetelogia,* Volume 16, pp. 17-24.

Schwarz, G., 1978. Estimating The Dimension Of A Model. *The Annals Of Statistics,* 6(2), pp. 461-464.

Scrucca, L., Fop, M., Murphy, T. B. and Raftery, A. E., 2016. mclust 5: Clustering, Classification and Density Estimation Using Gaussian Finite Mixture Models. *The R Journal,* 8(1), pp. 289-317.

Shapiro, S. S. and Wilk, M. B., 1965. An Analysis of Variance Test for Normality (Complete Samples). *Biometrika,* 52(3/4), pp. 591-611.

World Health Organization, 2016. *Global Report On Diabetes.* [pdf] France: WHO. Available at: <http://apps.who.int/iris/bitstream/10665/204871/1/9789241565257_eng.pdf?ua=1> [Accessed 20 June 2017].

World Health Organization, 2017. *The top 10 causes of death.* [online] Available at: <http://www.who.int/mediacentre/factsheets/fs310/en/> [Accessed 24 June 2017].

APPENDIX A

R Codes for Training Set and Test Set Preparation

```
#Loading datasets.
library(locfit)
data(chemdiab)
oriClass <- chemdiab$cc
oriData <- chemdiab[,-6]

#Sampling data, separation of training set and test set.
train <- sample(1:nrow(chemdiab), size = nrow(chemdiab)*0.6, replace = FALSE)
trainSet <- oriData[train,]
trainClass <- oriClass[train]
testSet <- oriData[-train,]
testClass <- oriClass[-train]
table(trainClass)
table(testClass)

#Plot the distribution of training set and test set.
isTrain <- rep(0, 145)
isTrain[train] <- 1
isTrain <- as.factor(isTrain)
pairs(oriData, col = isTrain, pch = c(1, 17)[as.numeric(isTrain)])

#Plot the scatterplot matrix for the training set.
pairs(trainSet, col = trainClass, pch = c(17, 1, 15)[as.numeric(trainClass)])

#Plot the scatterplot matrix for the original test set.
pairs(testSet, col = testClass, pch = c(17, 1, 15)[as.numeric(testClass)])
```

APPENDIX B

R Codes for Classification Methods

```
#Naive Bayes Classificaton
library(e1071)
nbModel <- naiveBayes(trainSet, trainClass)
nbResult <- predict(nbModel, testSet)
nbTable <- table(testClass, nbResult)
nbAcc <- mean(nbResult == testClass)

#Plot the scatterplot matrix for the nbModel test set.
pairs(testSet, col = nbResult, pch = c(17, 1, 15)[as.numeric(nbResult)])

#Gaussian Mixture Model for Classification
library(mclust)
gmmModel <- MclustDA(trainSet, trainClass, G = 1:4)
#G is used to control the range for the number of components.

summary(gmmModel, testSet, testClass, parameters = FALSE)
#If want to obtain the parameters, change its option to TRUE.

#Illustration of GMM clusters
plot(gmmModel, what = "scatterplot", colors = c(1, 2, 3), symbols = c(17, 1,
15))

#Plot the scatterplot matrix for the gmmModel test set.
gmmResult <- predict(gmmModel, testSet)$classification
pairs(testSet, col = gmmResult, pch = c(17, 1, 15)[as.numeric(gmmResult)])

#Gaussian Process Classification (VBMP)
library(vbmp)
system.time(gpModel <- vbmp(trainSet, trainClass, testSet, testClass,
                        theta = rep(1.0, ncol(trainSet)),
                        control = list(
                          sKernelType = "cauchy",
                          bThetaEstimate = TRUE,
                          bMonitor = TRUE,
                          InfoLevel = 1
                        )))
gpmResult <- as.factor(apply(gpModel$Ptest,1,which.max))
gpmTable <- table(testClass, gpmResult)

#Plot the scatterplot matrix for the gpModel test set.
pairs(testSet, col = gpmResult, pch = c(17, 1, 15)[as.numeric(gpmResult)])
```