IMPROVING HANDWRITTEN DIGIT RECOGNITION USING HYBRID FEATURE SELECTION ALGORITHM

WONG KHYE MUN

BACHELOR OF SCIENCE (HONS)

STATISTICAL COMPUTING AND

OPERATIONS RESEARCH

FACULTY OF SCIENCE

UNIVERSITI TUNKU ABDUL RAHMAN

APRIL 2022

IMPROVING HANDWRITTEN DIGIT RECOGNITION USING HYBRID FEATURE SELECTION ALGORITHM

By

WONG KHYE MUN

A project report submitted to the Department of
Physical and Mathematical Science

Faculty of Science

Universiti Tunku Abdul Rahman
in partial fulfillment of the requirements for the
degree
of Bachelor of Science (Hons)
Statistical Computing and Operations Research

April 2022

ABSTRACT

IMPROVING HANDWRITTEN DIGIT RECOGNITION USING HYBRID FEATURE SELECTION ALGORITHM

WONG KHYE MUN

In the field of machine learning, handwritten digit recognition was known as one of the crucial problems for pattern recognition and computer vision applications. There were a few applications of handwritten digit recognition, which include recognizing the digits on a utility map, zip code on a postal mail, identifying bank check amount processing and many more. Offline handwritten digits have different traits, such as size, orientation, position, and thickness. Every individual's handwriting was unique in such a way that it would increase the difficulty level of the classification process. High outline similarities between certain digits and overfitting issues for high dimensional data would further affect the computational time and cost. Therefore, many researchers have applied and developed various machine learning algorithms that could efficiently tackle the handwritten digit recognition problem. In this report, the main objective was to obtain the binary classification accuracy of handwritten digit recognition in a Multiple Feature dataset (MFEAT). Minimum Redundancy and Maximum Relevance (mRMR) was used as the primary approach in this report because, being a filter method, it had the greater advantage over a wrapped and embedded method. mRMR could save computational time and effectively considering the relevance of subset features and redundancy within the selected handwritten digit feature. While mRMR was capable of identifying a subset of features that were highly relevant to the targeted classification variable, it still carry the weakness of capturing redundant features along with the algorithm. Support Vector Machine Recursive Feature Elimination (SVM-RFE) as an embedded method, was selected as an alternative approach besides mRMR. SVM-RFE could further select the subset features based on ranking weights criterion, insignificant features with small ranking weights will be removed while retaining only significant features that have greater influence. However, RFE was flawed by the fact that those features selected by RFE were not ranked by importance albeit RFE could effectively eliminate the less important features and exclude redundant features. In view of their respective strength and deficiency, this study combined both these methods and used a support vector machine (SVM) as the underlying classifier anticipating the mRMR to make an excellent complement to the SVM-RFE. The hybrid method was exemplified in a binary classification between digits ‘4’ and ‘9’ from a multiple features dataset. The proposed hybrid method together with two extra predictive models, namely the mRMR and the SVM-RFE, were built for comparison. As a result, four significant features were shortlisted to achieve the highest accuracy which was 100.00% by using the proposed hybrid method. Apart from that, the proposed hybrid method was capable of selecting the highest test accuracy of 99.2% when only one feature was included. The result showed that the hybrid mRMR+SVM-RFE was better than both the sole SVM-mRMR and the sole SVM-RFE approaches in the sense that the hybrid approach achieved higher classification accuracy by using a smaller amount of features.

ACKNOWLEDGEMENT

First and foremost, I would like to express my deep and sincere gratitude to my supervisor, Ms. Chin Fung Yuen from the Department of Physical and Mathematical Science in Universiti Tunku Abdul Rahman. She has been a great supervisor for providing continuous support, sharing her knowledge, and giving valuable advice for my final year project. Without proper guidance from her, I would not have been able to complete the writing for my final year project. I am very much appreciative of her willingness to spare time to guide and encourage me throughout this project.

I would like to take this opportunity to express my gratitude to Universiti Tunku Abdul Rahman as it had provided me with the opportunity to conduct this project. I believe this project has provided me with wonderful and invaluable knowledge towards machine learning. Finally, I would like to express my appreciation towards my parents that always supports and sacrifices for educating and preparing me for my future.

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.

(WONG KHYE MUN)

APPROVAL SHEET

This report entitled **"IMPROVING HANDWRITTEN DIGIT RECOGNITION USING HYBRID FEATURE SELECTION ALGORITHM"** was prepared by **WONG KHYE MUN** and submitted as partial fulfillment of the requirements for the degree of Bachelor of Science (Hons) Statistical Computing and Operations Research at Universiti Tunku Abdul Rahman.

Approved by:

Emmy

____

(Ms, Chin Fung Yuen) Date: 13 April 2022

Supervisor

Department of Physical and Mathematical Science

Faculty of Science

Universiti Tunku Abdul Rahman

FACULTY OF SCIENCE

UNIVERSITI TUNKU ABDUL RAHMAN

Date: 11th APRIL 2022

PERMISSION SHEET

It is hereby certified that **WONG KHYE MUN** (ID No: **19ADB03125**) has completed this final year project entitled "IMPROVING HANDWRITTEN DIGIT RECOGNITION USING HYBRID FEATURE SELECTION ALGORITHM" under the supervision of MS. CHIN FUNG YUEN 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,

____

(WONG KHYE MUN)

TABLE OF CONTENTS

LIST OF TABLES

The following are the list of tables used in this project:

LIST OF FIGURES

The following is the list of figures used in this project:

LIST OF ABBREVIATIONS

The following is the list of abbreviations and notations used in this project:

AUC	**Area Under Curve**
CA	**Classification Accuracy**
CBR	**Correlation Bias Reduction**
CM	**Confusion Matrix**
CV	**Cross-validation**
FN	**False Negative**
FNR	**False Negative Rate**
FP	**False Positive**
FPR	**False Positive Rate**
ICA	**Independent Component Analysis**
LDA	**Linear Discriminant Analysis**
LSA	**Latent Semantic Analysis**
MFEAT	**Multiple Feature Dataset**
MI	**Mutual Information**
MIFS	**Mutual Information Feature Selector**
mRMR	**Minimum Redundancy and Maximum Relevance**
MRRMRR	**Minimum Regularized Redundancy Maximum Robust Relevance**
PCA	**Principal Component Analysis**
PLS	**Partial Least Square**
RF	**Random Forest**
RFE	**Recursive Feature Elimination**

ROC	**Receiver Operating Characteristics**
SVM	**Support Vector Machine**
SVM-RFE	**Support Vector Machine Recursive Feature Elimination**
TN	**True Negative**
TNR	**True Negative Rate**
TP	**True Positive**
TPR	**True Positive Rate**

CHAPTER 1

INTRODUCTION

1.1 Research Background

Handwritten digit recognition has been a vital and significant area in the machine learning field. It serves one of the crucial roles in pattern recognition problems, which aims to ensure that the handwritten characters or digits will get converted into their respective notational form (Kalita, Gautam, Kumar Sahoo, and Kumar, 2019). Handwritten recognition can be divided into two types, which consist of offline and online handwritten recognition. Online handwritten recognition is usually concerned with how well the written digitized text will be recognized. Offline recognition problems will have to deal with a much more complex recognition process before getting digitized. The reason is due to variations in handwriting style, strokes, resemblance in the handwriting outline and other additional noise from different individuals will only increase the difficulty of the handwritten recognition process (Morera, Sánchez, Vélez, and Moreno, 2018).

Generally, handwritten digit recognition will go through several processes like Pre-processing, Segmentation, Dimensionality reduction, and classification. Data conditioning will happen at the Pre-processing stage to ensure that it will qualify to proceed to the next segmentation stage (Singh, Verma, and Chaudhari, 2014). However, in the real world, the number of handwritten digits features are often large, due to the presence of different aspects in an individual's handwriting, resulting in getting high dimensional data. Therefore, dimensionality reduction must play a vital role in reducing the number of handwritten digit features and improving the recognition speed. Dimensionality reduction can be divided into two approaches, namely feature extraction and feature reduction.

Feature extraction can be used in transforming a large set of original data into a low dimension linear or non-linear dataset. The examples of feature extraction are commonly known as Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), and Multidimensional Scaling. According to Cilia, De Stefano, Fontanella, and Scotto di Freca (2019), Feature selection on the other hand will be used to look out for the optimal compact subset of features and remove the irrelevant handwritten digit features from the original dataset to acquire the finest recognition result. A good feature selection of compact subset handwritten digit features is equally important in the classification process and formation of the classifier model. Before determining the suitable classifier model, feature selection will be used as an advantage in reducing the overfitting issue, minimizing the number of handwritten digit features, improving computational time, and enhancing the accuracy of recognition (Singh, Verma, and Chaudhari, 2014).

Feature selection may be a part of dimensionality reduction, but it is different in the data preprocessing or terms of attributes selection. Feature selection will take appropriate measures in decreasing the volume and complexity of the handwritten digit dataset by choosing and retaining attributes that are critical to building a strong classification model, without creating a new combination of features which is performed by using the dimensionality reduction technique (Tadist et al., 2019). In modern days, there are 6 main types of feature selection approaches, but only the 3 basic feature selection techniques will be mentioned in this report, namely Filter, Wrapper, and Embedded method. The filter method is known as a classifier independent and comparatively faster in obtaining relevant features. This technique focused on evaluating each feature criterion, accessing the score relevance between the set of features and the responding variable, and ensuring the best feature subset is chosen and adopted by the learning model algorithm. The filter method will be considered a less expensive approach in using statistical measures such as correlation, distance, fisher score, and mutual information to rank each subset feature and reduce computational time as well as complexity. However, the faster computational speed of the filter method may lead to failure in acquiring the best relevant features, resulting in lower recognition accuracy (Jovic, Brkic, and Bogunovic, 2015).

The wrapper method serves as an alternative feature selection method that measures and searches for the subset of quality features that possess good characteristics for predictive modelling. This is a classifier dependent technique that performs and evaluate the sets of features that are sufficiently good in getting better classification result. Although the wrapper method is slower and more expensive as compared to the filter method, it is more capable than the filter method in terms of getting the subset of features that performed better via the evaluation of the modelling algorithm. According to Jovic, Brkic, and Bogunovic (2015), the wrapper is empirically proven to work better than the filter method in obtaining good quality feature subsets, because the modelling algorithm can evaluate and require only the attributes with better performance to be selected. Lastly, the embedded method integrates the best of both filter and wrapper methods. The reason will be due to the filter method can fulfil the shortcoming of the wrapper method of being slow and cost-inefficient, but the wrapper method, on the other hand, can achieve better recognition results as compared to the inefficiency of the filter method in getting relevant features. Therefore, a hybrid method was introduced to combine the best quality of filter in terms of faster training time and wrapper method in terms of high recognition accuracy. Examples of embedded methods will be LASSO and Random Forest. This report will focus on using a hybrid feature selection method to improve handwritten digit recognition.

1.2 Problem Statement

Handwritten digit recognition has long been a difficult task to resolve in the area of pattern recognition. The first problem is the process of segmentation of handwritten digits, because the presence of distorted characters and high similarities between outlines of certain digits will result in slower performance and redundancy in classification. The next problem in this study is the implementation of one feature selection method alone might not yield an optimal classification accuracy result for handwritten digit recognition. Therefore, a hybrid feature selection method will be proposed to increase the accuracy of classification, improve computational time and reduce high dimension data effectively.

1.3 Research Objectives

The objectives of this study are as follows:

(i) To obtain a compact subset of features using high relevance and low redundancy criterion.

(ii) To improve the performance of the predictive model using Support Vector Machine Recursive Feature Selection. (SVM-RFE)

(iii) To compare the performance of the predictive model with other feature selection methods.

1.4 Significance of the research

The research will propose a hybrid algorithm for classifying the handwritten binary digit features to enable further improvement in the accuracy of handwritten digit recognition. A comparative analysis will be carried out between three models to determine which feature selection algorithms will have the most compact subset selected in building predictive model that gives higher recognition accuracy. mRMR will be used to select the most important subset features in a binary dataset to build the first predictive model. The second model will be developed under the implementation of the SVM-RFE approach in eliminating the irrelevant features and orthogonality theory of correlation techniques, retaining only significant features via the feature ranking approach to achieve higher classification accuracy. The last model will focus on applying the selected features filtered from mRMR into SVM-RFE to further rank the important features and improve the classification accuracy of predictive modelling via hybrid mRMR with the SVM-RFE approach. It is expected that the proposed model using a hybrid method between mRMR with SVM-RFE will show significant improvement in classification accuracy, as compared to the other two methods when building a model that utilizes the existing selected features. Through this research, researchers or practitioners will be able to implement the combination of mRMR and SVM-RFE algorithm to easily identify the issue of highly confused outline commonalities between binary digits.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter is a review of past studies related to dimension reduction, mutual information, Support Vector Machine (SVM), and feature ranking with Recursive Feature Elimination (RFE) will be presented.

2.2 Dimension Reduction

Machine learning data is expanding exponentially throughout these few years and therefore a huge amount of information is gathered to form a high dimensional dataset for further data mining analysis. However, not all of the features in the dataset were crucial as most often there was a great possibility that the received features might be inconsistent, irrelevant, and redundant. Irrelevant data can be defined to be the data that is not significant or bring any influence on the output of the data. Redundant attributes occur when they can take the role of another feature and that may influence the result (Ladha and Deepa, 2011; Pino and Morell, 2013).

Features in a dataset were commonly known as the attributes or variables of data. The presence of high dimensionality data has nevertheless increased the cost and prolonged the time for classification and other data mining analysis. The optimal solution was to use the dimension reduction method as a data preprocessing step in reducing the complication of eliminating the redundant and irrelevant features in high dimensionality data. Dimension reduction method only chose the features which were most relevant and important, as it would later ease machine learning algorithms in performing better classification. By reducing data dimensions and having better data quality, researchers were able to build and design a much more effective predictive model.

Dimension reduction was normally categorized into two parts: feature selection and feature extraction. According to Pino, A., and Morell, C. (2013), feature selection had been an ever-evolving problem due to the rise of big data in recent years. Feature selection aimed to find the smaller number of essential features out of the high dimensionality data, containing the best subset features with the least number of dimensions to contribute and improve the classification accuracy (Kalina, J. and Schlenker, A., 2015). The three main groups of feature selection consist of the filter method, wrapper method, and embedded method. The filter method employed the statistical way of evaluating each subset without the dependence on the classifier. The wrapper method, on the other hand, ~~would be~~ is classifier dependent and it utilized a machine learning algorithm to find out the prediction power gained in the evaluated dataset. Therefore, it would cause computational complexity as the validation process took place for every subset evaluated. The embedded method learned learns the best attributes for improving the accuracy of the predictive model when the model was is set. The embedded method integrated the feature selection process with the model training process, both processes were completed in an optimization process. The mRMR was a filter method and the SVM-RFE was an embedded method.

Feature extraction on the other hand is a process where it transforms the feature from high dimensional space into lower dimensional space by using the fusion of the first and original feature, thus keeping the most relevant information for further classification process (Aziz, Verma and Srivastava, 2017). Some examples of feature extraction methods include Latent Semantic Analysis (LSA), Linear Discriminant Analysis (LDA), Independent Component Analysis (ICA), Partial Least Square (PLS), etc. Among the feature extraction methods, PCA, ICA, and PLS stand out the most as they were the most effective methods in extracting important features (Velliangiri et al., 2019).

In Machine learning, there will be two learning methods to train a classification model: supervised learning method and unsupervised learning method. The supervised learning method was an approach where machine learning algorithms were trained by labelled input data for a certain output. The unsupervised learning method worked the other way round, it was trained to recognize the patterns on its own, without the aid of label data. One the example of an unsupervised learning method will be Principal Component Analysis (PCA), a feature extraction method. The other example would be Partial Least Square (PLS), that considered a widely used method in supervised learning feature extraction. PCA will grab the most important and relevant features, and then replace the original variable in the dataset with the numerical variable known as a principal component. It is also a method that mathematically transforms the data that restrict duplication via covariance and relates the data to a different coordinate system that expands to acquire the greatest variance (Olaolu et al., 2018; Velliangiri et al., 2019). PCA can determine and identify the similarities in the pattern together with the differences in the data, by compressing the determined pattern and reducing the number of dimensions without losing a lot of useful information.

PLS on the other hand is a method that commonly uses the connection of linking explanatory variables blocks with modelling, by using latent variables. Latent variables are unobservable measures, and they can make mathematical by using the observable variables. In this case, PLS will try to search for an uncorrelated linear transformation of the original independent variable that possesses a high covariance with dependent variables. Although the two methods are widely used, they still have their differences in extracting important information. PCA tends to ignore the dependent variable in the dimension reduction process while PLS would incorporate the dependent variable to extract features. Besides that, PLS could perform better than PCA in terms of extracting features, due to it only identifying the number of feature components, as compared to PCA which selects the crucial components (Olaolu et al., 2018).

2.3 Mutual Information and mRMR

There are three basic concepts in information theory, which consist of Entropy, Divergence, and Mutual Information (MI). In the early stage, information theory was utilized within the communication theory background. The early-stage has laid a foundation for the integration of information theory into machine learning. According to Cover and Thomas (2012), the notion of information theory was too wide to be captured in a single definition. Despite that, a quantity can be known as Entropy in the context of a probability distribution, which corresponds with the notion definition of what a measure of information shall be. The concept was later expanded to establish the definition of Mutual Information. Mutual Information is a measurement of the quantity of information that both the random variables have in common and simultaneously. It is an evaluation of statistical independence which possesses two characteristics. The first characteristic of this well-known feature selection method was the ability in detecting the non-linear relationship between two features. MI is unchanging under any given transformation in the invertible feature space, such as rotations and translations, while maintaining the original order details of the feature vector (Vergara and Estevez, 2013).

A part of the MI formula originated from the Entropy (*H*), it was used to measure the uncertainty of random variables that associate with an event's probability distribution. Instinctively, high entropy will usually represent that every event would likely have an equal probability of occurrence. On the other hand, low entropy will represent that each event will experience a different probability of occurrence. The formula of entropy of a random variable, x will start by joining the mass probability of $p(x) = \Pr\{X = x\}$, is measured as:

$$H(x) = -\sum_{i=1}^{n} p(x)\, log_2\big(p(x)\big). \tag{1}$$

Entropy, $H(x)$ can be interpreted as the expected value between the mass probability and the logarithm of the mass probability with base two. The following step would be to let x and y be the two random variables for the joint entropy. The joint mass probability of $p(x, y)$ will be the total uncertainty for both of the random discrete variables. The joint entropy is computed as below:

$$H(\{x, y\}) = -\sum_{i=1}^{n}\sum_{j=1}^{n} p(x, y) \cdot log_2(\, p(x, y)). \qquad (2)$$

Next, MI would be measured by the information shared between the two random discrete variables x and y. It can also be interpreted as how much a random variable talk about another. The complete formula for MI was defined by:

$$I(x, y) = \sum_{i=1}^{n}\sum_{j=1}^{n} p(x, y) \cdot log\left(\frac{(p(x, y)}{p(x). p(y)}\right). \qquad (3)$$

The mutual information will be large if the two random variables have a strong linear relationship, whereas the mutual information between both random variables will be 0 if both are having an independent relationship (Vergara and Estevez, 2013).

Mutual Information has an advantage in gauging the arbitrary dependency between both of the random variables. However, MI became less efficient whenever there is a large dimensionality of the feature input vector, particularly when the number of samples and computational time is taken into consideration (Battiti, 1994). Battiti overcame the issue by adopting the Mutual information feature selector method (MIFS). MIFS is a greedy feature selection algorithm that considers the most relevant feature *k* out from the original set of features, *n,* and also the mutual information concerning the output class. MIFS can solve the weakness in MI by optimizing the information about the class and subtracting the quantity proportional to MI with the previously selected feature.

Later, another study by Kwak and Choi (2002) found out that there was still a limitation in the MIFS proposed by Battiti (1994). They instead proposed a better solution method known as MIFS-U. MIFS-U is better in terms of obtaining a more precise estimation between input features and output class in MI, as compared to MIFS. Despite MIFS-U being a better feature selection algorithm than MIFS, there were still some limitations between these two methods. The left-hand side and right-hand side of the formula in MIFS-U conflict with each other and are unable to make any comparison, which led to a feature selection algorithm to choose the redundant features that eventually will affect the efficiency of the classification (Estevez et al., 2009). The redundancy issue in MIFS-U was then minimized by using a method called Minimum redundancy and maximum relevance (mRMR) proposed by Peng et al. (2005). The maximal relevance of mutual information will enhance the minimum redundancy criterion to become more representative of the target features. However, it was also claimed that mRMR might select a high relevant feature which also caused high redundancy at the same time because the selection was based on the difference between relevancy and redundancy (Ding and Peng, 2005). Aside from that, the author also mentioned that when additional features that contain noise were included aside from mRMR features would lead to fluctuation of classification error, but the problem did not solve as it was beyond the scope of the report.

Research on improvements regarding mRMR has been done by a few researchers, including Jan Kalina and Anna Schlenker that used Minimum Regularized Redundancy Maximum Robust Relevance (MRRMRR) as a novel optimization problem to solve the redundancy, outlier, and noise issue of mRMR. However, the method still possesses some drawbacks such as high computational complexity, lower stability, and classification ability due to the smaller number of selected features (Kalina, J. and Schlenker, A., 2015). Regarding the past studies, this study would continue to focus on tackling the existing redundancy and fluctuation issue of mRMR together with another feature selection method to choose a less optimal subset of features with high accuracy.

2.4 Support Vector Machine (SVM)

In machine learning, the Support Vector Machine (SVM) is was an effective machine learning classifier which originated from the statistical learning theory founded by Vladmir Vapnik in 2005. He was the first person to propose the idea of adopting SVM with the Structural Risk Minimisation Principle (SRM), because SVM has the overall capability of maximizing a model's general ability while SRM was able to minimize the decision bound of the model while ensuring the generalization error of the model greatly reduced. It was also a supervised learning technique that was commonly used due to its superiority and better generalization capability in delivering much better performance at reducing classification error and improving classification accuracy as compared to other supervised learning data classification methods. SVM was also well known for its overall strong ability in classification and regression function, especially when applied to kernel machine learning models. Despite that, SVM will be mostly used in classification models rather than regression, due to the efficiency of the classification method in handling a broad range of datasets (Cervantes et al., 2020). SVM has drawn attention from machine learning communities, pattern recognition, and data mining over the past few years, due to its high discriminative power and obtaining a good optimal solution record (Durgesh et al., 2010).

As compared to other supervised learning techniques, SVM tends to stand out among the rest due to its excellent classification skills in dealing with a dataset with a small number of data inputs, high in dimension, and nonlinear problems (Cervantes et al., 2020). SVM can easily solve a nonlinear classification problem by applying a nonlinear mapping function that can map the original nonlinear data with low-dimensional space into a higher dimensional feature space. From there, the high dimensional feature space with nonlinear and inseparable problems can be transformed into a linear and separable classification problem, making SVM an excellent and promising supervised learning technique for creating an optimal hyperplane. On top of that, SVM exhibits the ability to classify and predict unseen samples with great accuracy (Kari et al., 2018). Moreover, SVM has been widely used as a robust tool in tackling the issue of binary classification (Cervantes et al., 2020).

2.5 Feature ranking with Recursive Feature Elimination (RFE)

In supervised learning, a predictive model often oversees the data or features inside a dataset, and jeopardizing its ability to generalize well, this condition will be known as overfitting. Due to the existence of an overfitting problem, there lies inconsistency of accuracy in the training set and testing set, because the model which is over-fitted might have perfect accuracy on the training set, but it would have a difficult time handling the information about a feature in the testing set. Apart from that, overfitting happens when a predictive model includes the noise in a limited size training data set instead of focusing on learning the meaning behind the data features (Ying et al., 2019). The Recursive Feature Elimination (RFE) method can therefore effectively cut down the overfitting problem, eliminating uncorrelated noise data and irrelevant data.

RFE is an embedded feature selection, which was first introduced by Guyon et al. (2002), recursively eliminates the features which are irrelevant with a small feature weight or a subset of features that has a much lower position or rank. In every iteration, RFE orderly discards the worst feature that affects the classification accuracy to drop after building the predictive modelling. The integration could measure the importance of features instead of classification accuracy and eliminate the feature that has the least importance to build the model (Jeon et al., 2020). RFE can also work well with other classifier approaches, such as Random Forest (RF). The reason will be due to RF has its own built-in feature evaluation method.

RFE will start by having a classifier to train the training dataset and each of the original set of features will have weights, w assigned to them accordingly. Next, the weight of the features would be sorted from the largest to the smallest, the feature which had a smaller weight value shall be eliminated from the list of surviving features. The process is ~~was~~ then repeated for each iteration until all of the features which have smaller ranking criteria are ~~were~~ removed so that no features are left for the classifier to train anymore. At the end of the iteration, the desired number of selected features would be obtained using RFE as a feature ranking mechanism.

Besides the discovery of the RFE approach, Guyon et al. (2002) continued their work and proposed an embedded feature selection method that used the RFE approach to integrate with the Support Vector Machine (SVM) to form SVM-RFE, which will be used in the next chapter of this study. SVM-RFE was considered an approach or an application of RFE that will be able to use the criteria derived from coefficients of SVM models to obtain features, and then recursively discard features that have small criteria or weights. However, SVM-RFE also raised a time-consuming issue when selecting candidate features with high accuracy, which became a topic of interest for many researchers.

Later, Zhou et al. (2009) discovered that although SVM-RFE was able to build a predictive model with high accuracy, the ranking method used in SVM-RFE might not work well in selecting the first most relevant feature. Consequently, the predictive model performed well only when many features are selected but would give low accuracy when only one or two features are included. Another research from Ke Yan and David Zhang ~~in 2015~~ Ke and Zhang (2015) also mentioned that SVM-RFE also faced a high correlation bias problem when some of the candidate features were highly correlated and influenced, which will lead to underestimation in the ranking of importance for significant features. Therefore, they proposed the integration of Correlation Bias Reduction into SVM-RFE (SVM-RFE+CBR) to reduce the possible bias in Linear SVM-RFE and non-linear SVM-RFE and improve the ranking and stability of feature selection results (Yan and Zhang, 2015).

Though there were past studies that have addressed and overcome the limitations of mRMR and SVM-RFE, both feature selection methods still suffer from three drawbacks that cannot be solved easily: redundancy issues, overfitting, and ranking problem. In this study, attempts were made to hybridize the SVM-RFE method with the mRMR method hoping to bring improvement in getting the most relevant features selected. We believe that using the highly relevant features shortlisted by the mRMR method prior to the linear kernel SVM-RFE will benefit its ranking. Besides that, we also hope that the shortlisted features by mRMR can save computational time for SVM-RFE to re-rank the shortlisted features. The proposed hybrid method would be simulated and experimented with to solve the past limitations of SVM-RFE and mRMR. The idea ~~was tested~~ would be tested on the binary classification between digits ‘4’ and ‘9’.

(if you are planning to do it.
then write would be tested).

CHAPTER 3

METHODOLOGY

3.1 Minimum Redundancy and Maximum Relevance (mRMR)

In the digital era, big data is ever-evolving daily and continues to expand into a higher dimension. The number of features or attributes present in the data is increasing exponentially, but not all of them are essential and contribute to the process of machine learning. The presence of noisy and redundant features will only result in rising the computational cost and increase the complexity of classification. Therefore, it is crucial to have a feature selection process to choose the most relevant feature and eliminate the redundant attributes. The filter method as one of the three main feature selection method, have the advantage due to its generalizability in a wide range of machine learning models and possesses more efficiency in saving computational time (Zhao et al., 2019). Due to its overall capability of producing higher effectiveness in reducing redundant features while maintaining the relevant features for building effective predictive modelling, mRMR as one of the important members of the filter method will be introduced in this section.

In the early stage, mRMR was considered a powerful filter method that was mostly used in handling gene classification. As years go by, the usage of mRMR in other classification fields is becoming more and more frequent, this was mainly due to its capability in providing excellent trade-offs between computational stability and classification accuracy. mRMR is an algorithm that will rank the importance of a set of attributes about their relevance to the target, but at the same time, it can eliminate the redundancy among features to save time for effective classification. Mutual Information, $I(x, y)$ was used in mRMR to find the maximum dependency within a set of attributes and its given label class. The mutual information formula developed by Ding et al. (2005) for the categorical and discrete variable is computed as follows:

$$I(x, y) = \sum_{y} \sum_{x} p(x, y) \cdot log \left(\frac{(p(x, y)}{p(x). p(y))} \right) \tag{4}$$

For the continuous variable, the mutual information is defined as follows:

$$I(x, y) = \iint p(x, y) \cdot log \left(\frac{(p(x, y)}{p(x). p(y))} \right) dydx \tag{5}$$

Both variables have joint mass probability, which is represented by $p(x, y)$ for the two features of x and y. The $p(x)$ is the marginal probability for variable x and y will have the marginal probability of $p(y)$. There are two stages for mRMR to choose the optimal subset of features. The first step required in mRMR is to apply the maximum relevance, which will be used to select a set of features (S) with the k attributes $\{x_i\}$ that contain the most relevant information to their class label, h (Ding et al., 2005). The relevance formula is derived as follows:

$$max\ V_i = \frac{1}{|S|} \sum_{i \in S} I(h, i) \tag{6}$$

where $|S|$ is the number of features in the set S.

The second step is to minimize the redundancy of features because mutual information will measure the quantity of similarity between the information for both of the random variables. The reason for applying minimum redundancy will be due to the chosen feature from maximum relevance may contain a high number of redundancy features, which will not provide any useful information for the classification model (Peng et al., 2005). The minimum redundancy concept is to choose the features that have the mutually dissimilar trait. Suppose S represent the set of features and the redundancy formula will apply to pick the mutually exclusive features:

$$min\ W_i = \frac{1}{|S|^2} \sum_{i,j \in S} I(i,j) \tag{7}$$

Eventually, a set of features of mRMR will be acquired based on the combination of equations (6) and (7), which is known as the "minimum-redundancy-maximum relevance" criterion. The operator of $max\ \Phi(V_i, W_i)$ will be combined into a single selection criterion. The formula for the mRMR selection criterion will be derived as:

$$max\ \Phi(V_i, W_i), \Phi = V_i - W_i. \tag{8}$$

$$\begin{matrix} max \\ i \in S \end{matrix} f^{mRMR}(X_i) = I(i,h) - \frac{1}{|S|} \sum_{j \in S} I(i,j) \tag{9}$$

Both equations (8) and (9) are alike in respect of their application in ranking the order of subset features. The feature that fulfils the condition of having high relevance will be retained and the ones that are irrelevant or redundant shall be eliminated from the list of features. The retained features will obtain a higher score in terms of importance and they will be assigned to the list to form the best subset feature, thanks to the implementation of the mRMR feature selection algorithm.

3.2 Support Vector Machine Recursive Feature Elimination (SVM-RFE)

In general, it is unavoidable that high similarities between features may cause overfitting issues, especially when researchers are often dealing with high dimensional data that will further affect the computational time and cost. An embedded method was introduced by Guyon et al. to solve the overfitting issue, that is by using Support Vector Machine Recursive Feature Elimination (SVM-RFE). SVM-RFE is a feature selection method that utilizes the criteria acquired from the SVM's coefficient to choose selected features and recursively remove features that contain fewer criteria or weight, in a backward elimination manner. SVM-RFE also does not rely on the cross-validation accuracy to determine the relevant features from the training data, due to its advantage of fully utilizing the training data, being less inclined towards overfitting, and performing much faster when the algorithm is given a training data that contains thousands of features (Yan et al., 2015).

Overall, the selection of relevant features for SVM-RFE can be divided into three stages. The first stage will be the input data will be inserted into the classifier SVM for classification. The second stage will involve the process of calculation for all of the features in terms of ranking weights. The last stage will include the deletion of features that have a smaller ranking weight (Huang et al., 2014). SVM-RFE can also be divided into two categories, one is linear SVM-RFE which is preferable when the optimal function is linear and the other will be non-linear SVM-RFE which works directly opposite of linear SVM-RFE. In this study, linear SVM-RFE will be used for its simplicity in segregating the high dimensional data into classes to better train the classification model. Suppose X_0 will be the input training data, y will be the class label of X_0.

$$X_0 = [x_{1,}x_{2,} \dots, x_k]^T$$

$$y = [y_{1,}y_{2,} \dots, y_k]^T$$

The SVM-RFE will start by obtaining the subset of surviving features, s and the feature will be ranked accordingly in a list, r. The process will be repeated until all the feature ranked list is obtained. The remaining sorted feature will be kept for the SVM classifier to train the data (Guyon et al., 2002). The formula of SVM being a training data classifier can be defined as:

$$\alpha = SVM - train(X, y)$$

The following step will be taken over by Recursive Feature Elimination (RFE) to obtain the ranking score and ranking weight for each feature. The ranking score of the trained features will be computed according to the weight vector, w. Suppose a_k is the Lagrange Multiplier that is involved in maximizing the margin of separation of class labels, k is the feature and n is the number of features.

$$w = \sum_{k=1}^{n} a_k\, y_k x_k \qquad (10)$$

Next, the ranking criterion, C_k for the surviving feature will be computed by obtaining the square of the k-th element of the weight vector, w.

$$C_k = w_k^2 \ , k = 1,2,3, \dots \tag{11}$$

The feature that has the smallest ranking criterion will be identified and eventually eliminated due to its insignificance to be included in the classification. For each of the loops or iteration of RFE, a linear SVM model will be trained, and the surviving features will be kept for the next iteration. The process keeps on repeating until all of the features are discarded, and then they will be sorted according to the removal sequence. The latter a feature being discard, it means that the more significant that feature is, it will be given the higher rank. The process eventually produces an optimal feature subset (Yan et al., 2015).

According to Jeon et al. (2020), the overall process of feature-importance-based RFE will be shown in Figure 2.1. Suppose N represents the number of features, FS represents feature selection and FR represents feature ranking.

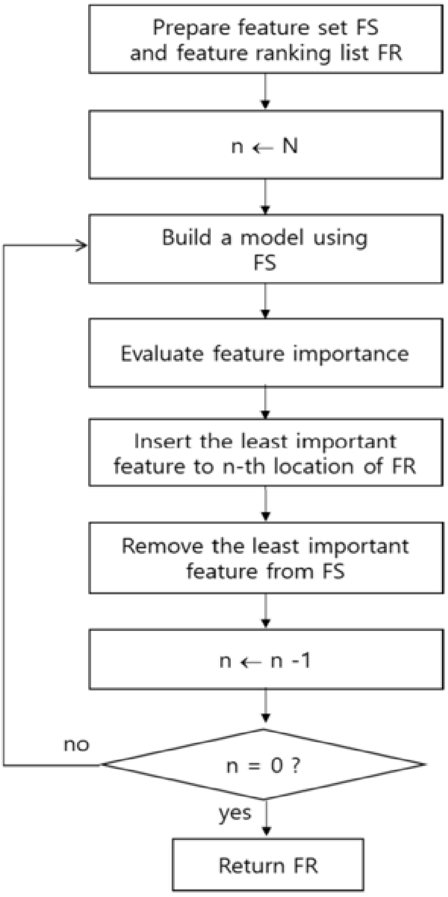
Prepare feature set FS and feature ranking list FR
n ← N
Build a model using FS
Evaluate feature importance
Insert the least important feature to n-th location of FR
Remove the least important feature from FS
n ← n -1
no
n = 0 ?
yes
Return FR

Figure 3.1: Overall process of feature-importance-based RFE (Source: Jeon et al., 2020)

3.3 Proposed Hybrid Method

According to the No Free Lunch Theorem proposed by David Wolpert and William Macready in 1996, a single machine learning algorithm that searches for an optimal solution for every problem does not exist and may not necessarily be superior to any other machine learning algorithms (Wolpert, 1996). Therefore, this has given rise to the birth of hybrid machine learning algorithms that integrate the benefits of different machine learning techniques and diminish the disadvantage caused by the individual algorithm that can take advantage of various generalization mechanisms which could deteriorate the classification model. In this study, the proposed hybrid method aims to create a predictive model that consists of the combination of utilizing the mRMR algorithm together with the SVM-RFE algorithm to produce better classification by just employing a few most significant features.

The hybrid method will start by having the dataset split into a training set and a test set according to the ratio of 7:3. The same training set and test set were also used by the mRMR algorithm and SVM-RFE algorithm to select an optimal subset of features, in turn to make a performance comparison with the hybrid method. After the splitting process, mRMR was applied to rank the features according to equation (8), the shortlisted features contained the most relevant features. In this study, the number of features that arbitrarily took was $k = 15$ while implementing the mRMR algorithm. It is important to take note that the number of features to keep k have to be preset by the researcher. The process of obtaining the shortlisted number of features reduced the high dimensional data to a smaller data set. The weight, w of each feature from the shortlisted features was calculated. The weights of the features according to equation (10) were then sorted in descending order and the feature having a smaller weight value were eliminated from the list of surviving features. A linear SVM model will be trained for each iteration and the process was repeated until all the features with smaller ranking criteria obtained in equation (11) were removed such that no more features were left for training. At the end of the iteration, the desired number of selected features will be obtained using RFE as a feature ranking mechanism. Figure 3.2 shows the flowchart of the proposed hybrid method.

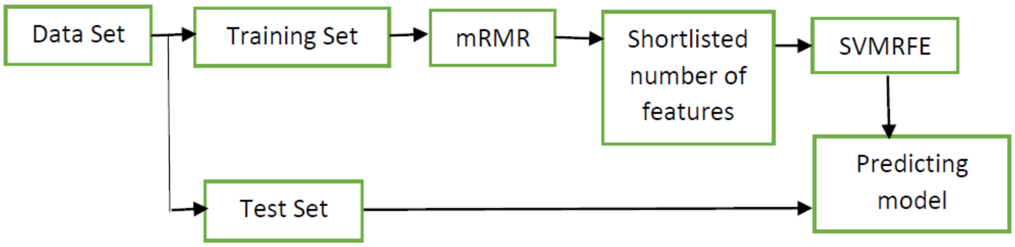

Figure 3.2: Flowchart of the proposed hybrid method

The underlying reason for combining both algorithms was due to the individual's limitations in the process of obtaining the optimal subset of features. mRMR has shown to be good at selecting the most relevant features, but it has also included some redundant features in the process. On the other hand, SVM-RFE as an embedded method has no doubt caused high computational cost and time to obtain high classification accuracy. Therefore, mRMR as a filter method that requires only less computational time can first shortlist the number of features to cut down the computational time of SVM-RFE, while SVM-RFE can solve the redundant issue faced by mRMR, fulfilling the purpose of proposing the hybrid method in this study.

3.4 Cross-Validation

Cross-validation, commonly known as CV is a resampling technique commonly used in measuring the performance of machine learning models. The main approach of cross-validation is to offer an unbiased estimate of the performance of machine learning models. Cross-validation is widely used in the small dataset due to its general capability of obtaining a more accurate estimate in dealing with generalization errors. In this study, k-fold cross-validation is adopted to evaluate the performance of each classification model.

The word “flow” means the number of resulting subsets. The process k-fold cross-validation will start by having the variable k to be the number of groups that the data points are randomly distributed into. In other words, the dataset is divided into k equal parts. At each iteration, a group of k will be randomly selected to be used as the test set or known as the validation set. Then, the remaining k-1 groups will be used as the training set to train the predictive model. The k-fold cross-validation process will be iterated k times such that each of the groups can be chosen once to be used in the test set. Lastly, the cross-validated performance will be the average of k performance measurement on the estimate of k test set. (Maleki et al., 2020). In other words, the overall performance of the model is measured by calculating the average test error value across the k iterations.

Therefore, 10-folds cross-validation will be used in this study as the variance of the model performance will be greatly reduced with the model being validated k times to produce more reliable results for machine learning models, as shown in Figure 3.3. It is worth taking note that the researcher can self-determined and evaluate the k parameter before starting the cross-validation process.

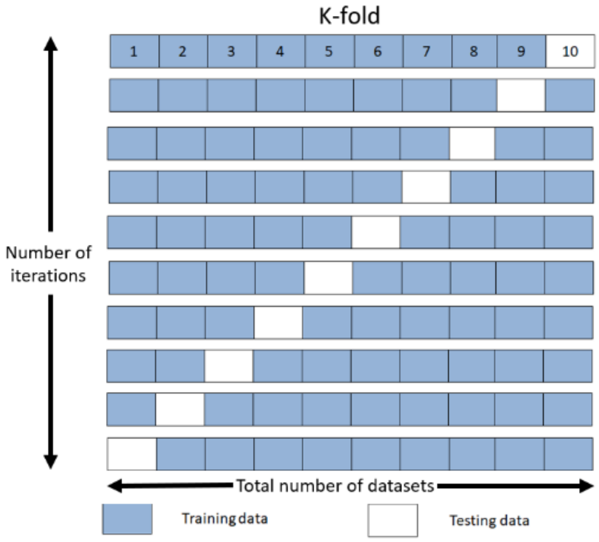

Figure 3.3: 10-fold cross-validation

3.5 Performance Evaluation Measure

To evaluate the performance of the classification model, classification accuracy, confusion matrix, and area under the curve will be employed.

3.5.1 Classification accuracy

Classification accuracy (CA) is metric used in measuring the classification model (Arumugam et al., 2021). The accuracy of a model is also meant by the number of correct predictions gained out from the total number of predictions. The accuracy can be calculated by:

$$Accuracy = \frac{Number\ of\ correct\ predictions}{Total\ number\ of\ predictions} \tag{12}$$

For binary classification problems, the accuracy can be expressed in terms of positives and negatives, as shown by:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{13}$$

where TP represents True Positive, TN represents True Negative, FP represents False Positive, and FN represents False Negative.

3.5.2 Confusion Matrix (CM)

A confusion matrix is a summary table that displays the prediction performance of a classification model. The table is often useful in showing not only the confusion of prediction done by the classification model but also the type of errors made by the classifier.

		Predicted Class	
		Positive (Digit ‘4’)	Negative (Digit ‘9’)
True Class	Positive (Digit ‘4’)	True Positive (TP)	False Negative (FN)
	Negative (Digit ‘9’)	False Positive (FP)	True Negative (TN)

Table 3.1: Confusion Matrix

The confusion matrix in Table 3.1 is suitable to be used in a binary classification problem, this is due to it can let researchers find out different types of misclassification predictions broken down by each class for a supervised learning model. In this study, the True Positive (TP) will represent the number of samples that are correctly classified as the handwritten digit ‘4’. Similarly, True Negative (TN) will represent the number of samples that are correctly classified as the handwritten digit of ‘9’. On the other hand, False Positive (FP) can be described as the number of samples with the handwritten digit ‘9’ misclassified as the handwritten digit ‘4’ whereas False Negative (FN) can be understood as the number of samples with the handwritten digit ‘4’ misclassified as the handwritten digit ‘9’.

3.5.3 Receiver Operating Characteristic Curve (ROC)

Besides using a confusion matrix to explore the prediction performance of a classification model, According to Arumugam et al. (2021), the ROC curve serves as an alternative to display the classification performance on a binary classification problem. There are two parameters inside the curve, which are the True Positive Rate (TPR) and False Positive Rate (FPR). The area under the ROC curve (AUC) is a measurement that can be used to illustrate the whole two-dimensional area underneath the entire ROC curve. AUC will have a range value falling from 0 to 1. True Positive Rate (TPR) or known as the recall is given by:

$$True\ Positive\ Rate\ (TPR) = \frac{TP}{TP + FN} \tag{14}$$

True Negative Rate (TNR) can be known as specificity, is defined as:

$$True\ Negative\ Rate\ (TNR) = \frac{TN}{TN + FP} \tag{15}$$

False Positive Rate (FPR) is given by equation (16):

$$False\ Positive\ Rate\ (FPR) = \frac{FP}{FP + TN} \tag{16}$$

False Positive Rate (FPR) is defined by equation (17):

$$False\ Negative\ Rate\ (FNR) = \frac{FN}{FN + TP} \tag{17}$$

Along with that, the Precision of a classification model can be calculated as:

$$Precision = \frac{TP}{TP + FP} \tag{18}$$

3.6 Experimental Setup

Throughout the whole study, the binary handwritten digit of ‘4’ and ‘9’ from the MFEAT dataset was applied to find out the effectiveness of the proposed hybrid classification algorithm. All the experimental procedures were conducted using MATLAB software. Before the start of the experiment, 400 out of 2000 samples were extracted from the dataset for conducting binary handwritten digit classification. The dataset was then randomly split into two sections, 70% for the training set and 30% for the test set. The training set was used in two situations, to be used for performing feature selection and to train in building the three classification models with the help of supervised learning. The test set on the other hand was used to validate the predictive model’s accuracy. To indicate the superiority of the proposed hybrid method, two extra predictive models, namely the mRMR and the SVM-RFE, were built for comparison. The experiment began with the mRMR method, which was used to select the most relevant features from the dataset. SVM-RFE method was then used to build a predictive model with high accuracy. The proposed hybrid method aimed to combine the advantages and overcome the shortage of the mRMR and the SVM-RFE methods. The three models were trained by using SVM classifier with 10-fold cross-validation to obtain cross-validation accuracy. After that, the remaining 30% of the test set was used to validate the label prediction of the three classification models. In addition, the classifier’s performance on the three models could also be evaluated and compared by using classification accuracy, confusion matrix, ROC, and AUC.

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 Dataset

The dataset used in this paper was the Multiple Feature (MFEAT) dataset. It was a dataset that consists of features of handwritten digits (0-9) extracted from a collection of Dutch utility maps. The rows represented the number of samples present in the dataset and the columns represented the handwritten digits. This dataset contained a total of 649 features and 2000 samples. However, only 400 samples were used in this study because the main focus is to obtain the binary classification between 4 and 9 instead of the multiclass classification covering digits 0 to 9. Digits 4 and 9 were selected due to the occurrence of the misleading contour of handwriting and the high resemblance between these two digits.

4.2 Baseline accuracy of full features without any feature selection algorithm

First and foremost, all of the features containing 400 samples will be included in the classification model for training and SVM classifier will be used to obtain the baseline accuracy for all of the 649 features. The main objective for including all of the features into the model training was to know how well the SVM classifier could classify the features without applying any feature selection algorithms. In this study, SVM is chosen as the main classifier to focus on, due to the training data used in this study will be much smaller than the number of features ($k > n$). On top of that, SVM can handle outlier problems better than other data classifiers. The baseline accuracy for all of the 649 features using Support Vector Machine (SVM) with 10-fold cross-validation is 99.3%. The test accuracy for the full features with the SVM classifier is 99.2%. However, without this dataset going through any feature selection, there is a chance that the dataset is still under high complexity with full features in it, which will result in misleading classification accuracy. Therefore the following sections will show the performance of different feature selection algorithms in comparison to the baseline accuracy.

4.3 Performance using the mRMR algorithm

High dimensional data no doubt will cause a whole load of problems towards classification accuracy. A large number of features will only create unnecessary noise and affect the performance of predictive modelling. Therefore, feature selection such as mRMR will be used to select only features that are relevant, nonredundant, and consistent. By that, it can decrease the feature space and hence allow the more useful features to build an effective model.

The experiment will first begin by applying the training dataset to the mRMR algorithm to rank and find the best subset features. The top 15 ranked features shown in Table 4.1 were chosen by mRMR from the dataset to build a classification model. Cross-validation accuracy for the top 15 shortlisted features will then be obtained iteratively, starting from $k = 1$, where k refers to the number of features. After performing cross-validation accuracy for the classification model, test accuracy will be used to validate the result for the top 15 shortlisted features. Figures 4.1 and 4.2 will show the cross-validation accuracy result and test accuracy result for the top 15 shortlisted features selected by mRMR respectively.

Based on Figure 4.1, it is observable that the cross-validation accuracy of selected features using the mRMR algorithm improves to 99.6% as compared to 99.3% of baseline accuracy for full features. In contrast, the accuracy for the test dataset in Figure 4.2 reached a peak at 100% accuracy when $k = 7$. The model required the first seven features, which consist of $f_{649}, f_{173}, f_{201}, f_{105}, f_{129}, f_{141}, f_{185}$ to reach the maximum test accuracy of 100%. By comparison of the two results with the baseline accuracy, it has been shown that mRMR algorithm not only improves accuracy but also reduces the number of irrelevant features at the same time. Although the application of mRMR significantly improves the cross-validation accuracy result, it is still unable to reach 100% by using the top 15 shortlisted features. In addition, accuracy curves for mRMR in both Figures 4.1 and 4.2 showed up-down fluctuation when more features were included. This revealed the fact that while the mRMR method selects the most relevant features, it also includes some redundant features during the process. Hence, the SVM-RFE feature selection algorithm in the following section will be used as a comparison in terms of performance besides its ability to solve the redundant and overfitting issue.

	Ranked Features
mRMR algorithm	$f_{649}, f_{173}, f_{201}, f_{105}, f_{129}, f_{141}, f_{185}, f_{257}, f_{189}, f_{209}, f_{261}, f_{260}, f_{233}, f_{269}, f_{262}$

Table 4.1: List of top 15 shortlisted features selected by mRMR algorithm

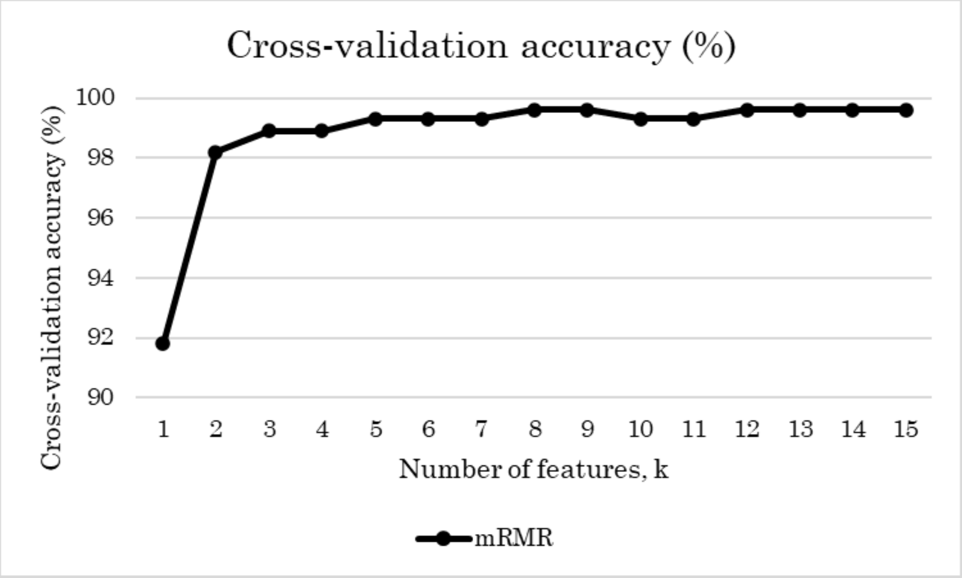

Figure 4.1: Cross-validation accuracy of top 15 shortlisted features selected by mRMR

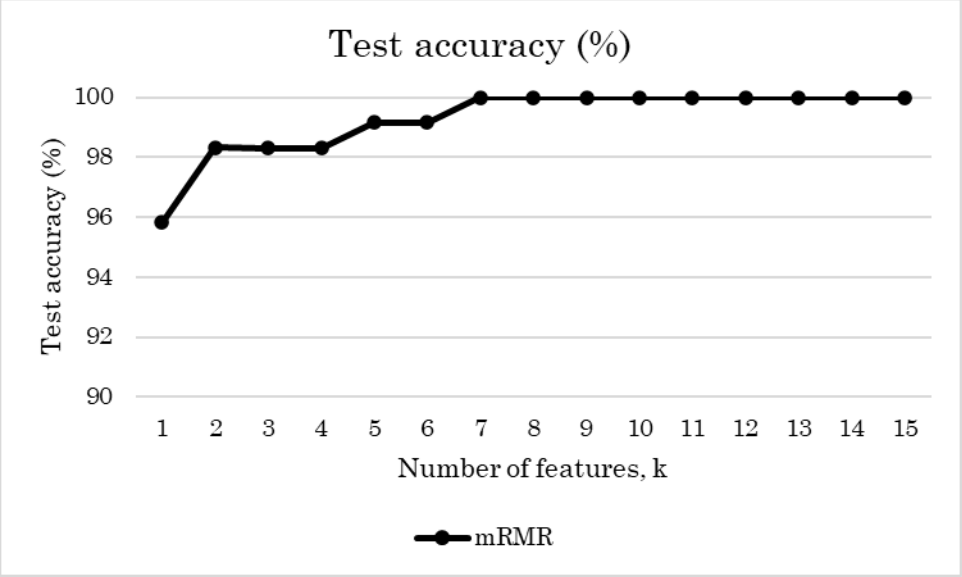

Figure 4.2: Test accuracy of top 15 shortlisted features selected by mRMR

4.4 Performance using the SVM-RFE algorithm

In machine learning, it is often the case that the presence of redundant features inside a classification model will result in higher complexity and affect the classification accuracy of a model. When a classification model is complex, the chances of getting a larger classification error will increase as well, thus leading to a bigger risk in getting overfitting problems. SVM-RFE as an embedded method is a technique where it uses the regularization of SVM to avoid overfitting problems and RFE to solve the redundant issue by selecting relevant and significant features with much higher classification accuracy.

Firstly, the training dataset will be input into the SVM-RFE algorithm to find the relevant, non-redundant, and significant features to be ranked. The top 15 ranked features shortlisted by SVM-RFE as shown in table 4.2 were used to build the second classification model. Cross-validation accuracy for the top 15 shortlisted features will be obtained iteratively. Subsequently, test accuracy will be used to validate the cross-validation result for the top 15 shortlisted features. Figures 4.3 and 4.4 will show the cross-validation accuracy result and test accuracy result for the top 15 shortlisted features selected by SVM-RFE respectively.

	Ranked Features
SVM-RFE algorithm	$f_{77}, f_{201}, f_{257}, f_{601}, f_{93}, f_{269}, f_{245}, f_{113}, f_{177}, f_{138}, f_{140}, f_{164}, f_{234}, f_{141}, f_{128}$

Table 4.2: List of top 15 shortlisted features selected by SVM-RFE algorithm

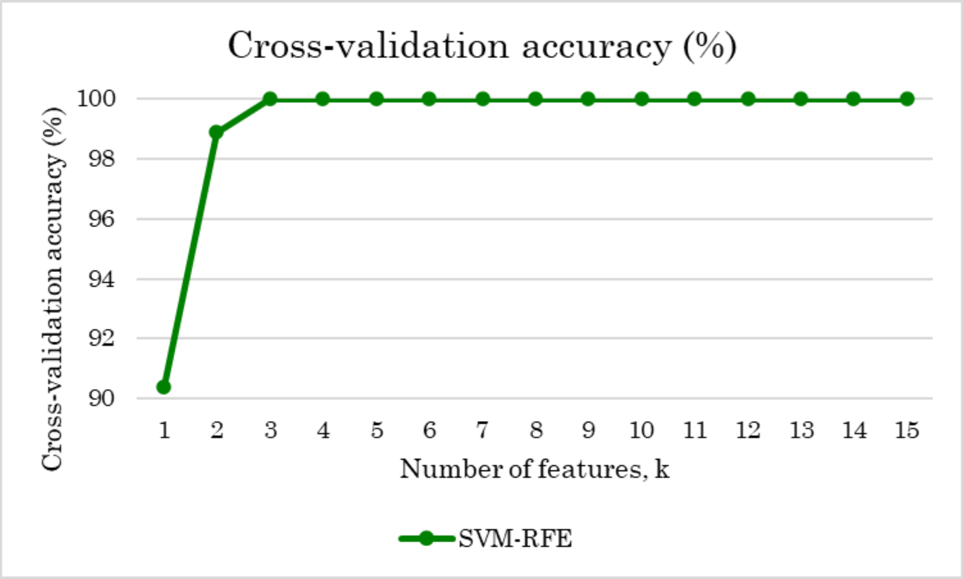

Figure 4.3: Cross-validation accuracy of top 15 shortlisted features selected by SVM-RFE

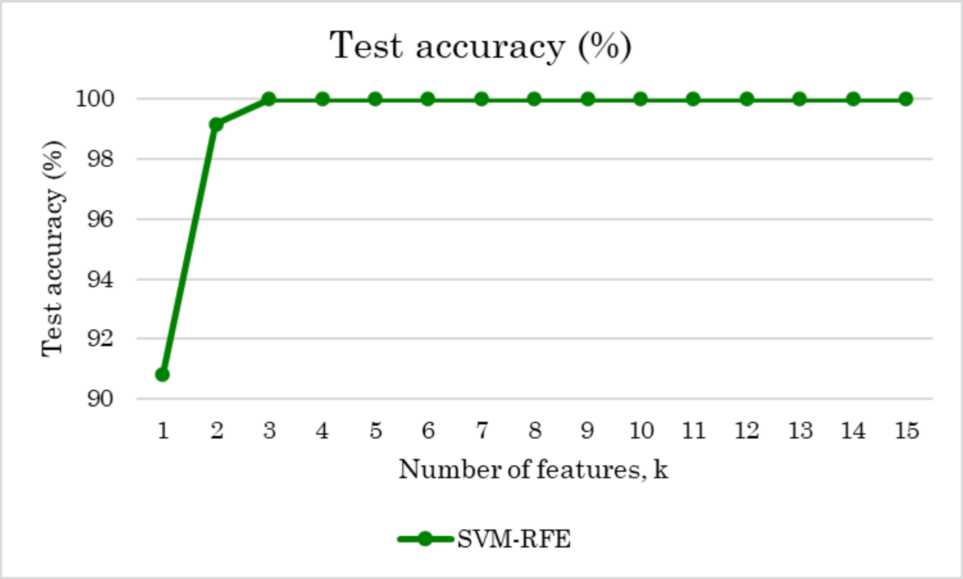

Figure 4.4: Test accuracy of top 15 shortlisted features selected by SVM-RFE

As shown in Figures 4.3 and 4.4, the cross-validation accuracy and test accuracy using SVM-RFE have significantly improved, and much lesser features were needed to achieve 100% accuracy. The model only required the first three features, which consist of f_{77}, f_{201}, f_{257} to reach the optimal accuracy of 100%. This could explain that the first three selected features have a strong relationship with their class labels. After the third feature hit 100% in cross-validation accuracy and test accuracy, the level of accuracy was maintained until the 15th selected feature. The performance of the SVM-RFE has also shown its consistency when more features were included in the predictive model. At the same time, SVM-RFE was able to achieve 100% classification accuracy when compared to the baseline accuracy of 99.3%.

A comparison between the performance result of mRMR from the previous section and SVM-RFE was done, as shown in Figures 4.5 and 4.6. The black line plotted in the line graph indicates the accuracy of mRMR while the green line represents the accuracy of SVM-RFE.

As shown in Figure 4.5, the cross-validation accuracy of SVM-RFE performs much better than mRMR. SVM-RFE only needed the first three features to achieve 100% accuracy while mRMR needed the first eight features to achieve its peak accuracy of 99.6%. Besides that, the overall cross-validation accuracy of SVM-RFE was much more consistent as compared to mRMR. This was because the the SVM-RFE classification model maintained a good accuracy record of 100% while the mRMR classification model showed signs of fluctuation in accuracy throughout the first seven features before it reached the maximum cross-validation accuracy of 99.6%.

Meanwhile, the overall test accuracy for SVM-RFE as shown in Figure 4.6 was much higher and better than mRMR. For SVM-RFE, the model only needed the first three features to reach the maximum test accuracy of 100% while mRMR can only achieve the same test accuracy when it reaches the seventh feature. This has shown the ability of SVM-RFE at achieving high accuracy faster with lesser features needed as compared to mRMR. Consistency in test accuracy continued to be shown by SVM-RFE as compared to mRMR, with a consistent record of 100% test accuracy from $k = 3$ until $k = 15$.

By comparing the two classification models with both feature selection methods, it is almost certain that SVM-RFE performs better than mRMR at achieving higher cross-validation accuracy and test accuracy with lesser features. However, if given the condition where only a single feature is included, SVM-RFE gave the lowest accuracy as compared to mRMR. This has no doubt posed an issue where the first feature from the SVM-RFE selected feature subset was not necessarily the most significant one. Besides, SVM-RFE as an embedded method was ineffective when it comes to long computational time in selecting features with high accuracy. Therefore, the next section will discuss the performance when using the proposed hybrid method to bring improvements in terms of accuracy, efficiency, and relevance.

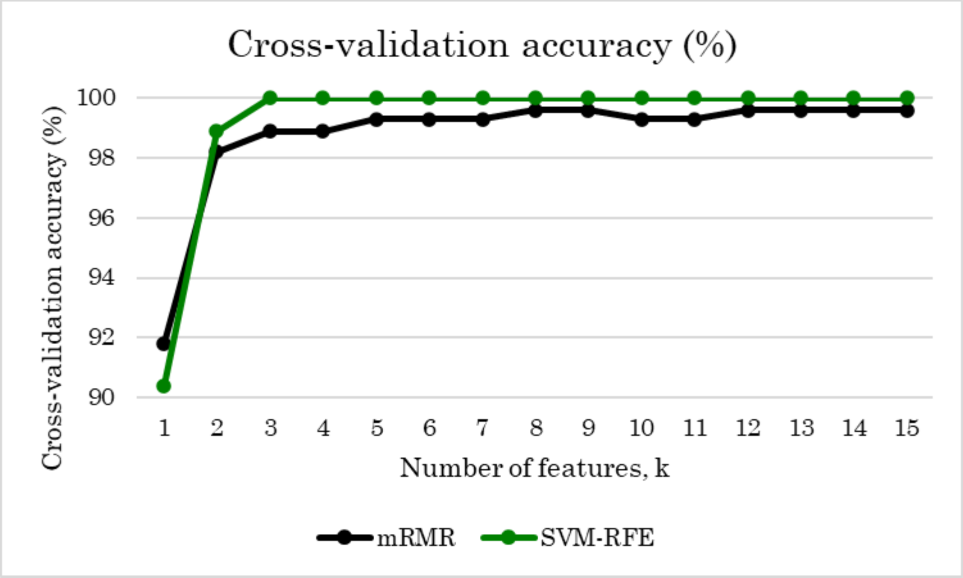

Figure 4.5: Comparison of the Cross-validation accuracy in terms of selected features by mRMR and SVM-RFE.

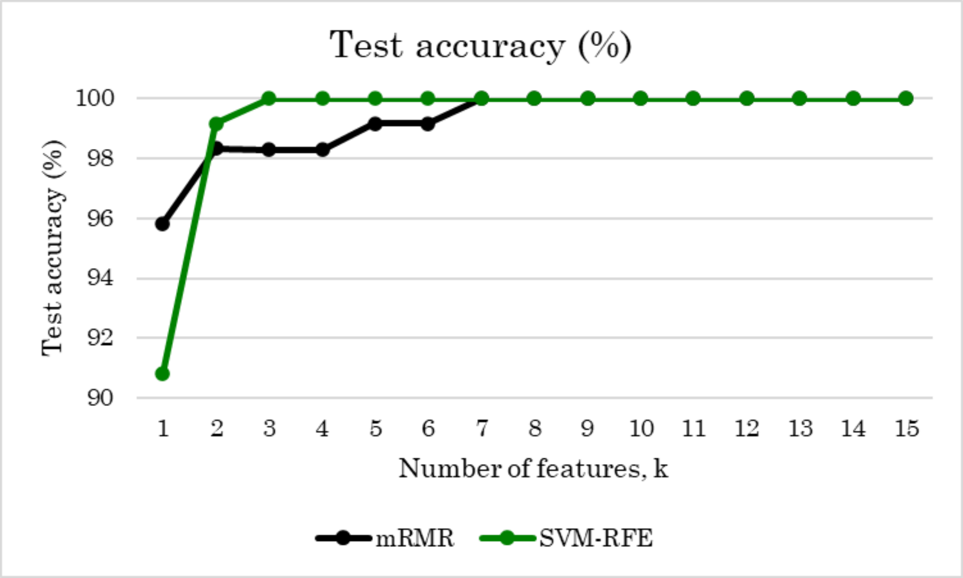

Figure 4.6: Comparison of the test accuracy in terms of selected features by mRMR and SVM-RFE.

4.5 Performance of proposed hybrid approach

The presence of a single feature selection algorithm might not be enough to achieve the highest accuracy by just using a smaller number of features, especially in the big data era where machine learning data results need to be optimized quicker and achieve high accuracy at the same time. In the previous two sections, both classification models of mRMR and SVM-RFE encountered their problems in achieving high accuracy when a smaller number of features were included, due to each limitation. Therefore, the proposed hybrid method in this study will combine both feature selection algorithms to complement each other’s limitations to achieve a higher classification accuracy with a fewer number of optimal features.

The process of an experiment for mRMR+SVM-RFE (Hybrid) started by having mRMR select the top 15 features from the training dataset and ranked them according to their importance. The top 15 ranked features shortlisted by mRMR will then be input into SVM-RFE which acts as a feature ranking mechanism to achieve the final ranking for the top 15 shortlisted features of the proposed hybrid model, as shown in table 4.3. After performing cross-validation accuracy for the hybrid model, test accuracy will be used to validate the result for the top 15 shortlisted features. Figures 4.7 and 4.8 will show the cross-validation accuracy result and test accuracy result for the top 15 shortlisted features selected by the hybrid method respectively.

	Ranked Features
mRMR+SVM-RFE (Proposed hybrid method)	$f_{189}, f_{209}, f_{262}, f_{257}, f_{129}, f_{201}, f_{260}, f_{141}, f_{261}, f_{105}, f_{173}, f_{233}, f_{185}, f_{269}, f_{649}$

Table 4.3: List of top 15 shortlisted features selected by mRMR+SVM-RFE (Hybrid) algorithm

As shown in Figure 4.7, the cross-validation accuracy using the hybrid method has significantly improved and maintained consistency of 99.6% in accuracy from $k = 4$ until $k = 15$. The model only required the first four features, which consist of $f_{189}, f_{209}, f_{262}, f_{257}$ to reach the peak accuracy of 99.6%. Although the hybrid model could not achieve 100% cross-validation accuracy as compared to the SVM-RFE model, it still managed to solve the accuracy fluctuation issue faced by mRMR and it achieved the peak accuracy of 99.6% faster than the mRMR model, which required $k = 8$ to reach the same result. Besides that, the cross-validation accuracy using the hybrid method has also proven to achieve a better result as compared to the baseline accuracy of 99.3%.

As for the test accuracy of the hybrid model in Figure 4.8, the accuracy for the test dataset reached the peak at 100% accuracy when $k = 4$. It is also noticeable that there is decreasing in test accuracy between $k = 9$ and $k = 10$. The sudden drop in test accuracy may be due to SVM-RFE in the hybrid model being still unable to completely solve the overfitting issue on the test dataset, as it may suggest that the size of 120 samples in the test dataset is considered to be small.

A larger size of test dataset should be used to avoid overfitting which causes the instability of SVM-RFE to lower the test performance of the hybrid model.

However, the hybrid model was still able to perform well as it can achieve a much better and stable test accuracy result when fewer features with higher classification accuracy are required. Although the hybrid model may seem to have many disadvantages in having consistency in accuracy as compared to the SVM-RFE model but selecting the most suitable model solely based on the cross-validation accuracy and test accuracy is still inadequate to make the final call. This is because there are still other performance metrics and comparisons to be made between the three classification models, which will be discussed in sections 4.6 and 4.7.

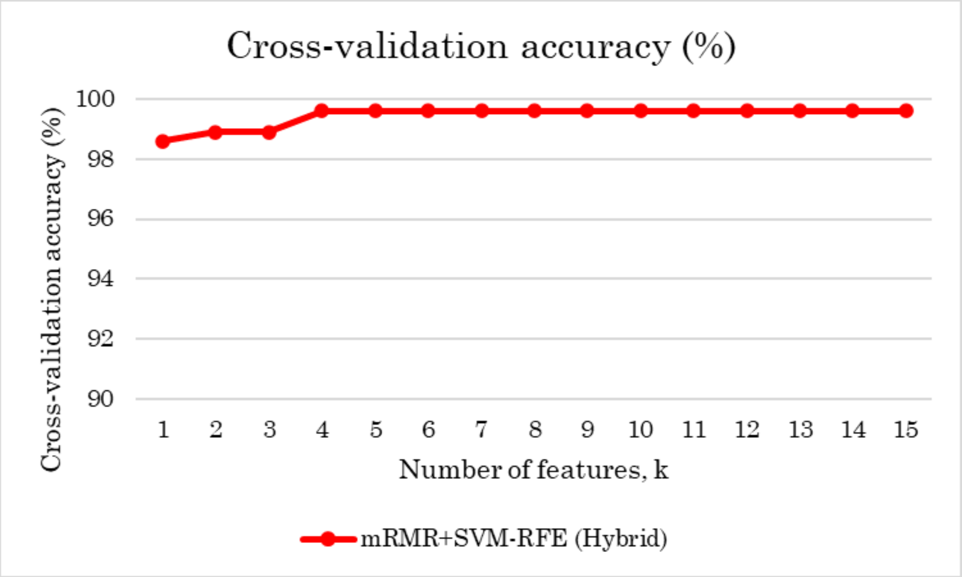

Figure 4.7: Cross-validation accuracy of top 15 shortlisted features selected by mRMR+SVM-RFE.

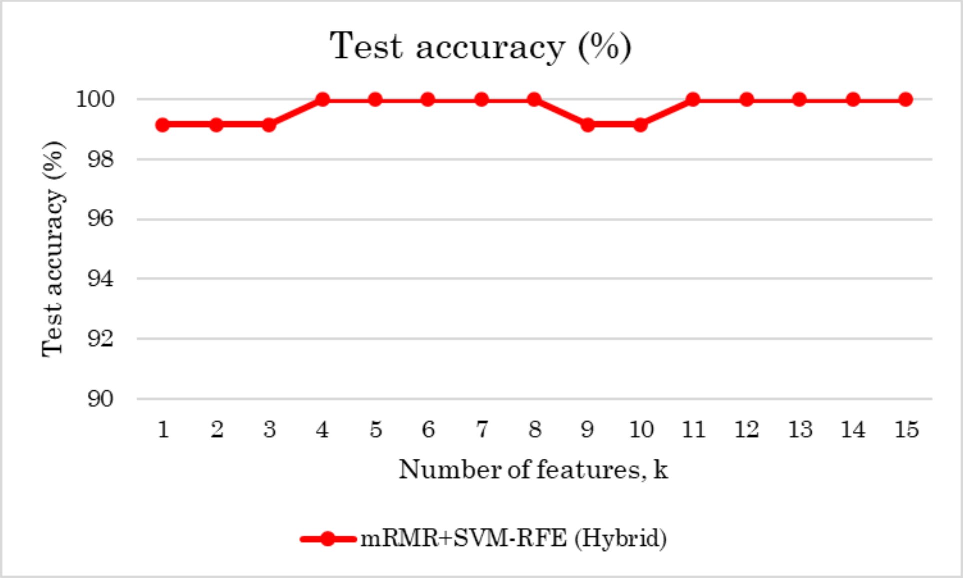

Figure 4.8: Test accuracy of top 15 shortlisted features selected by mRMR+SVM-RFE.

4.6 Comparison of three classification models

This section will discuss more on the comparison of cross-validation accuracy, test accuracy, and the average accuracy between the three classification models. The line graph plotted with black colour represents mRMR, SVM-RFE will be plotted in green line and the hybrid model will be plotted with a red line instead. The three coloured lines represent the cross-validation accuracy and test accuracy in Figures 4.9 and 4.10, respectively.

The cross-validation accuracy and test accuracy of mRMR, SVM-RFE, and mRMR+SVM-RFE were shown in Figures 4.9 and 4.10. The accuracy curves for mRMR in both Figs. 4.9 and 4.10 showed up-down fluctuation when more features were included. This revealed the fact that while the mRMR method selects the most relevant features, it also includes some redundant features during the process.

Meanwhile, the performance of the SVM-RFE was good only when more features were included in the predictive model. It was obvious that SVM-RFE gave the lowest accuracy compared to the other two methods if only the first feature was included. This showed that the first feature from the SVM-RFE-selected feature subset was not necessarily the most significant one. The fact that the features selected by SVM-RFE are not ranked in the order of importance was disclosed here. This has no doubt caused the SVM-RFE to be unstable in selecting the most important feature according to the order and prone to overfitting problems when the validation dataset is tested.

Among these methods, the proposed hybrid method yielded the highest accuracy when only one feature was selected, as shown in table 4.4. The hybrid model has proven that when only one feature is required for binary classification, 99.2% of the time it can be able to recognize the pattern of the handwriting digit ‘4’ and ‘9’. Unlike the mRMR method, the hybrid method performed more stable when more features were added in. Results showed that the hybrid method managed to improve the performance of the classification by addressing the redundant features and the ranking issue in the SVM-RFE. Besides that, the average cross-validation accuracy and average test accuracy in table 4.5 have shown that the proposed hybrid model can achieve higher accuracy of 99.44% and 99.72% as compared to the mRMR and SVM-RFE model.

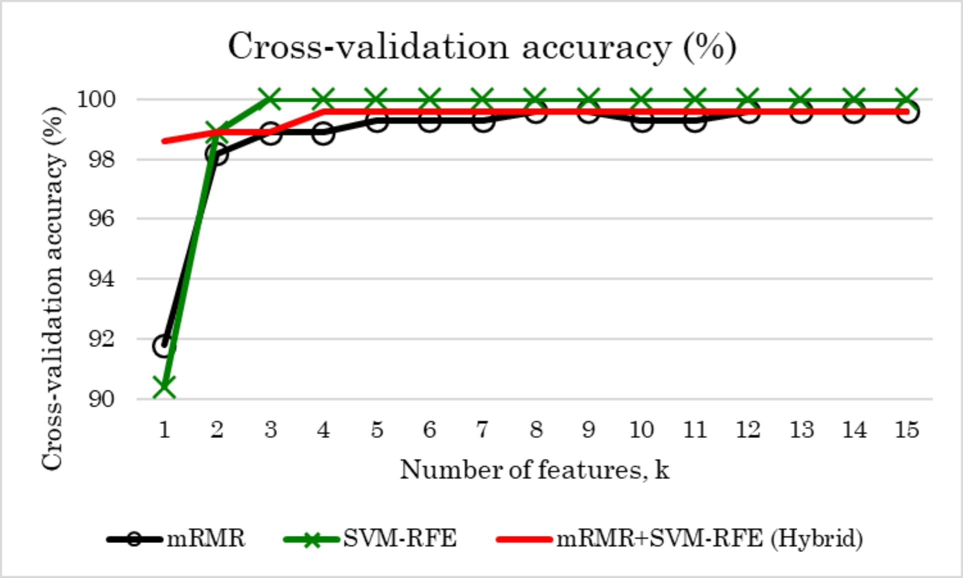

Figure 4.9: Comparison of the Cross-validation accuracy in terms of selected features by mRMR, SVM-RFE, and mRMR+SVM-RFE.

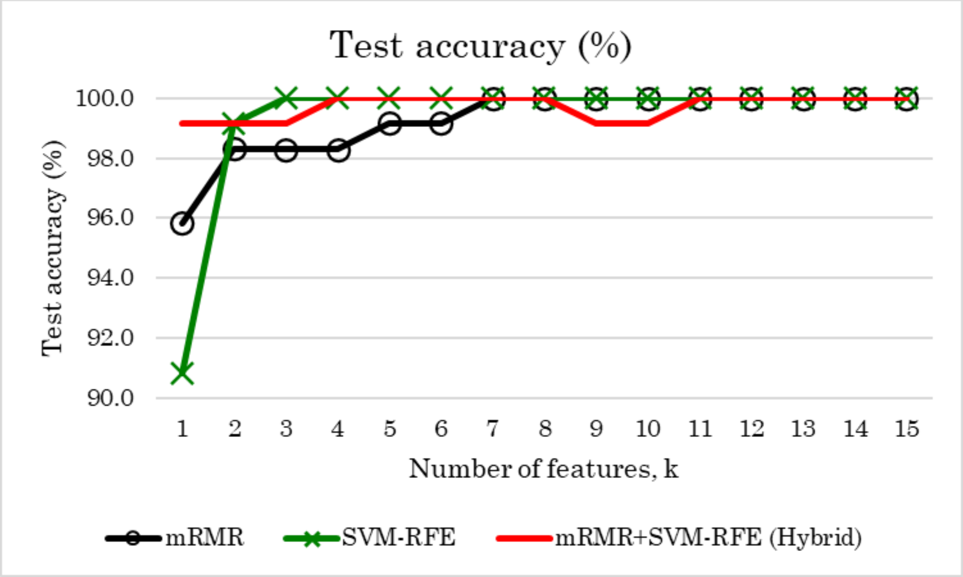

Figure 4.10: Comparison of the test accuracy in terms of selected features by mRMR, SVM-RFE, and mRMR+SVM-RFE.

Algorithm	Cross-validation accuracy (%)	Test accuracy (%)
mRMR	91.8	95.8
SVM-RFE	90.4	90.8
mRMR+SVM-RFE (Proposed hybrid method)	**98.6**	**99.2**

Table 4.4: Comparison of Cross-validation accuracy and test accuracy between mRMR, SVM-RFE, and hybrid model when $k = 1$.

Algorithm	Average Cross-validation accuracy (%)	Average test accuracy (%)
mRMR	98.79	99.27
SVM-RFE	99.29	99.33
mRMR+SVM-RFE (Proposed hybrid method)	**99.44**	**99.72**

Table 4.5: Comparison of average Cross-validation accuracy and average test accuracy between mRMR, SVM-RFE, and hybrid model.

4.7 Performance metrics between three classification models

As mentioned in section 4.5, before determining the final suitable model to be used, performance metrics of the three classification models needed to be done to obtain sufficient evaluation of the performance of the classification models. The performance metrics that will be discussed in the following subsection will be the confusion matrix (CM) and Receiver Operating Characteristic Curve (ROC).

4.7.1 Receiver Operating Characteristic Curve (ROC) and Confusion Matrix

The best-fitted model can be evaluated using Recall, Precision, AUC, and Classification accuracy. The best-fitted model is considered as a model that has an accuracy value that is nearer to 1 whereas the model that has accuracy nearer to 0 is considered a weak fitted model. There are two parameters inside the ROC curve, which consist of True Positive Rate (TPR) and False Positive Rate (FPR). A classification model that achieved a higher AUC value implied better classification performance. The AUC and the accuracy of the test data for the three models were summarized in table 4.6. From table 4.6, the average classification accuracy among the three models was close to each other and rather accurate. The comparison showed that the hybrid model exhibited the highest classification accuracy among the three models, with an accuracy of 99.45%, followed by SVM-RFE (99.29%) and lastly mRMR (98.80%). This was evidence that the feature selection combination of mRMR and SVM-RFE outperformed the single feature selection. Meanwhile, the confusion matrix summarizes the prediction performance of a classification model. Appendix A, B, and C will display the lists of the confusion matrix and ROC curve for the top 15 ranked features selected by mRMR, SVM-RFE, and the proposed hybrid method.

Based on table 4.6, it is noticeable that the overall result achieved using the proposed hybrid method has significantly improved. In comparison to mRMR and SVM-RFE methods, the proposed hybrid method has managed to improve the average positive value by lowering it to 0.19%. Besides that, the average specificity of the proposed hybrid method achieved 99.81% as compared to mRMR with 99.11% and SVM-RFE with 99.25%. The average precision has also significantly increased to 99.81%; the area under curve (AUC) had also been greatly optimized by the hybrid method to reach the value of 1. As a whole, the implementation of the hybrid method has proven to improve the binary handwriting digit feature classification accuracy using SVM classifier compared to the usage of a single feature selection method.

Average	**Feature Selection algorithm**		
	mRMR	**SVM-RFE**	**mRMR+ SVM-RFE** **(Hybrid)**
TPR (Recall) (%)	98.48	**99.32**	99.09
FPR (%)	0.89	0.75	**0.19**
TNR (Specificity) (%)	99.11	99.25	**99.81**
FNR (%)	1.52	**0.68**	0.91
Precision (%)	99.13	99.26	**99.81**
AUC	0.9960	0.9973	**1.0000**
Accuracy (%)	98.80	99.29	**99.45**

Table 4.6: Summary of ROC, Confusion matrix, AUC, and classification accuracy of test data between mRMR, SVM-RFE, and Proposed hybrid method for the top 15 ranked features.

4.8 Final feature subset selection

From the previous section, the proposed hybrid method was chosen as the best-fitted classification model among the other classification model. Therefore, the final number of features chosen by the proposed hybrid method has significantly been shortlisted from 649 features down to only 4 features. The 4 significant shortlisted features from the MFEAT dataset were $f_{189}, f_{209}, f_{262}, f_{257}$. The test set accuracy has managed to achieve 100.00% accuracy with 4 features as compared to 99.2% with full features. On top of that, the proposed hybrid method was also able to achieve the same test accuracy of 99.2% with only one significant feature, f_{189}, unlike mRMR and SVM-RFE models, which have only been able to achieve 95.8% and 90.8% when $k = 1$. The result in section 4.5 and 4.6 has shown the ability of the proposed hybrid method to select a small number of significant binary handwritten digit features to achieve higher classification accuracy. In other words, the proposed hybrid method was capable of selecting four significant features from the large number of features found in the MFEAT dataset, to achieve optimal classification results.

CHAPTER 5

CONCLUSION

5.1 Conclusion

Redundancy, irrelevance, and overfitting are common issues that deteriorate the overall classification accuracy. In this study, the proposed hybrid approach that consists of mRMR and SVM-RFE algorithm were combined to solve the redundancy, fluctuation, and overfitting issue faced by a single feature selection algorithm, to achieve high classification accuracy by utilizing only a small number of most significant features. Tested on the 4-9 binary classification, the hybrid method managed to achieve relatively higher classification accuracy in terms of AUC and average classification accuracy for the top 15 ranked features.

Furthermore, the results shown in Chapter 4 have proven that the hybrid classification model achieved better performance in Cross-validation and test accuracy, especially for the first selected feature. By comparison, the performance of the predictive model was more stable, and less fluctuated when more features were added. The hybrid method not only alleviated the issue of mRMR which often gives wavering performance when more features are added but also alleviated the ranking issue of SVM-RFE. The mRMR helped in shortlisting the most relevant features and these highly relevant features consequently made the SVM-RFE give higher accuracy. Via the discussion in the previous chapter, it has shown that the hybrid feature selection approach produced a much better result in performance metrics as compared to the single feature selection approach. In conclusion, the hybrid approach can be a feasible option for better classification for predicting models using only a few most significant features.

5.2 Limitation of the study

Throughout the study, the proposed hybrid method still posed a minor problem in handling the overfitting problem, which cause the inconsistency of test accuracy in two features. The overfitting issue caused by SVM-RFE could be potentially solved using Correlation Bias Reduction or using two-stage SVM-RFE in a future study. Besides that, the study only utilized linear SVM as a machine learning classifier instead of other kernels like polynomial, Gaussian, Sigmoid, and so on. This could cause a drawback to the study as the dataset might not be linearly separable.

5.3 Recommendations for future studies

It is recommended that the proposed hybrid approach can be implemented onto other classification problems and incorporated with other classifiers (such as KNN, decision tree, random forest, etc.) for further study. The other recommendation for future study would be to use different kernels that concern SVM, as the dataset may not be linearly separable. The comparison between the performance of each kernel can be done to ensure high accuracy and stability in the classification model. The implementation of Correlation Bias Reduction (CBR) or two-stage SVM-RFE can be done in the future study together with mRMR to get rid of the existing overfitting issue and further enhance the performance of the hybrid method. The sample size of the dataset should increase to counter the potential overfitting issue caused by the small test set. A much more complex multiclass handwritten digit recognition problem can be further explored by using the proposed hybrid method to produce better classification accuracy.

REFERENCES

Arumugam, P., Kadhirveni, Lakshmi Priya, G, Manimannan., 2021. Prediction, Cross Validation and Classification in the Presence COVID-19 of Indian States and Union Territories using Machine Learning Algorithms. *International Journal of Recent Technology and Engineering*, 10(1), pp.16-20. http://www.doi.org/10.35940/ijrte.A5659.0510121

Aziz, R., Verma, C. and Srivastava, N., 2017. A novel approach for dimension reduction of microarray. *Computational Biology and Chemistry*, 71, pp.161-169. Available at: <http://library.utar.my> [Accessed 10 October 2021].

Battiti, R., 1994. Using mutual information for selecting features in supervised neural net learning. *IEEE Transactions on Neural Networks*, 5(4), pp.537-550. https://doi.org/10.1109/72.298224

Cervantes, J., Garcia-Lamont, F., Rodríguez-Mazahua, L. and Lopez, A., 2020. A comprehensive survey on support vector machine classification: Applications, challenges and trends. *Neurocomputing*, 408, pp.189-215. https://doi.org/10.1016/j.neucom.2019.10.118

Cilia, N., De Stefano, C., Fontanella, F. and Scotto di Freca, A., 2019. A ranking-based feature selection approach for handwritten character recognition. *Pattern Recognition Letters*, 121, pp.77-86. DOI: 10.1016/j.patrec.2018.04.007

Cover, T.M. and Thomas, J.A., 2012. *Elements of Information Theory*. John Wiley & Sons.

Ding, C. and Peng, H., 2005. Minimum redundancy feature selection from microarray gene expression data. *Journal of Bioinformatics and Computational Biology*, 03(02), pp.185-205. https://doi.org/10.1142/s0219720005001004

Durgesh, K.S. and Lekha, B., 2010. Data classification using support vector machine. *Journal of theoretical and applied information technology*, 12(1), pp.1-7.

Estevez, P., Tesmer, M., Perez, C. and Zurada, J., 2009. Normalized Mutual Information Feature Selection. *IEEE Transactions on Neural Networks*, 20(2), pp.189-201. https://doi.org/10.1109/TNN.2008.2005601

Guyon, I., Weston, J., Barnhill, S. and Vapnik, V., 2002. Gene Selection for Cancer Classification using Support Vector Machines. *Machine Learning*, 46(1/3), pp.389-422. http://dx.doi.org/10.1023/A:1012487302797

Huang, M., Hung, Y., Lee, W., Li, R. and Jiang, B., 2014. SVM-RFE Based Feature Selection and Taguchi Parameters Optimization for Multiclass SVM Classifier. *The Scientific World Journal*, 2014, pp.1-10. https://doi.org/10.1155/2014/795624

Jeon, H. and Oh, S., 2020. Hybrid-Recursive Feature Elimination for Efficient Feature Selection. *Applied Sciences*, 10(9), p.3211. https://doi.org/10.3390/app10093211

Jovic, A., Brkic, K. and Bogunovic, N., 2015. A review of feature selection methods with applications. *2015 38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pp.1200–1205. https://doi.org/10.1109/MIPRO.2015.7160458

Kalina, J. and Schlenker, A., 2015. A Robust Supervised Variable Selection for Noisy High-Dimensional Data. *BioMed Research International*, 2015, pp.1–10. https://doi.org/10.1155/2015/320385

Kalita, S., Gautam, D., Kumar Sahoo, A. and Kumar, R., 2019. A combined approach of feature selection and machine learning technique for handwritten character recognition. *International Journal of Advanced Studies of Scientific Research*, 4(4). https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3408997

Kari, T., Gao, W., Zhao, D., Abiderexiti, K., Mo, W., Wang, Y. and Luan, L., 2018. Hybrid feature selection approach for power transformer fault diagnosis based on support vector machine and genetic algorithm. *IET Generation, Transmission & Distribution*, 12(21), pp.5672-5680. https://doi.org/10.1049/iet-gtd.2018.5482

Kwak, N. and Choi, C.-H., 2002. Input feature selection for classification problems. *IEEE Transactions on Neural Networks*, 13(1), pp.143-159. https://doi.org/10.1109/72.977291

Ladha, L. and Deepa, T., 2011. Feature selection methods and algorithms. *International Journal on Computer Science and Engineering*, 3(5), pp.1787-1797.

Maleki, F., Muthukrishnan, N., Ovens, K., Reinhold, C. and Forghani, R., 2020. Machine Learning Algorithm Validation: From Essentials to Advanced Applications and Implications for Regulatory Certification and Deployment. *Neuroimaging Clinics of North America*, 30(4), pp.433-445. https://doi.org/10.1016/j.nic.2020.08.004

Morera, Á., Sánchez, Á., Vélez, J. and Moreno, A., 2018. Gender and Handedness Prediction from Offline Handwriting Using Convolutional Neural Networks. *Complexity*, 2018, pp.1-14. https://doi.org/10.1155/2018/3891624

Olaolu, A., Abdulsalam, S., Mope, I. and Kazeem, G., 2018. A Comparative Analysis of Feature Selection and Feature Extraction Models for Classifying Microarray Dataset. *Computing & Information Systems*, 22(2), pp.29-37. Available at: <https://library.utar.my> [Accessed 10 October 2021].

Peng, H., Long, F. and Ding, C., 2005. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8), pp.1226-1238. https://doi.org/10.1109/TPAMI.2005.159

Pino, A. and Morell, C., 2013. Analytical and Experimental Study of Filter Feature Selection Algorithms for High-dimensional Datasets. *Proceedings of the Fourth International Workshop on Knowledge Discovery, Knowledge Management and Decision Support*, pp.339-349. https://dx.doi.org/10.2991/.2013.42

Shammakhi, M., Mirzaei, A., Khavari, P. and Pourahmadi, V., 2015. Combined mRMR-MLPSVM Scheme for High Accuracy and Low Cost Handwritten Digits Recognition. *2015 9th Iranian Conference on Machine Vision and Image Processing (MVIP)*. https://doi.org/10.1109/IranianMVIP.2015.7397526

Singh, P., Verma, A. and Chaudhari, N., 2014. Devanagri Handwritten Numeral Recognition using Feature Selection Approach. *International Journal of Intelligent Systems and Applications*, 6(12), pp.40-47. https://doi.org/10.5815/IJISA.2014.12.06

Singh, P., Verma, A. and Chaudhari, N., 2015. Feature selection based classifier combination approach for handwritten Devanagari numeral recognition. *Sadhana*, 40(6), pp.1701-1714. https://doi.org/10.1007/S12046-015-0419-X

Tadist, K., Najah, S., Nikolov, N., Mrabti, F. and Zahi, A., 2019. Feature selection methods and genomic big data: a systematic review. *Journal of Big Data*, 6(1), pp.1-24. https://doi.org/10.1186/s40537-019-0241-0

Varghese, D., 2019. *Comparative study on classic machine learning algorithms*. Medium. Available at: https://towardsdatascience.com/comparative-study-on-classic-machine-learning-algorithms-24f9ff6ab222 [Accessed October 16, 2021].

Velliangiri, S., Alagumuthukrishnan, S. and Thankumar joseph, S., 2019. A Review of Dimensionality Reduction Techniques for Efficient Computation. *Procedia Computer Science,* 165, pp.104-111. https://doi.org/10.1016/j.procs.2020.01.079

Vergara, J.R. and Estévez, P.A., 2013. A review of feature selection methods based on mutual information. *Neural computing and applications*, 24(1), pp.175-186. https://doi.org/10.1007/s00521-013-1368-0

Wolpert, D.H., 1996. The Lack of A Priori Distinctions Between Learning Algorithms. *Neural Computation,* 8(7), pp.1341-1390. http://dx.doi.org/10.1162/neco.1996.8.7.1341

Yan, K. and Zhang, D., 2015. Feature selection and analysis on correlated gas sensor data with recursive feature elimination. *Sensors and Actuators B: Chemical*, 212, pp.353-363. https://doi.org/10.1016/j.snb.2015.02.025

Ying, X., 2019. An Overview of Overfitting and its Solutions. *Journal of Physics: Conference Series*, 1168(022022), pp.1-6. http://dx.doi.org/10.1088/1742-6596/1168/2/022022

Zhao, Z., Anand, R. and Wang, M., 2019. Maximum Relevance and Minimum Redundancy Feature Selection Methods for a Marketing Machine Learning Platform. *2019 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*. https://doi.org/10.1109/DSAA.2019.00059

Zhou, Q., Hong, W., Shao, G. and Cai, W., 2009. A new SVM-RFE approach towards ranking problem. *2009 IEEE International Conference on Intelligent Computing and Intelligent Systems*, pp.270-273. https://doi.org/10.1109/ICICISYS.2009.5357684

APPENDICES

APPENDIX A – CONFUSION MATRIX AND ROC CURVE FOR mRMR FEATURE SELECTION ALGORITHM

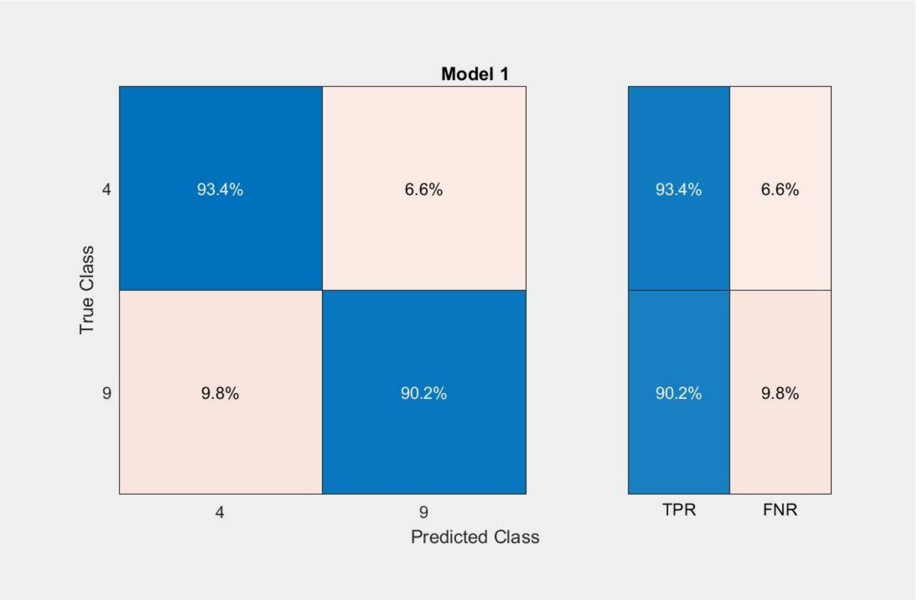

Figure A.i: Confusion Matrix of one feature selected by mRMR feature selection algorithm

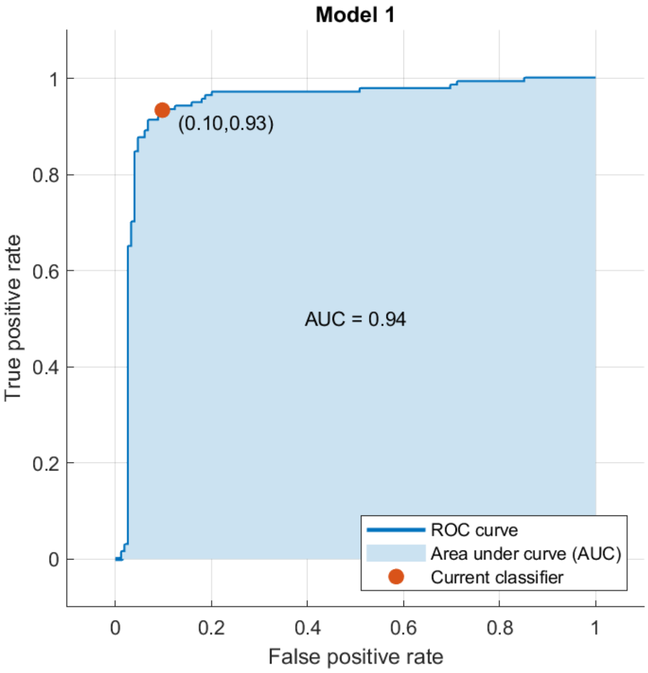

Figure A.ii: ROC Curve of one feature selected by mRMR feature selection algorithm

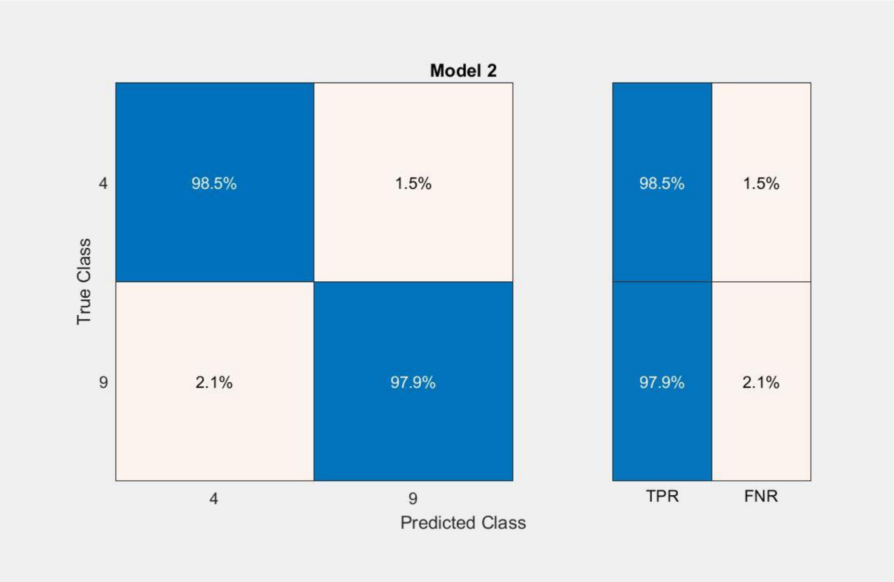

Figure A.iii: Confusion Matrix of two features selected by mRMR feature selection algorithm

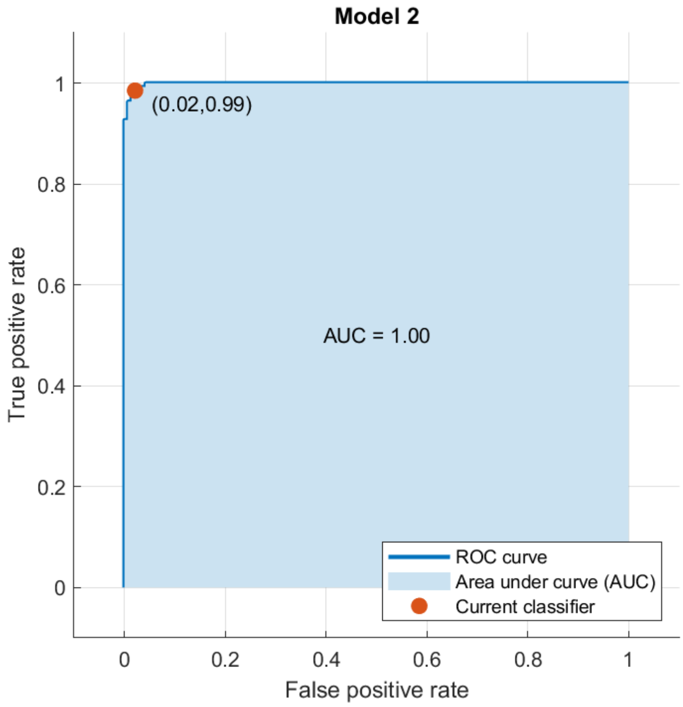

Figure A.iv: ROC Curve of two features selected by mRMR feature selection algorithm

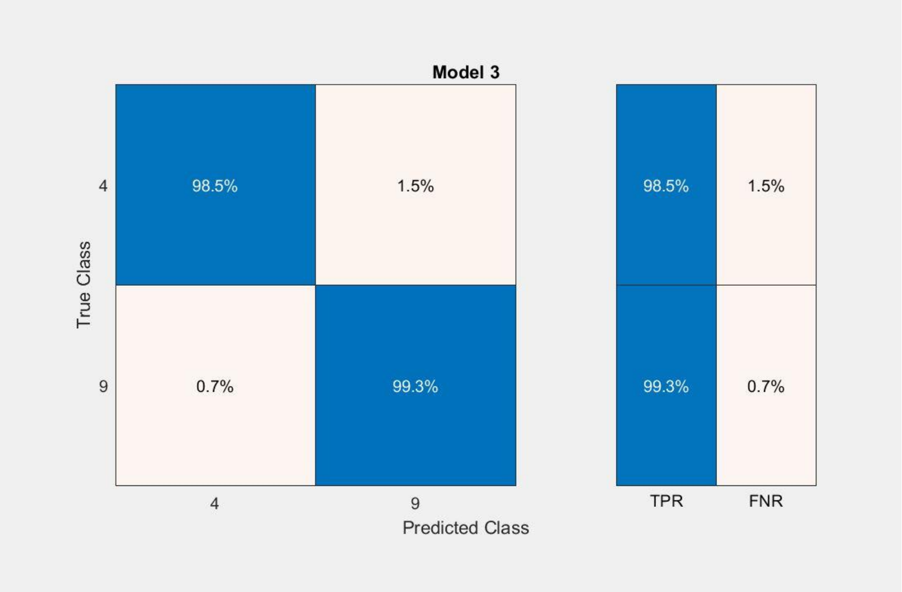

Figure A.v: Confusion Matrix of three features selected by mRMR feature selection algorithm

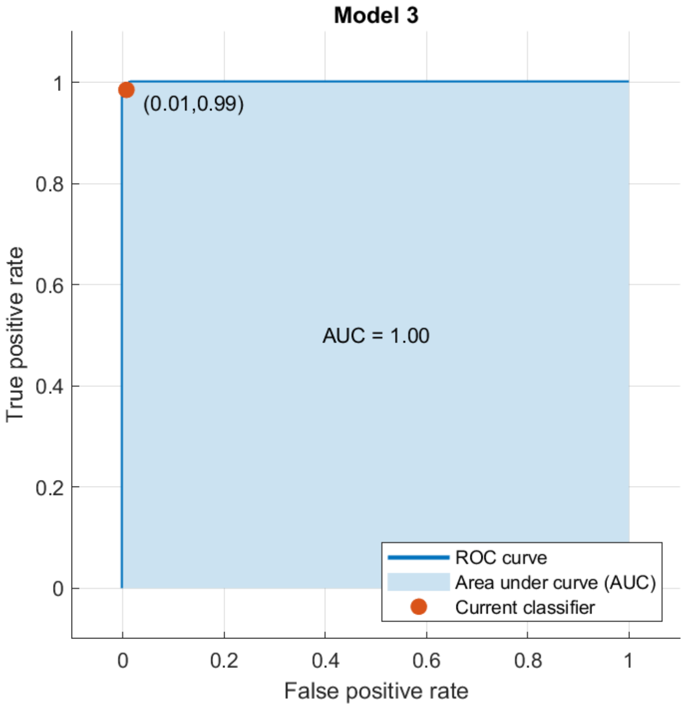

Figure A.vi: ROC Curve of three features selected by mRMR feature selection algorithm

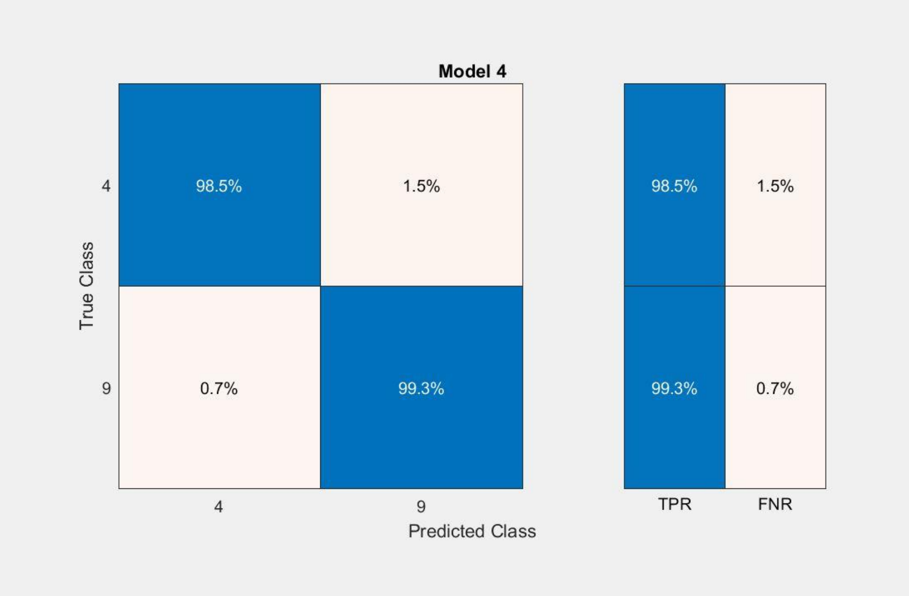

Figure A.vii: Confusion Matrix of four features selected by mRMR feature selection algorithm

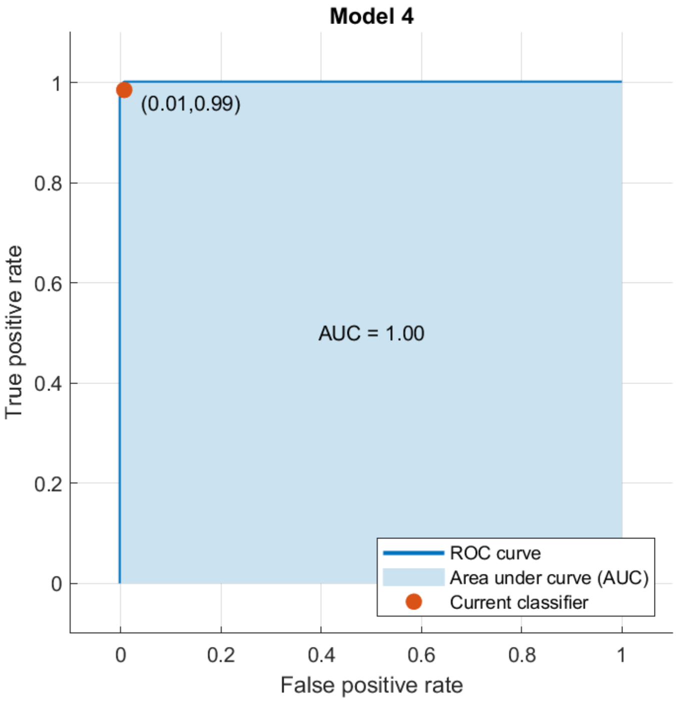

Figure A.viii: ROC Curve of four features selected by mRMR feature selection algorithm

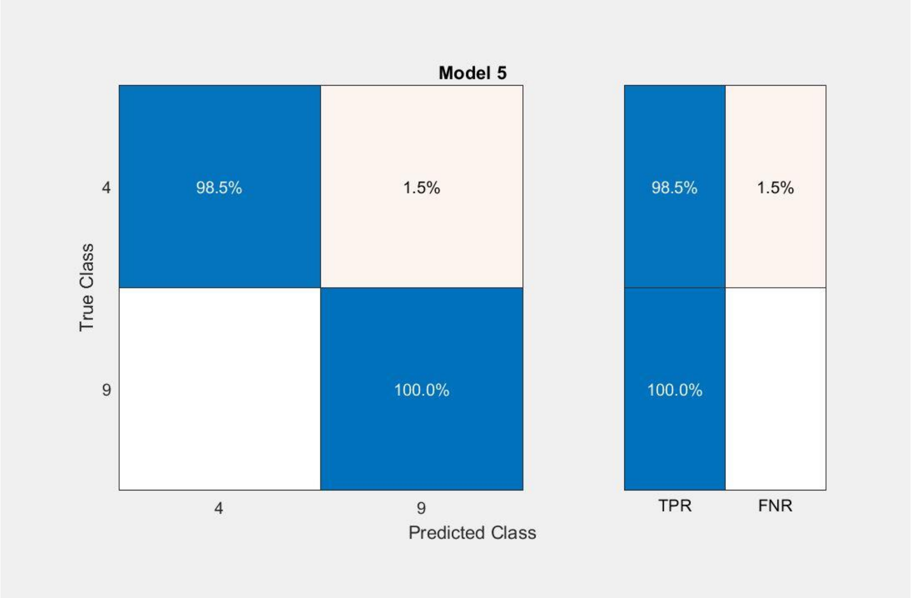

Figure A.ix: Confusion Matrix of five features selected by mRMR feature selection algorithm

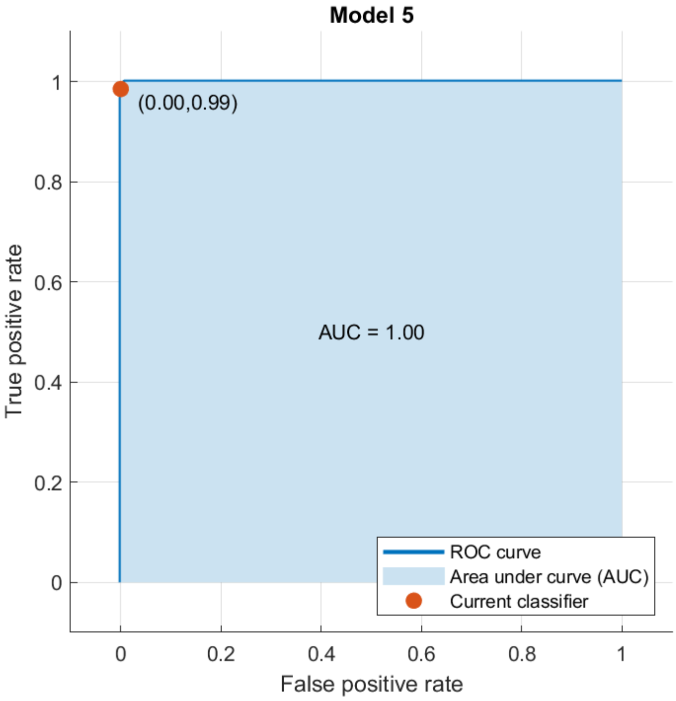

Figure A.x: ROC Curve of five features selected by mRMR feature selection algorithm

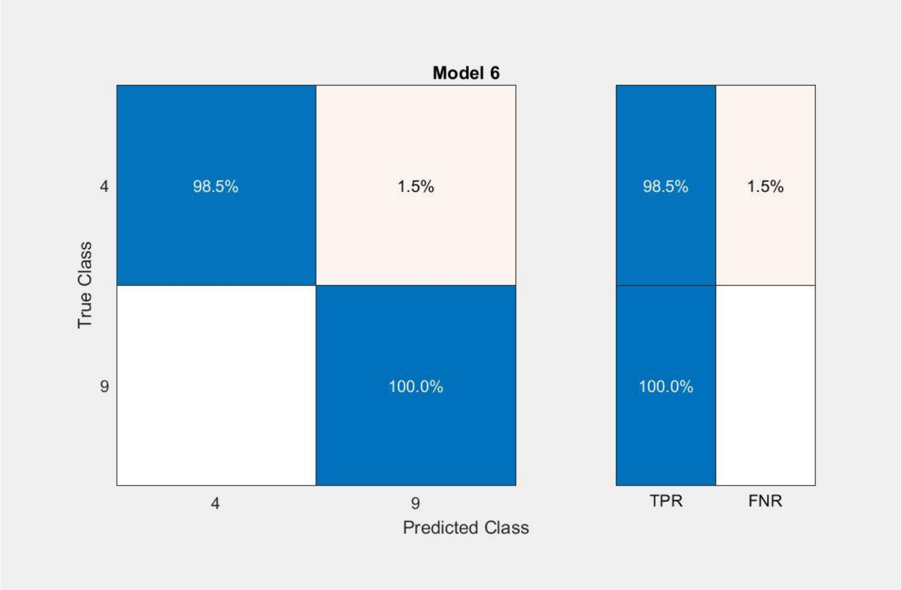

Figure A.xi: Confusion Matrix of six features selected by mRMR feature selection algorithm

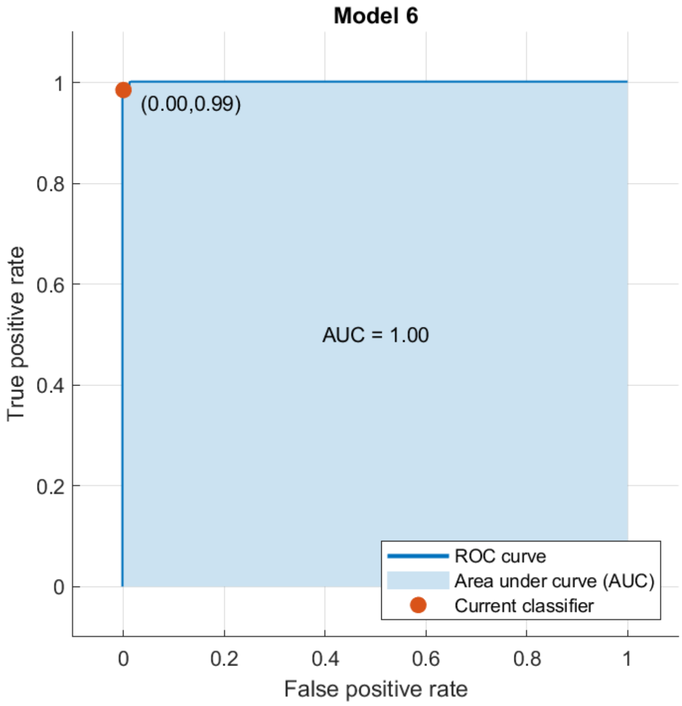

Figure A.xii: ROC Curve of six features selected by mRMR feature selection algorithm

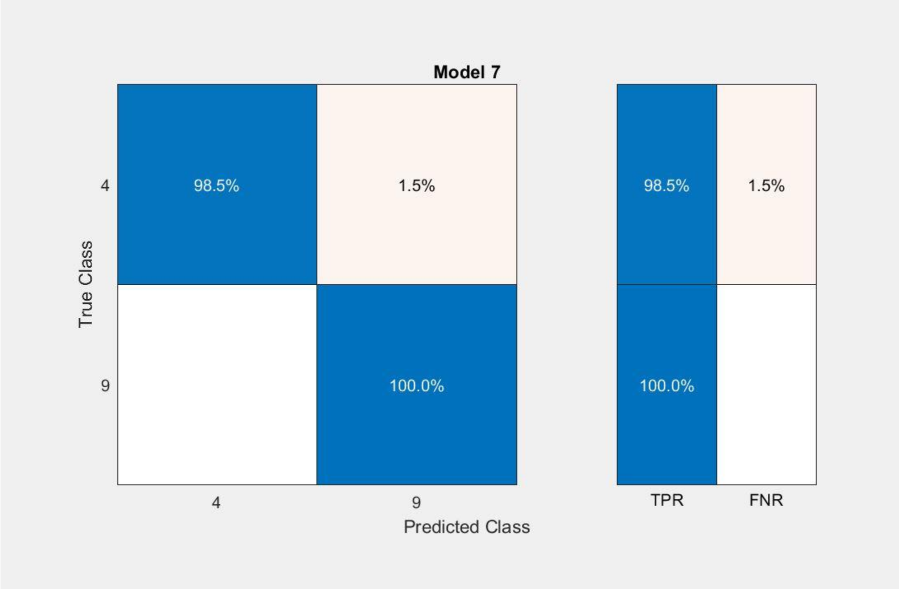

Figure A.xiii: Confusion Matrix of seven features selected by mRMR feature selection algorithm

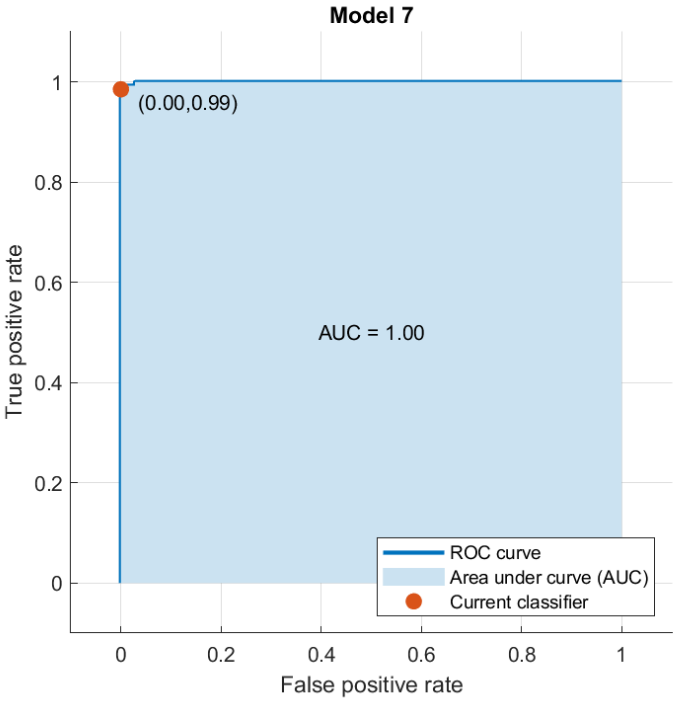

Figure A.xiv: ROC Curve of seven features selected by mRMR feature selection algorithm

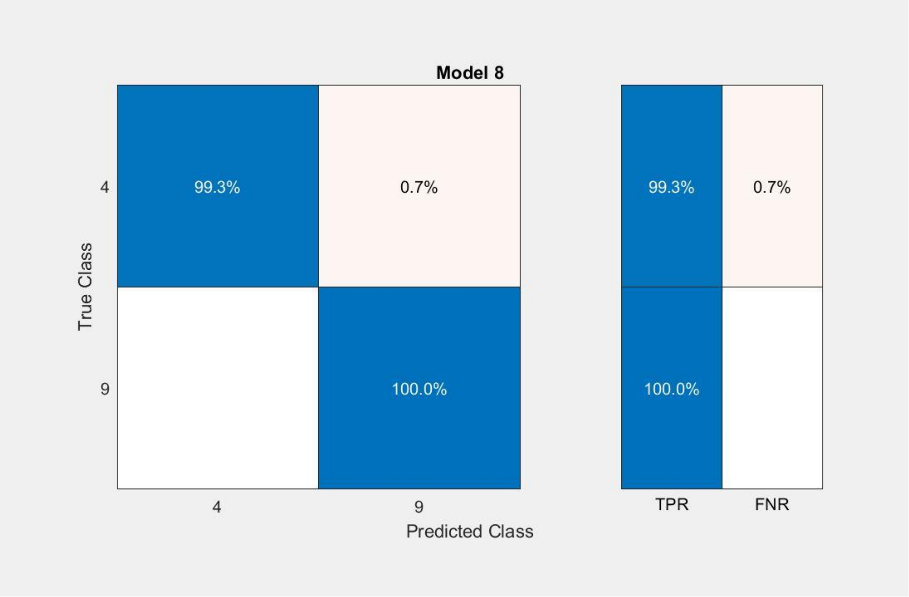

Figure A.xv: Confusion Matrix of eight features selected by mRMR feature selection algorithm

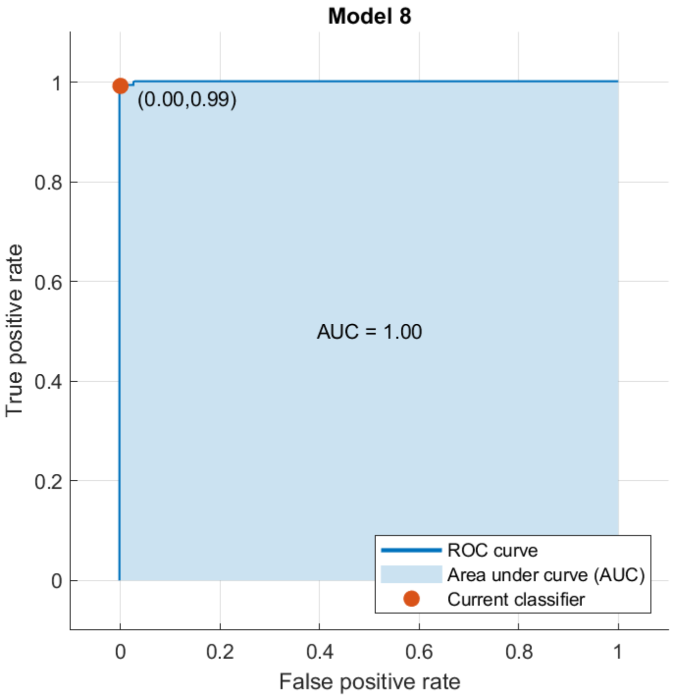

Figure A.xvi: ROC Curve of eight features selected by mRMR feature selection algorithm

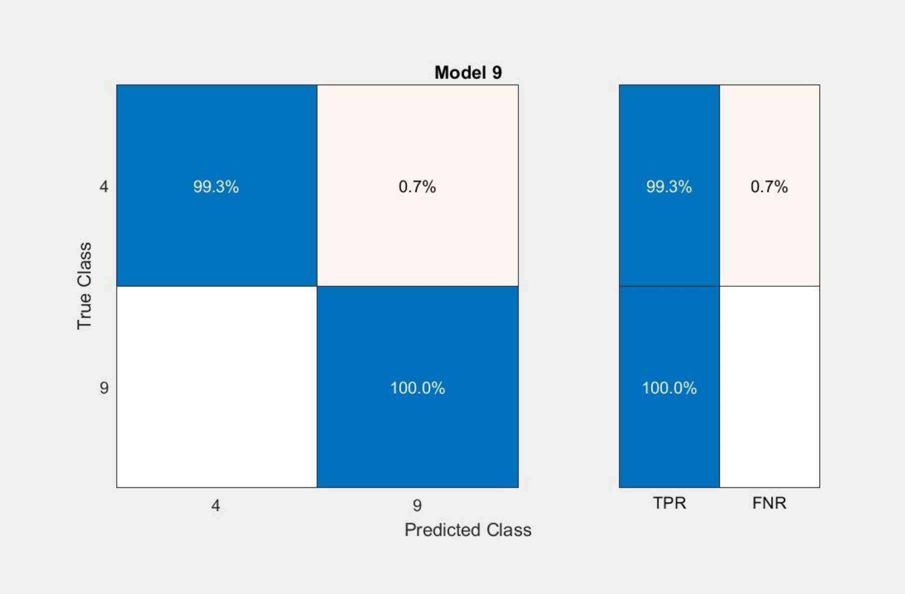

Figure A.xvii: Confusion Matrix of nine features selected by mRMR feature selection algorithm

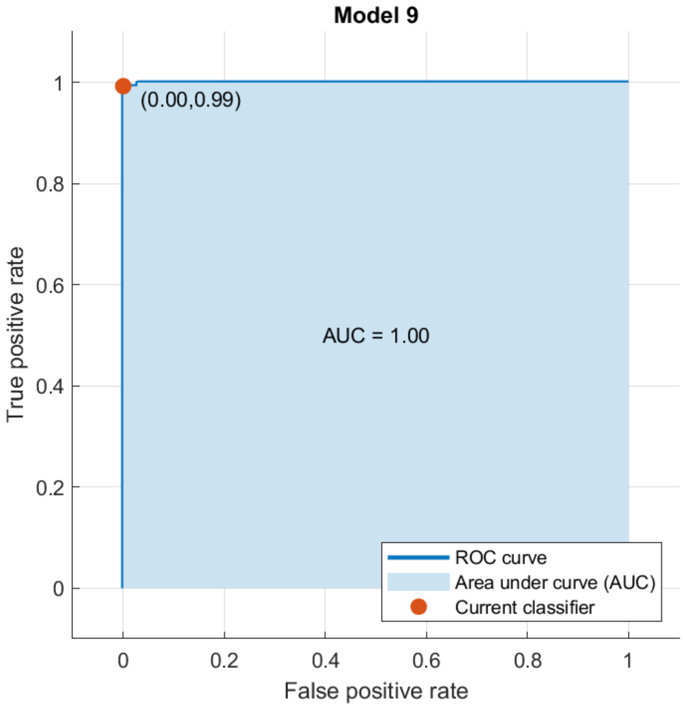

Figure A.xviii: ROC Curve of nine features selected by mRMR feature selection algorithm

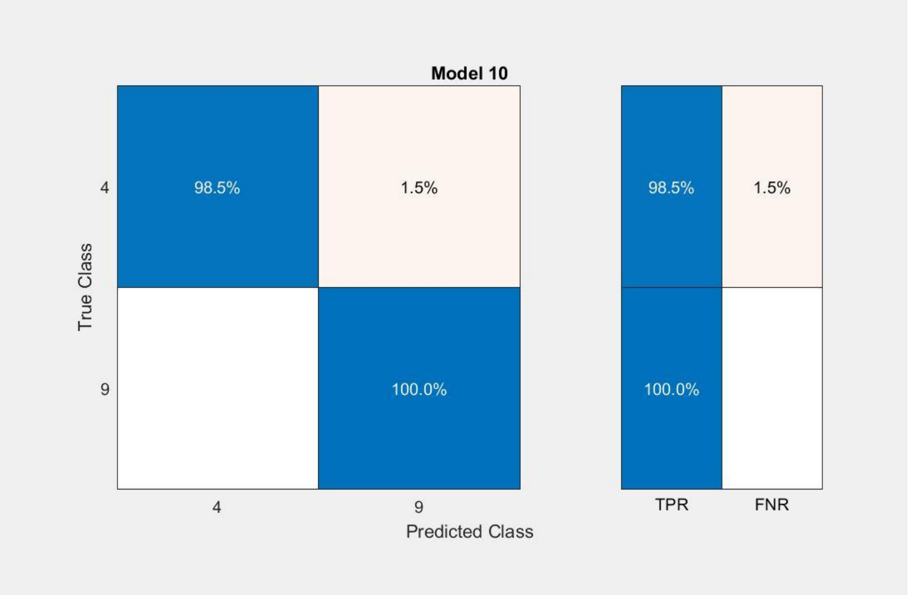

Figure A.xvix: Confusion Matrix of ten features selected by mRMR feature selection algorithm

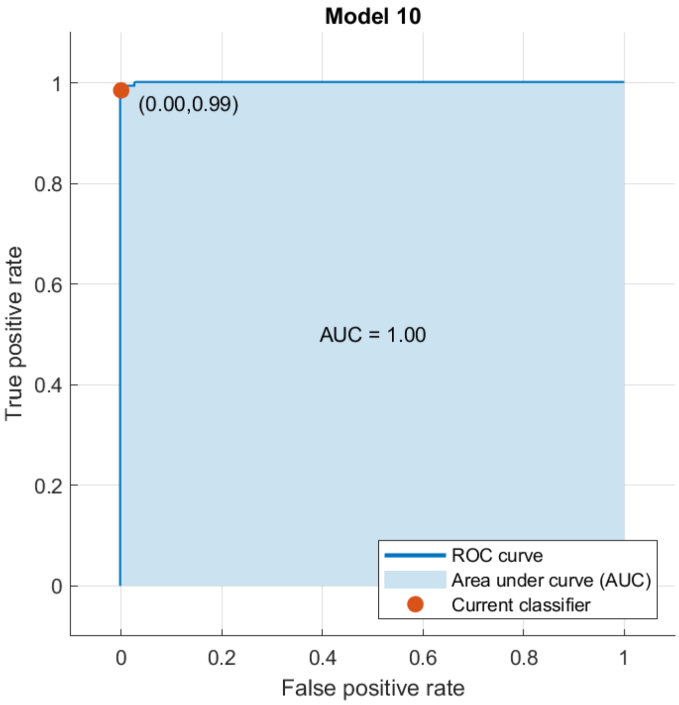

Figure A.xx: ROC Curve of ten features selected by mRMR feature selection algorithm

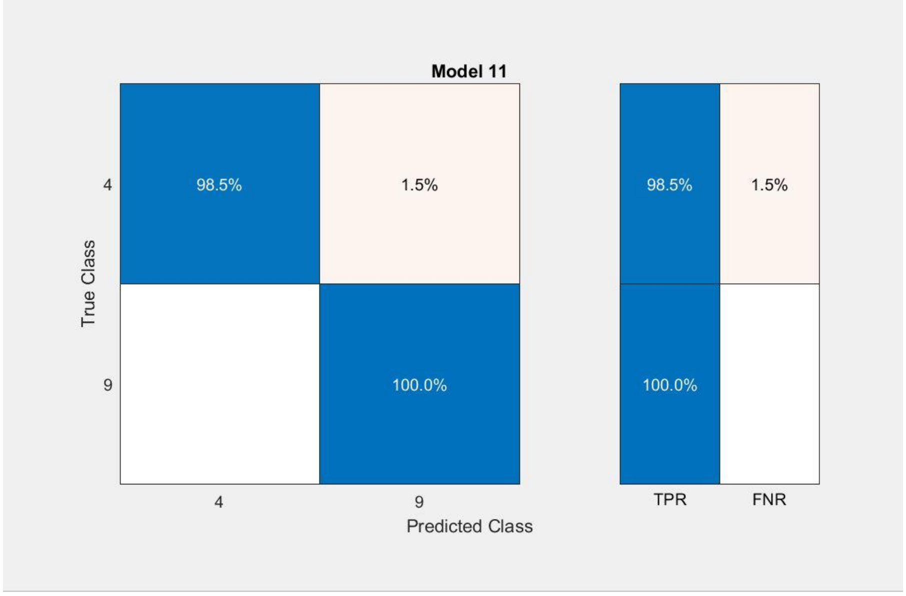

Figure A.xxi: Confusion Matrix of eleven features selected by mRMR feature selection algorithm

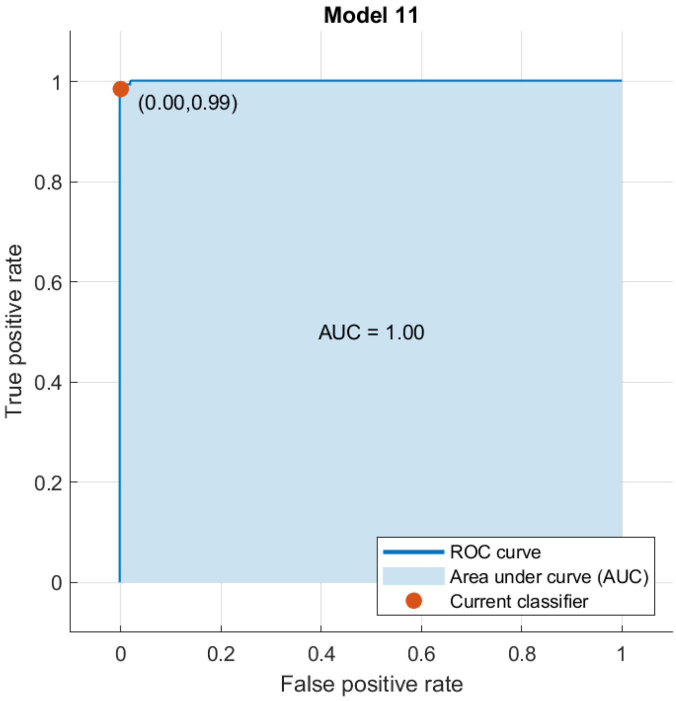

Figure A.xxii: ROC Curve of eleven features selected by mRMR feature selection algorithm

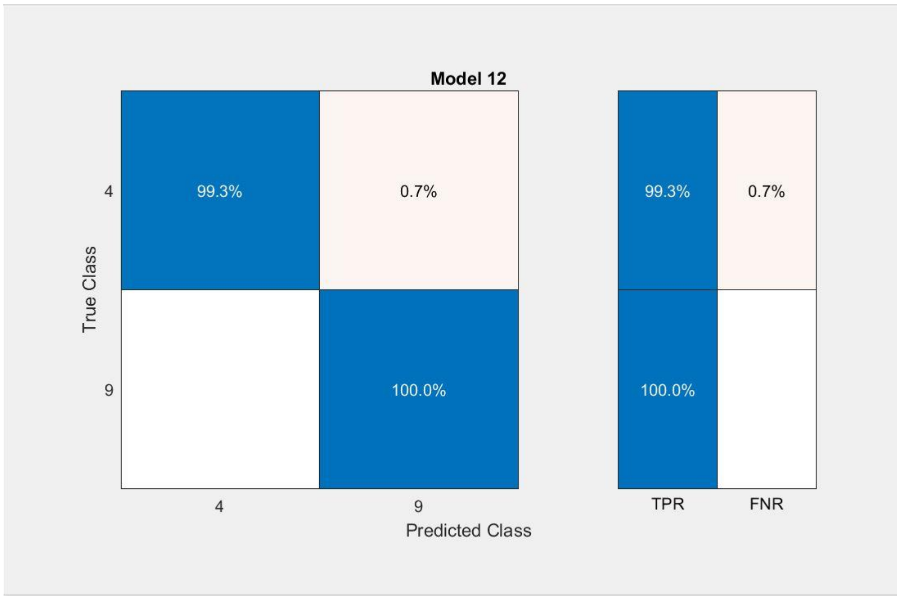

Figure A.xxiii: Confusion Matrix of twelve features selected by mRMR feature selection algorithm

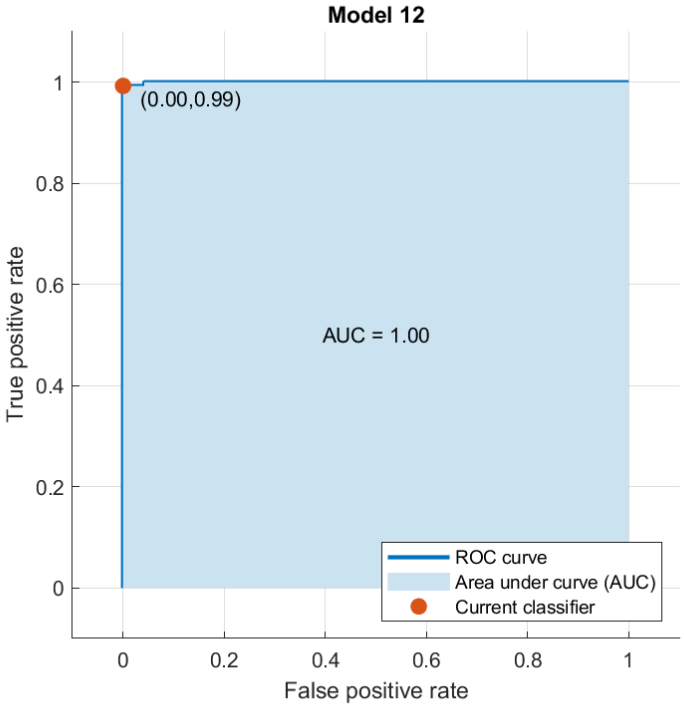

Figure A.xxiv: ROC Curve of twelve features selected by mRMR feature selection algorithm

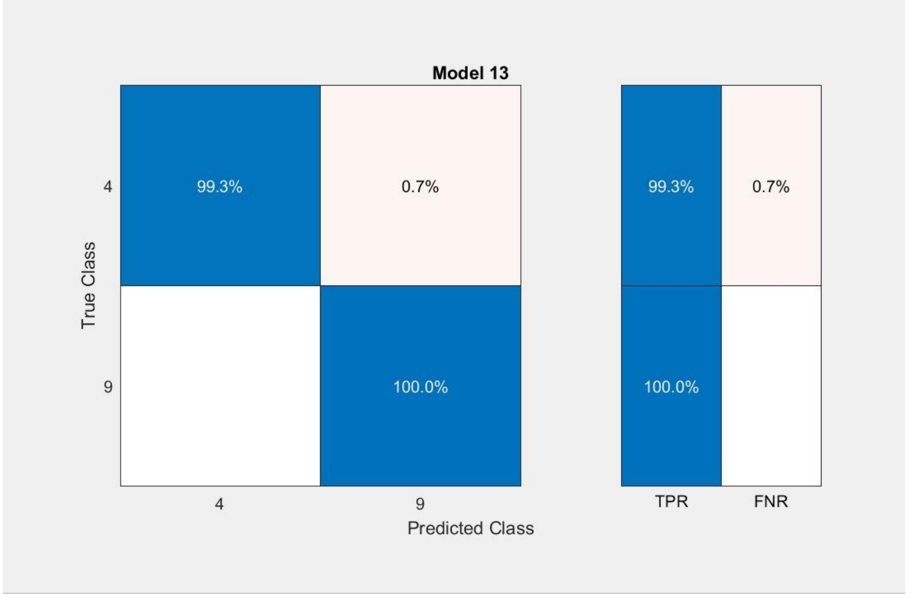

Figure A.xxv: Confusion Matrix of thirteen features selected by mRMR feature selection algorithm

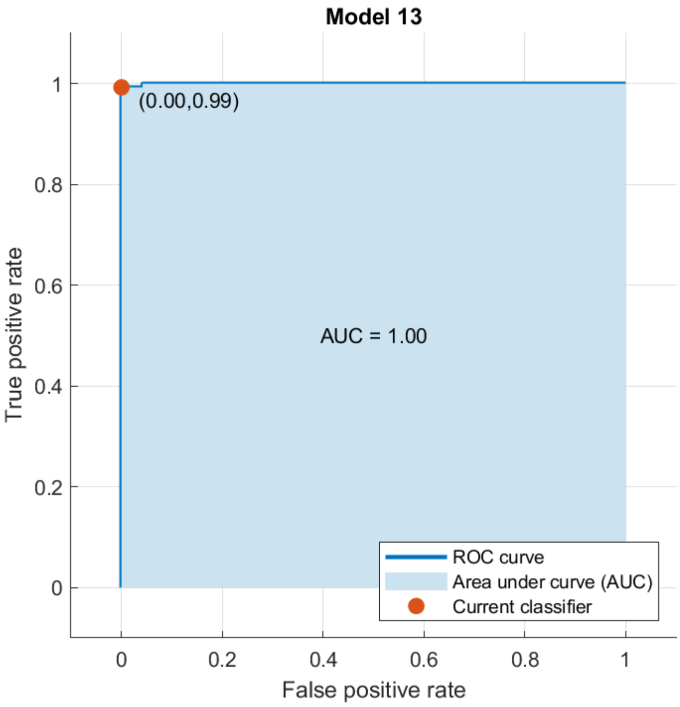

Figure A.xxvi: ROC Curve of thirteen features selected by mRMR feature selection algorithm

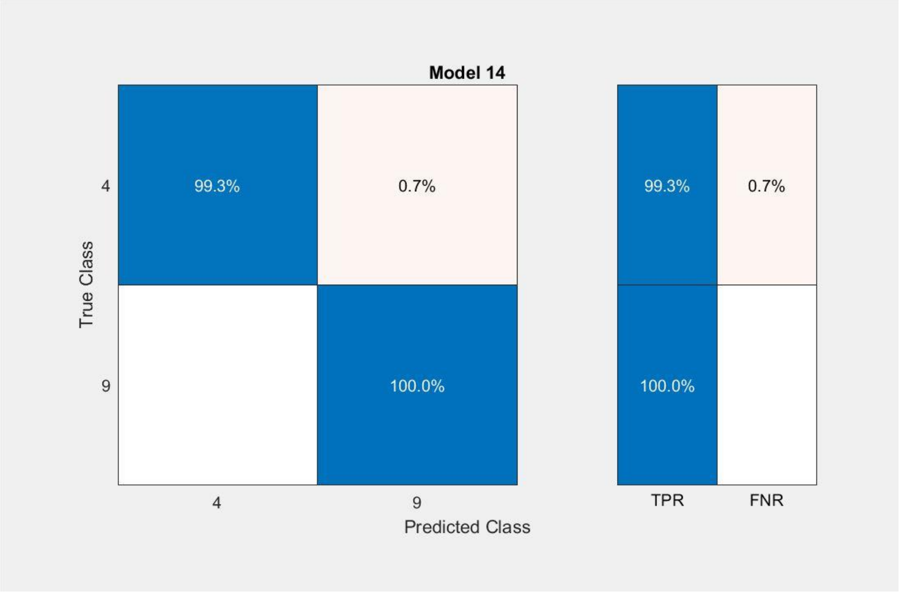

Figure A.xxvii: Confusion Matrix of fourteen features selected by mRMR feature selection algorithm

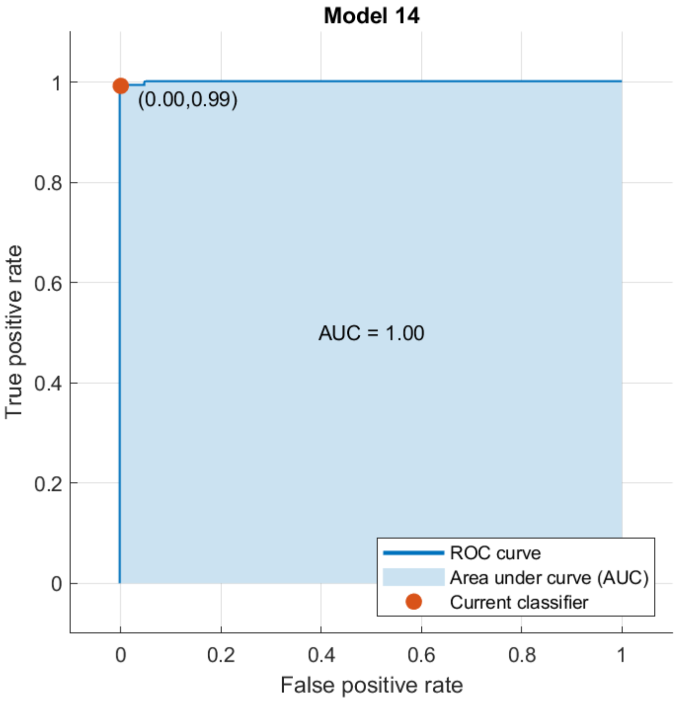

Figure A.xxviii: ROC Curve of fourteen features selected by mRMR feature selection algorithm

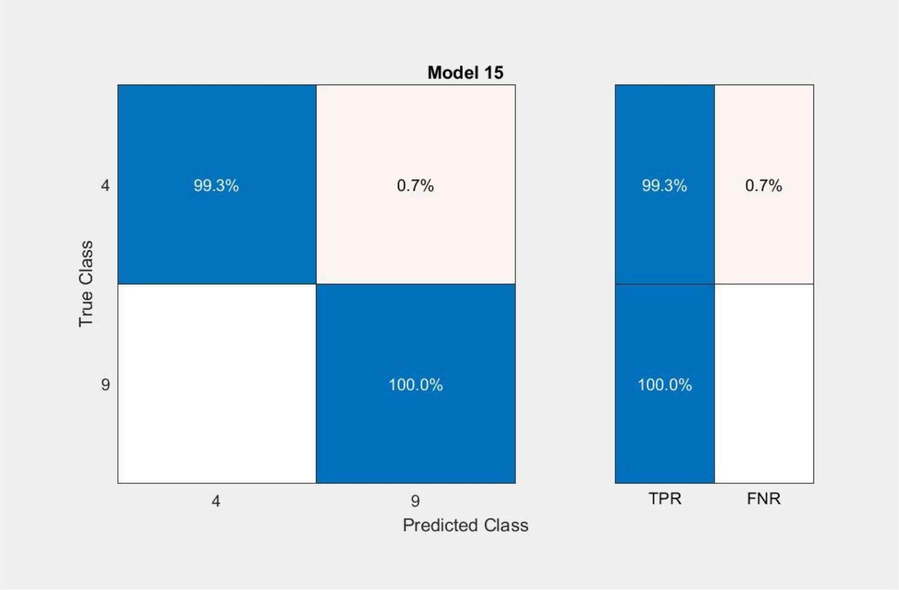

Figure A.xxvix: Confusion Matrix of fifteen features selected by mRMR feature selection algorithm

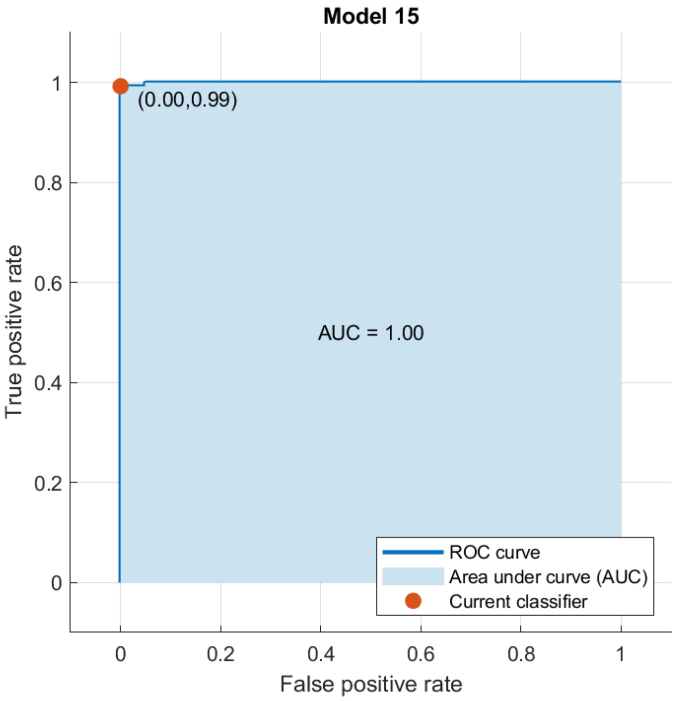

Figure A.xxx: ROC Curve of fifteen features selected by mRMR feature selection algorithm

APPENDIX B – CONFUSION MATRIX AND ROC CURVE FOR SVM-RFE FEATURE SELECTION ALGORITHM

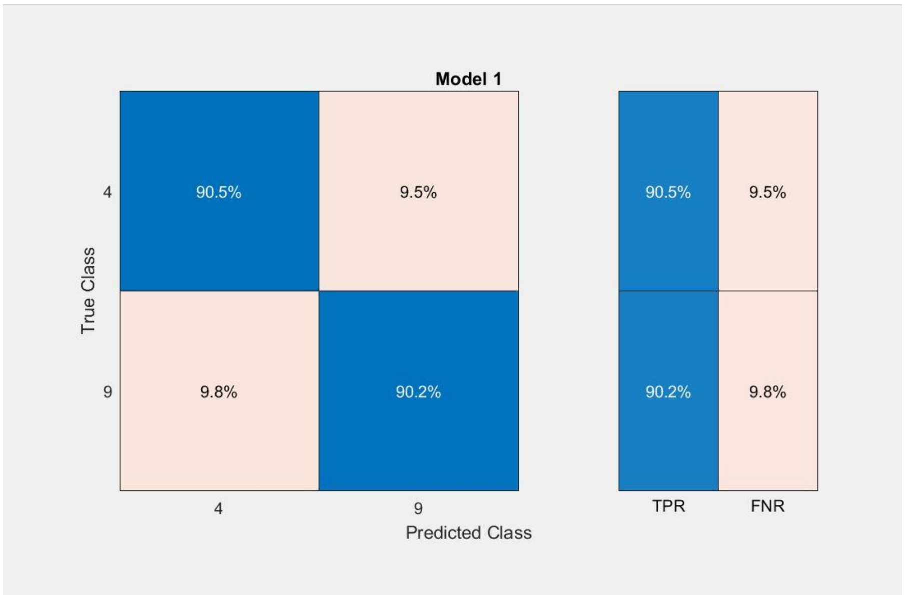

Figure B.i: Confusion Matrix of one feature selected by SVM-RFE feature selection algorithm

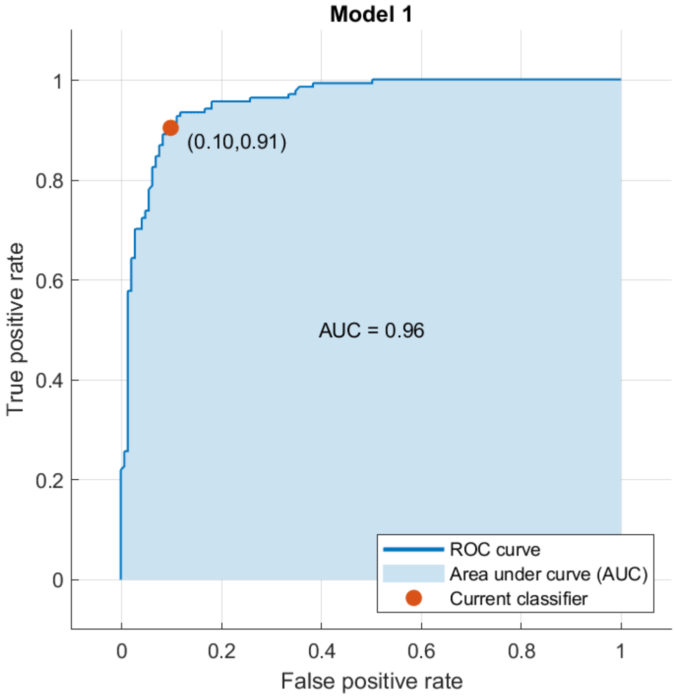

Figure B.ii: ROC Curve of one feature selected by SVM-RFE feature selection algorithm

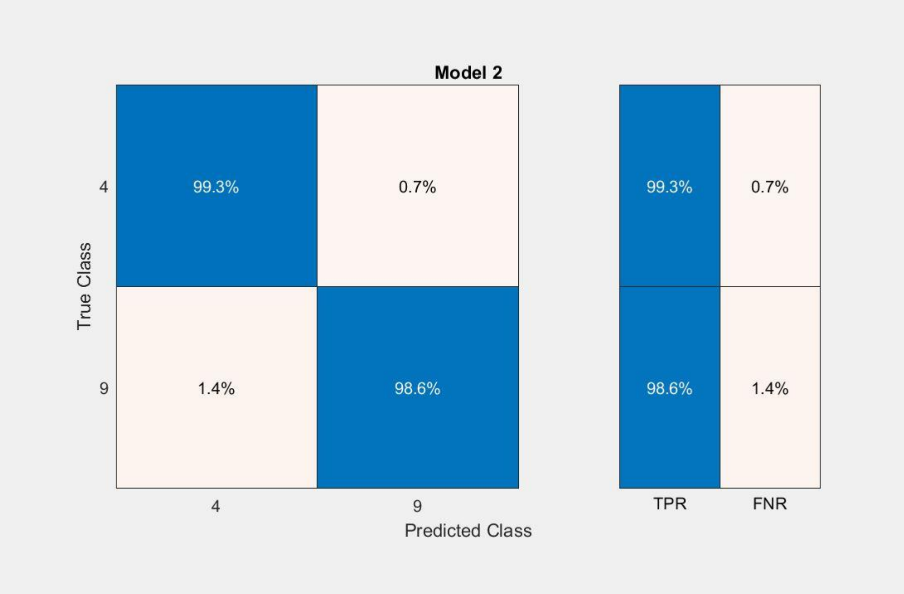

Figure B.iii: Confusion Matrix of two features selected by SVM-RFE feature selection algorithm

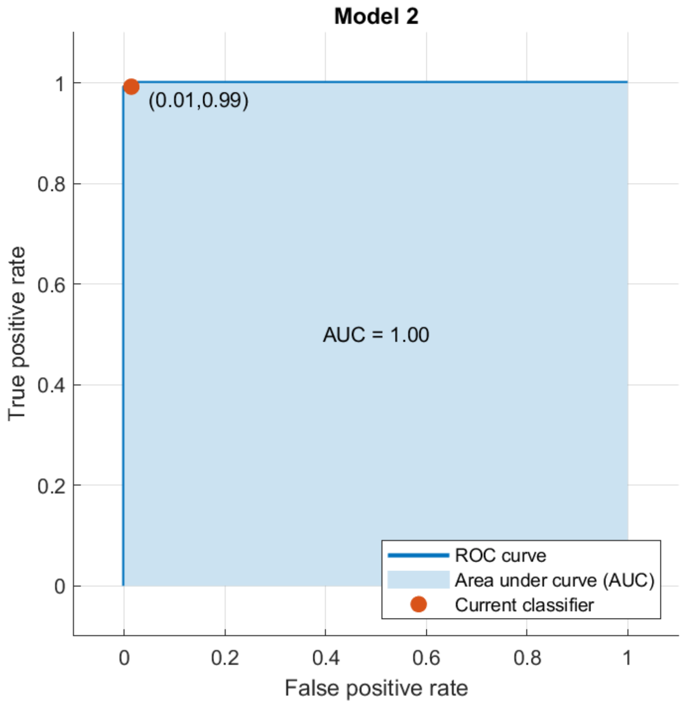

Figure B.iv: ROC Curve of two features selected by SVM-RFE feature selection algorithm

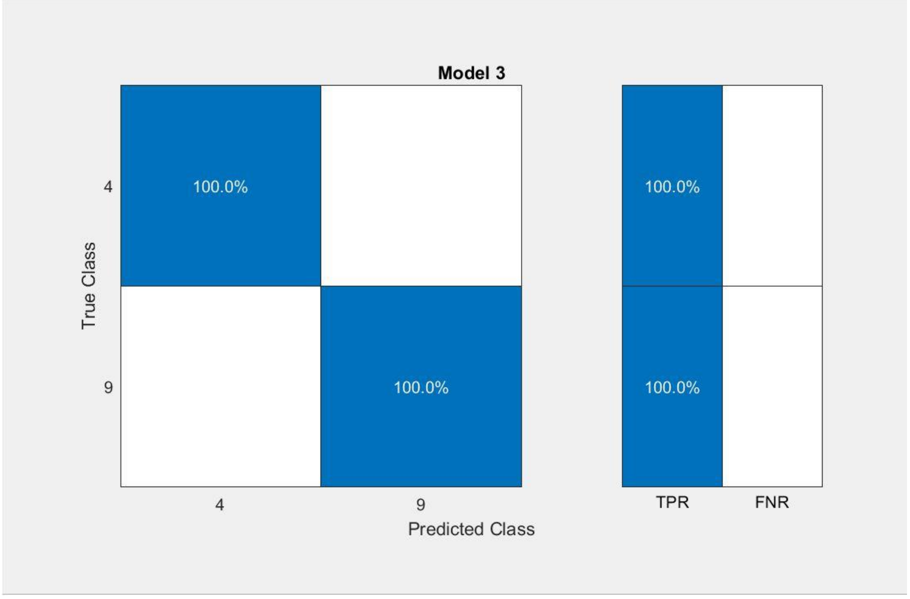

Figure B.v: Confusion Matrix of three features selected by SVM-RFE feature selection algorithm

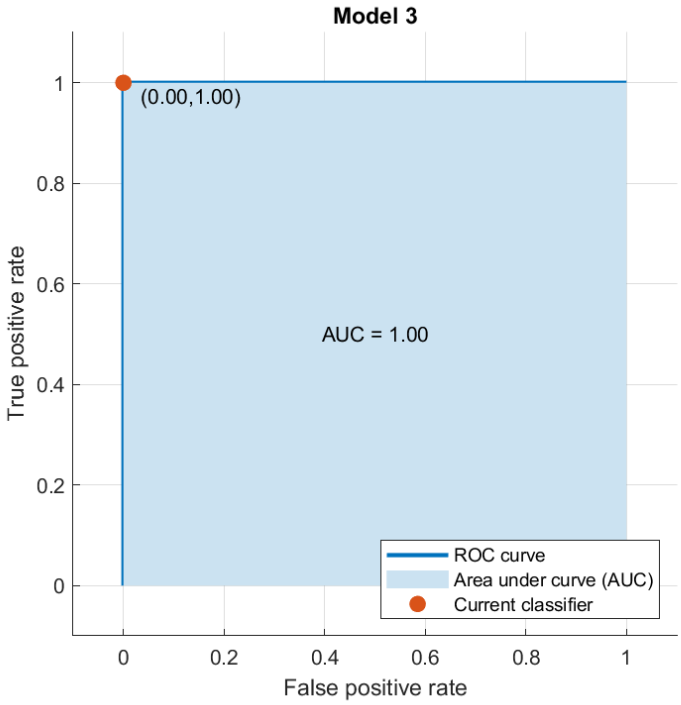

Figure B.vi: ROC Curve of three features selected by SVM-RFE feature selection algorithm

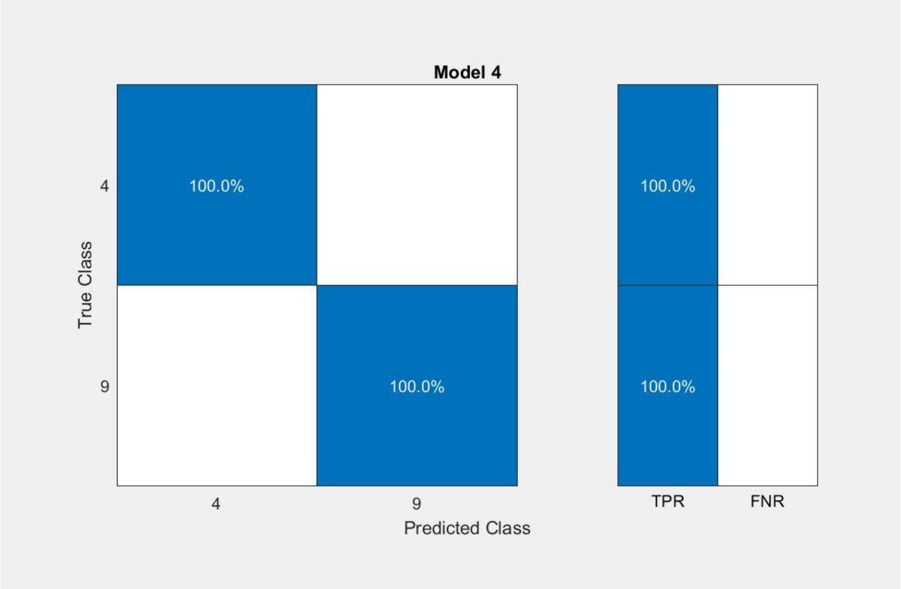

Figure B.vii: Confusion Matrix of four features selected by SVM-RFE feature selection algorithm

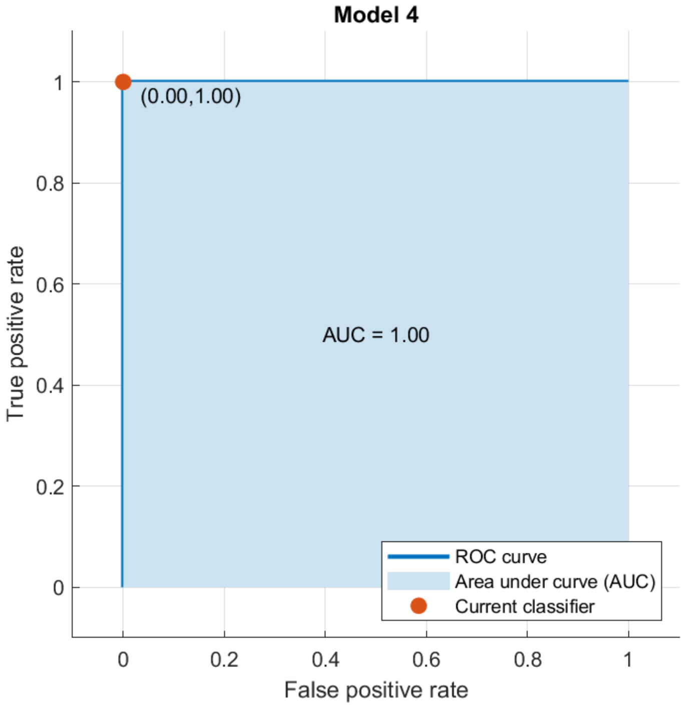

Figure B.viii: ROC Curve of four features selected by SVM-RFE feature selection algorithm

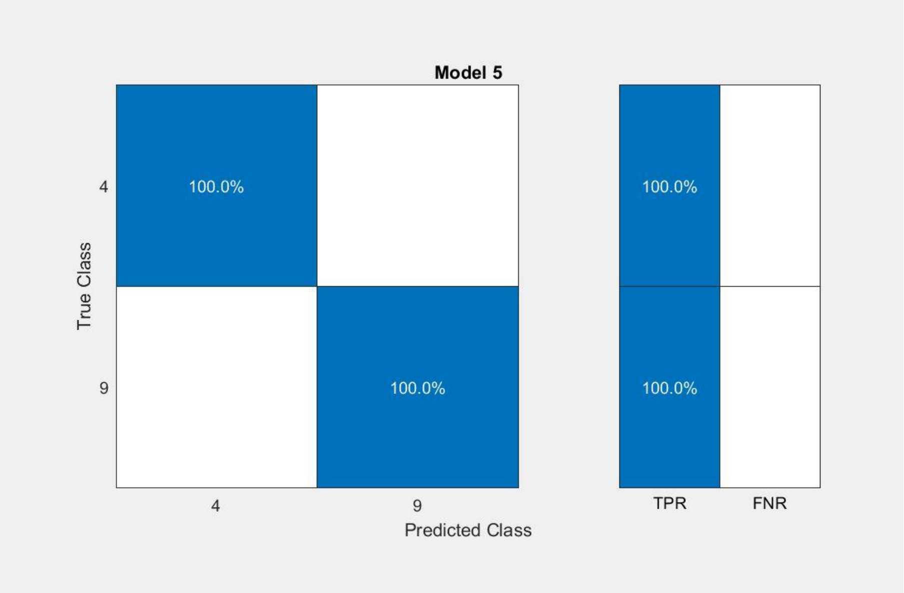

Figure B.ix: Confusion Matrix of five features selected by SVM-RFE feature selection algorithm

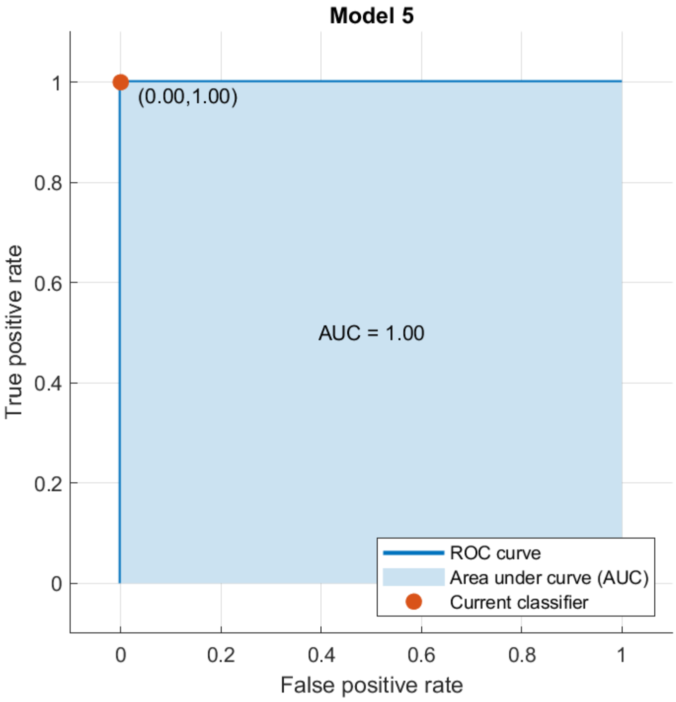

Figure B.x: ROC Curve of five features selected by SVM-RFE feature selection algorithm

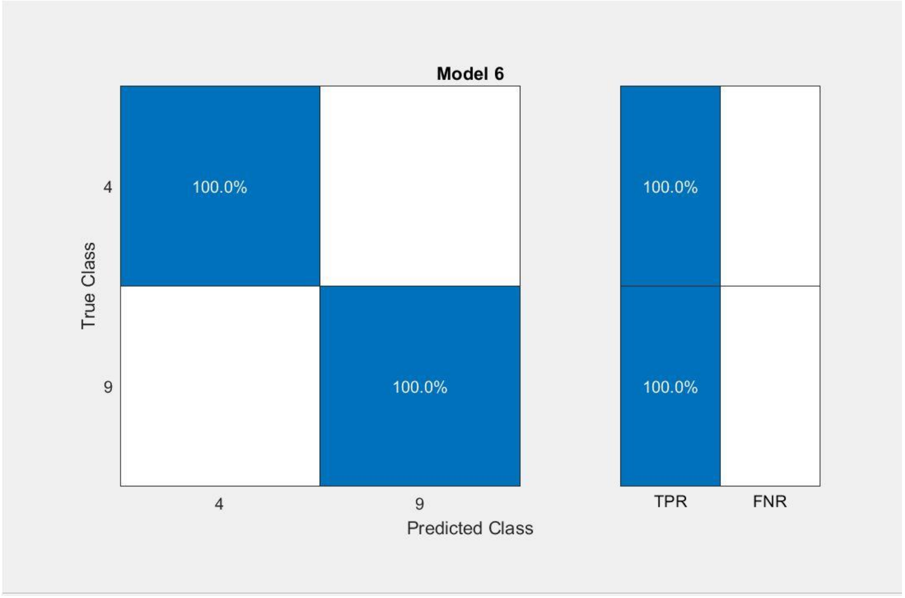

Figure B.xi: Confusion Matrix of six features selected by SVM-RFE feature selection algorithm

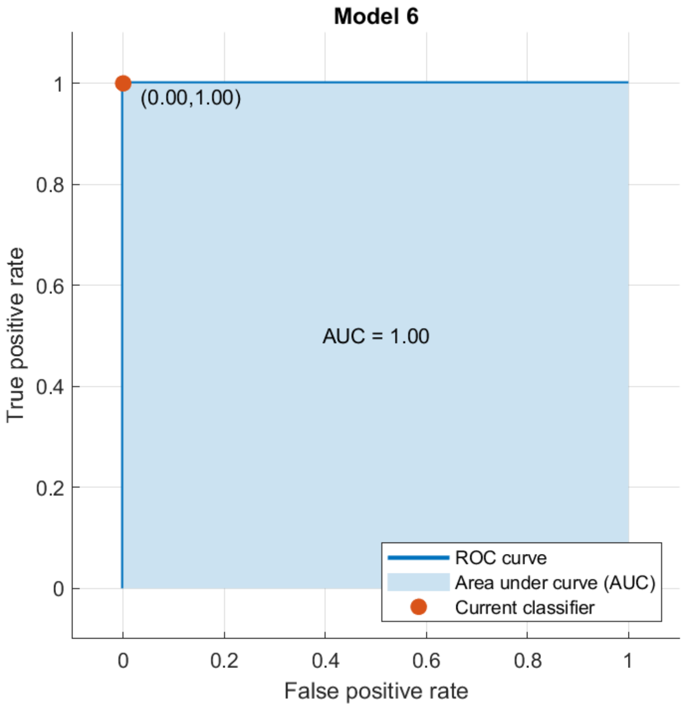

Figure B.xii: ROC Curve of six features selected by SVM-RFE feature selection algorithm

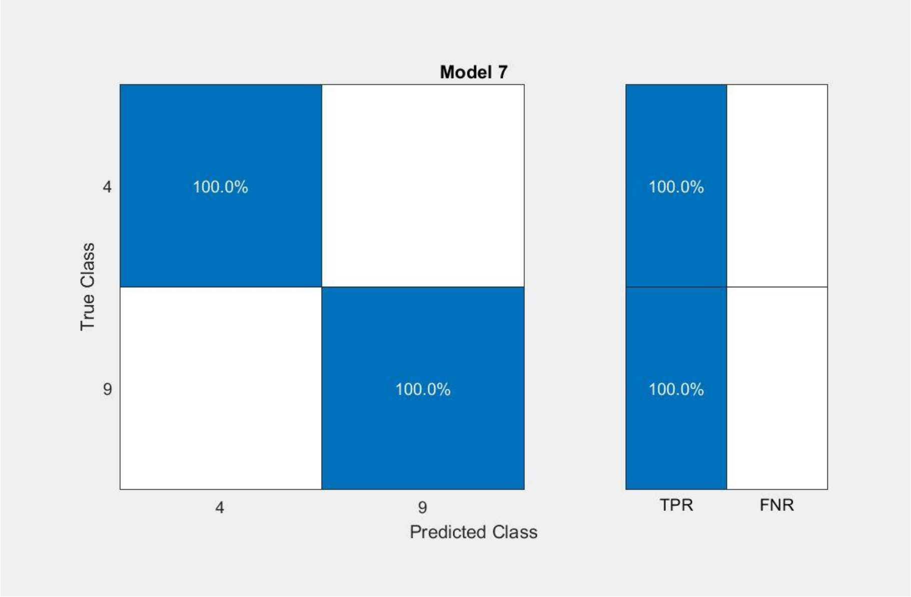

Figure B.xiii: Confusion Matrix of seven features selected by SVM-RFE feature selection algorithm

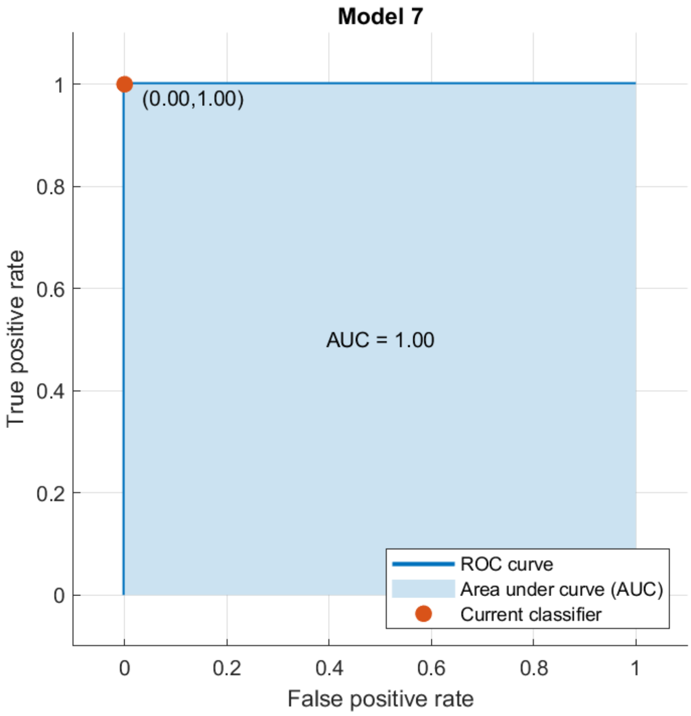

Figure B.xiv: ROC Curve of seven features selected by SVM-RFE feature selection algorithm

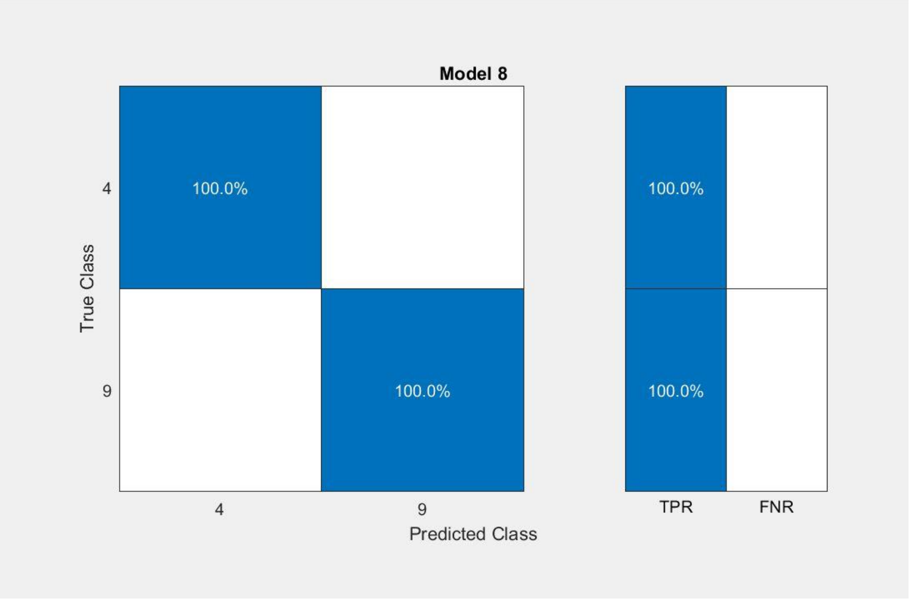

Figure B.xv: Confusion Matrix of eight features selected by SVM-RFE feature selection algorithm

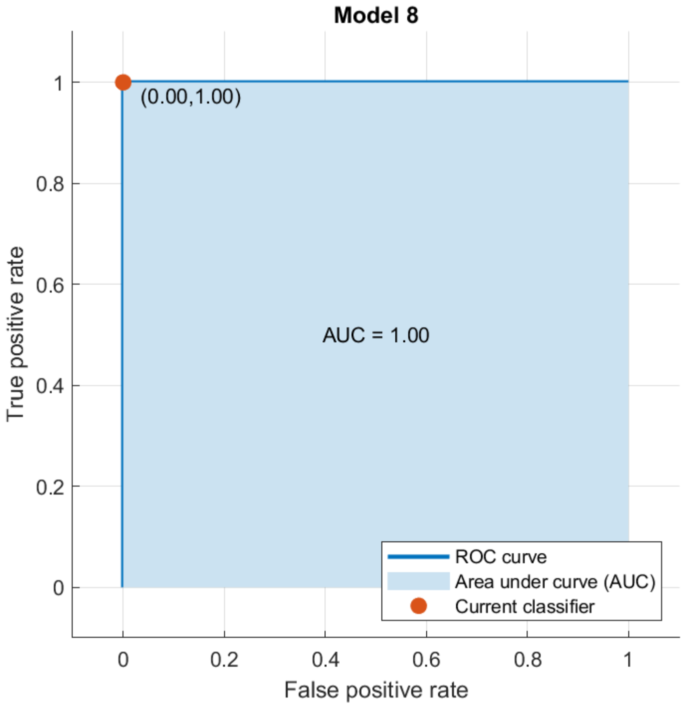

Figure B.xvi: ROC Curve of eight features selected by SVM-RFE feature selection algorithm

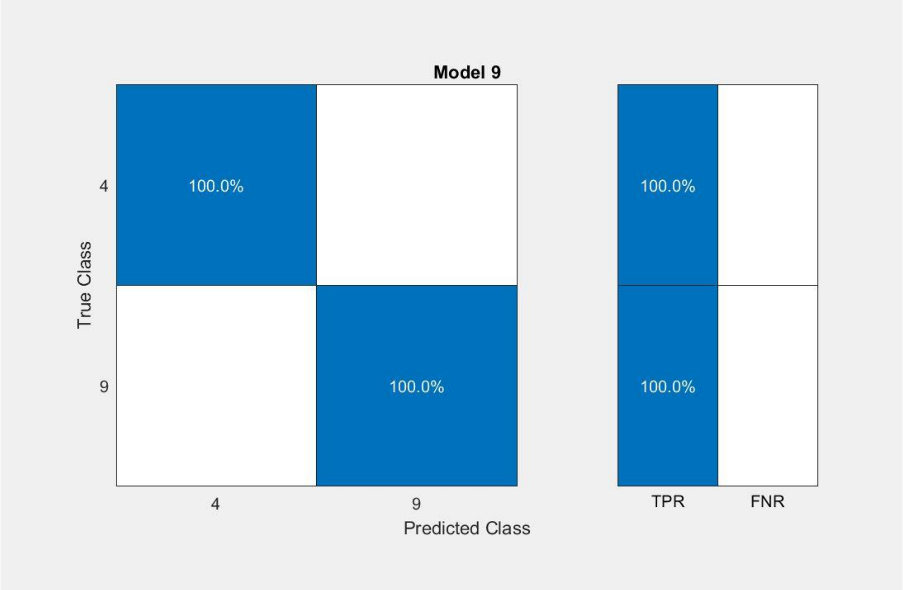

Figure B.xvii: Confusion Matrix of nine features selected by SVM-RFE feature selection algorithm

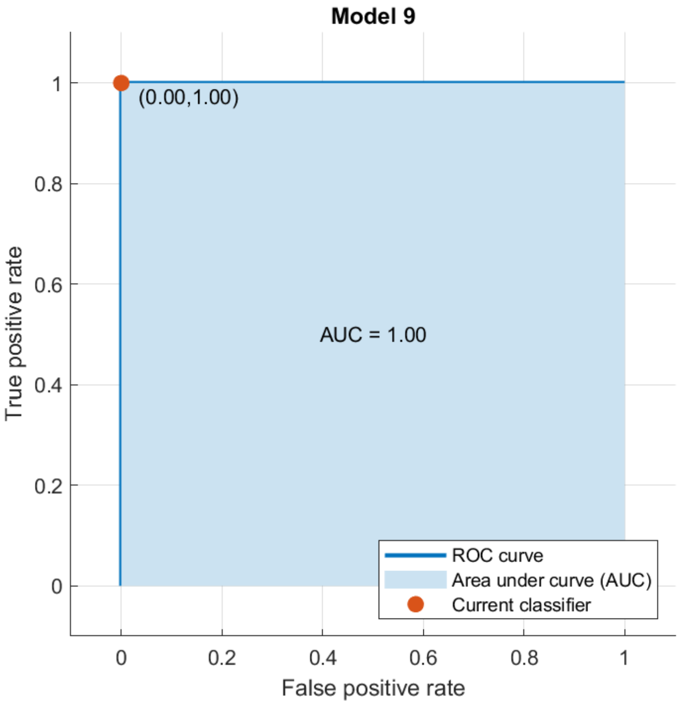

Figure B.xviii: ROC Curve of nine features selected by SVM-RFE feature selection algorithm

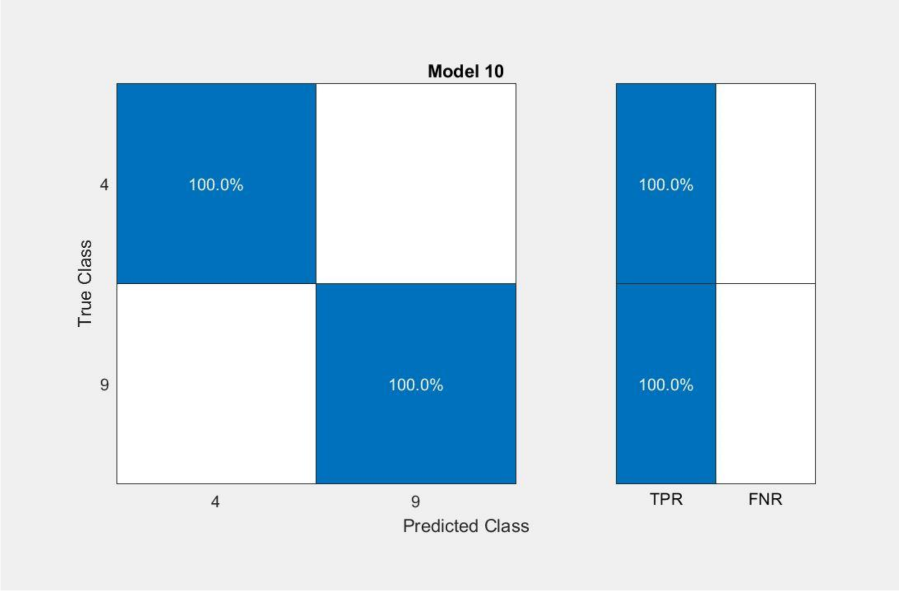

Figure B.xvix: Confusion Matrix of ten features selected by SVM-RFE feature selection algorithm

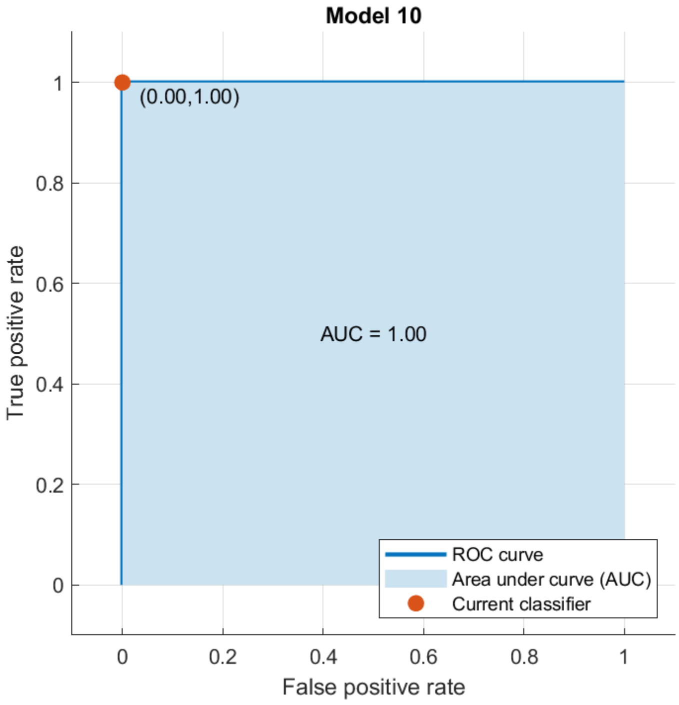

Figure B.xx: ROC Curve of ten features selected by SVM-RFE feature selection algorithm

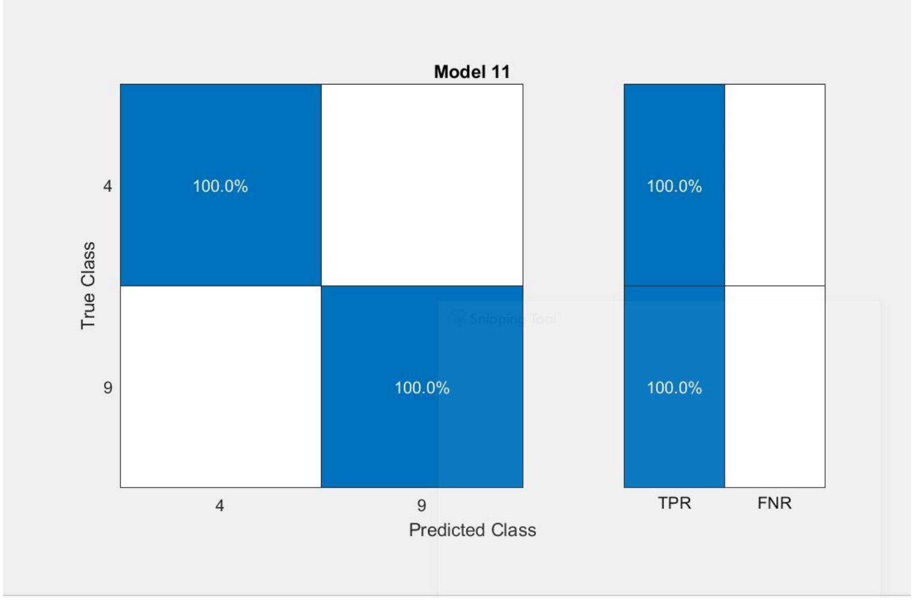

Figure B.xxi: Confusion Matrix of eleven features selected by SVM-RFE feature selection algorithm

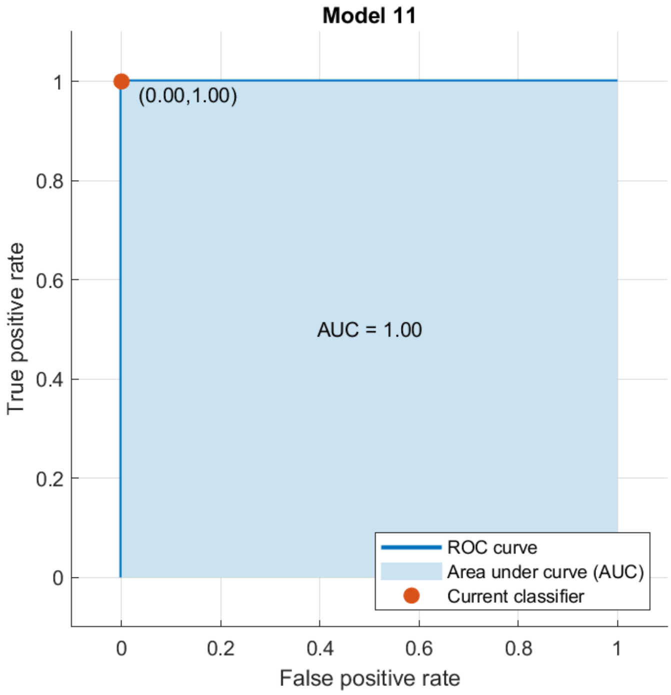

Figure B.xxii: ROC Curve of eleven features selected by SVM-RFE feature selection algorithm

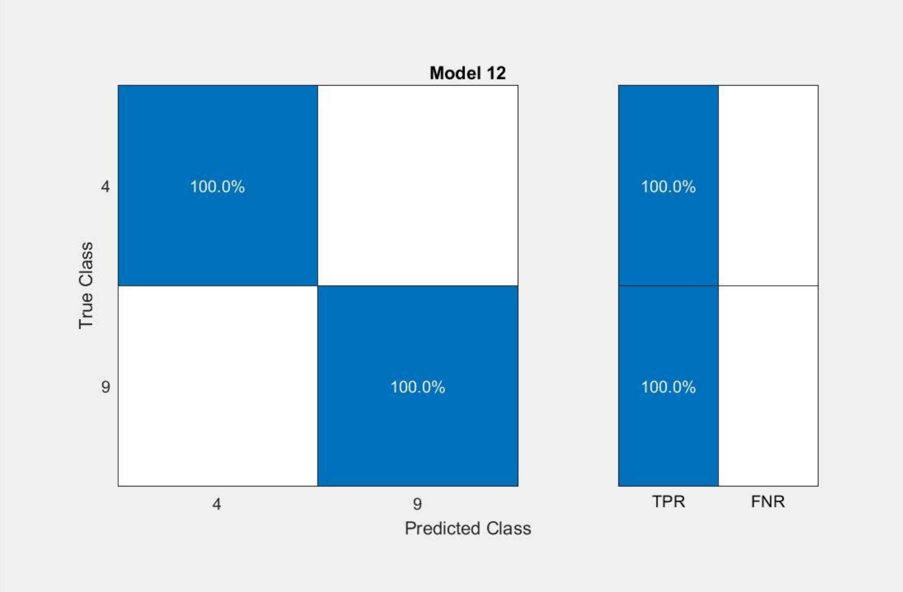

Figure B.xxiii: Confusion Matrix of twelve features selected by SVM-RFE feature selection algorithm

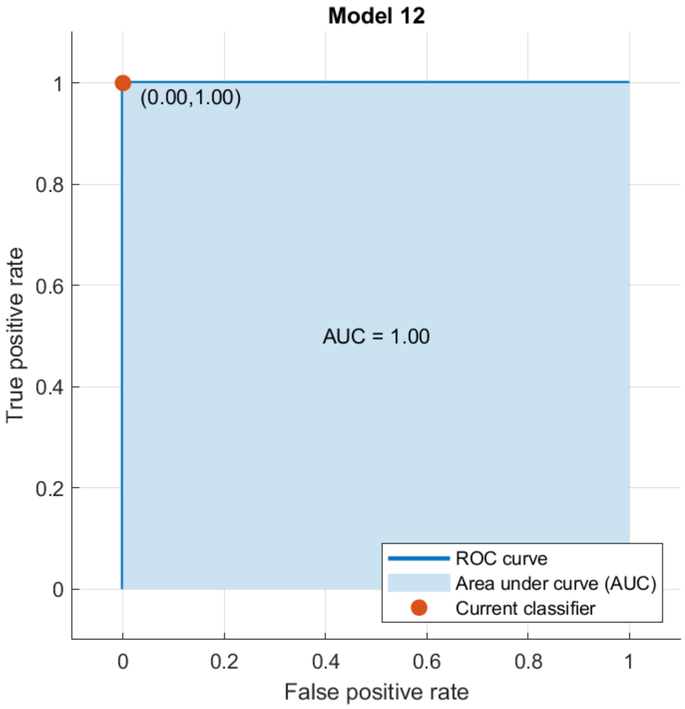

Figure B.xxiv: ROC Curve of twelve features selected by SVM-RFE feature selection algorithm

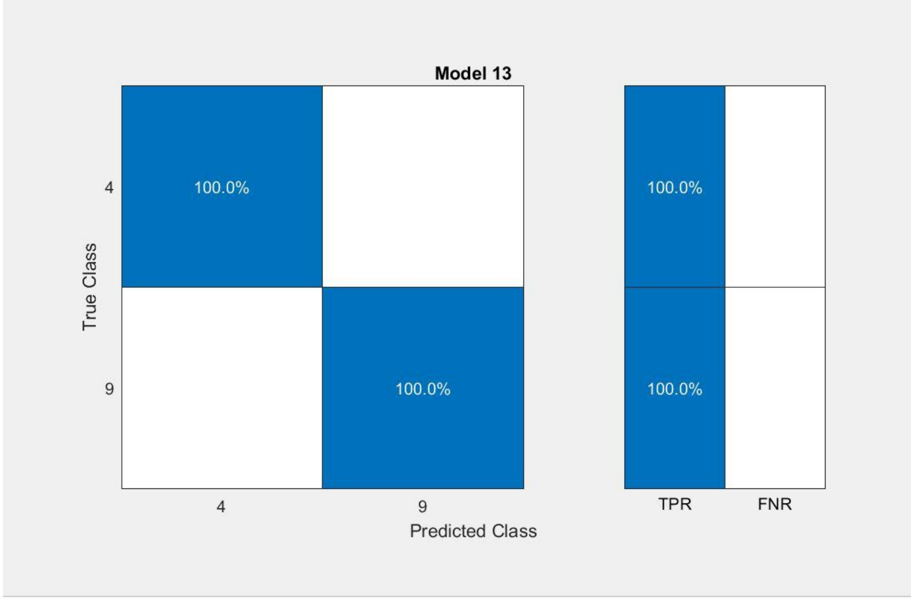

Figure B.xxv: Confusion Matrix of thirteen features selected by SVM-RFE feature selection algorithm

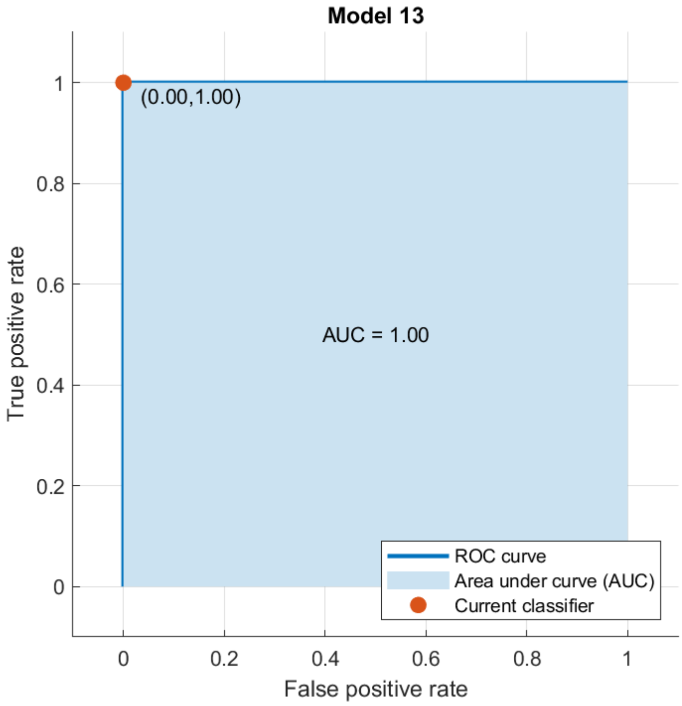

Figure B.xxvi: ROC Curve of thirteen features selected by SVM-RFE feature selection algorithm

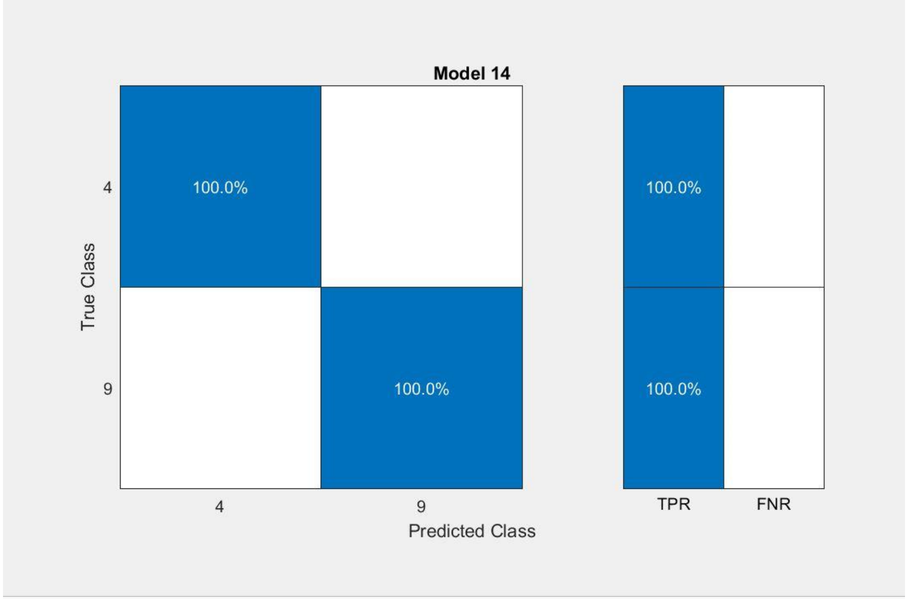

Figure B.xxvii: Confusion Matrix of fourteen features selected by SVM-RFE feature selection algorithm

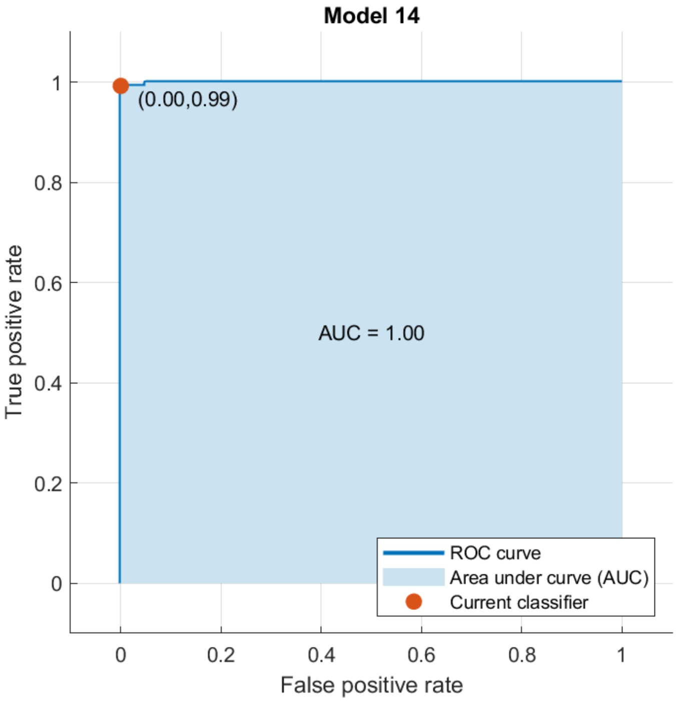

Figure B.xxviii: ROC Curve of fourteen features selected by SVM-RFE feature selection algorithm

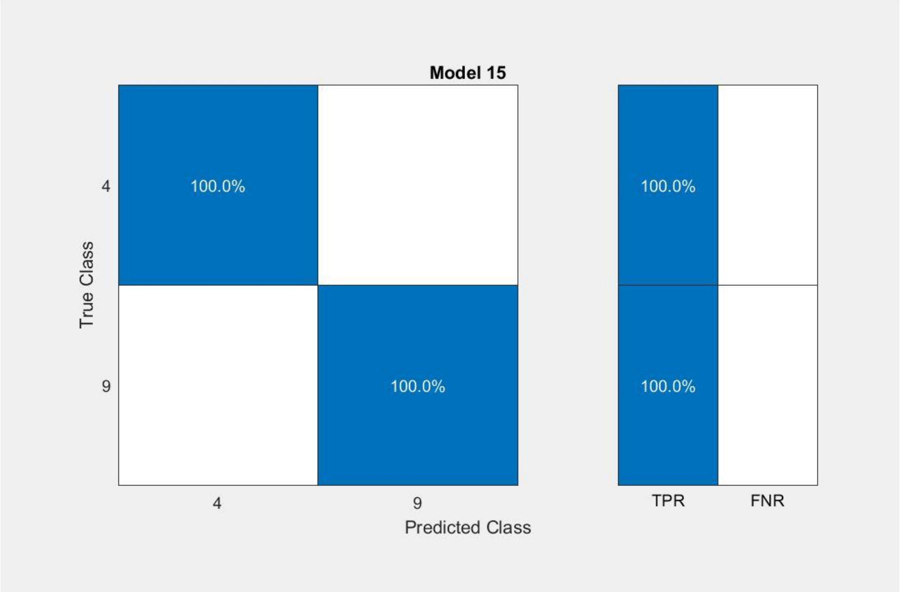

Figure B.xxvix: Confusion Matrix of fifteen features selected by SVM-RFE feature selection algorithm

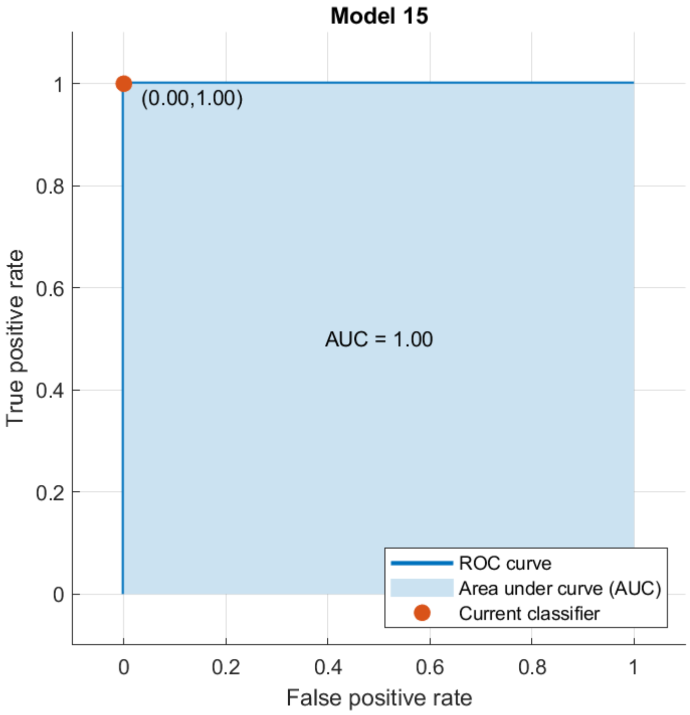

Figure B.xxx: ROC Curve of fifteen features selected by SVM-RFE feature selection algorithm

APPENDIX C – CONFUSION MATRIX AND ROC CURVE FOR PROPOSED HYBRID FEATURE SELECTION ALGORITHM

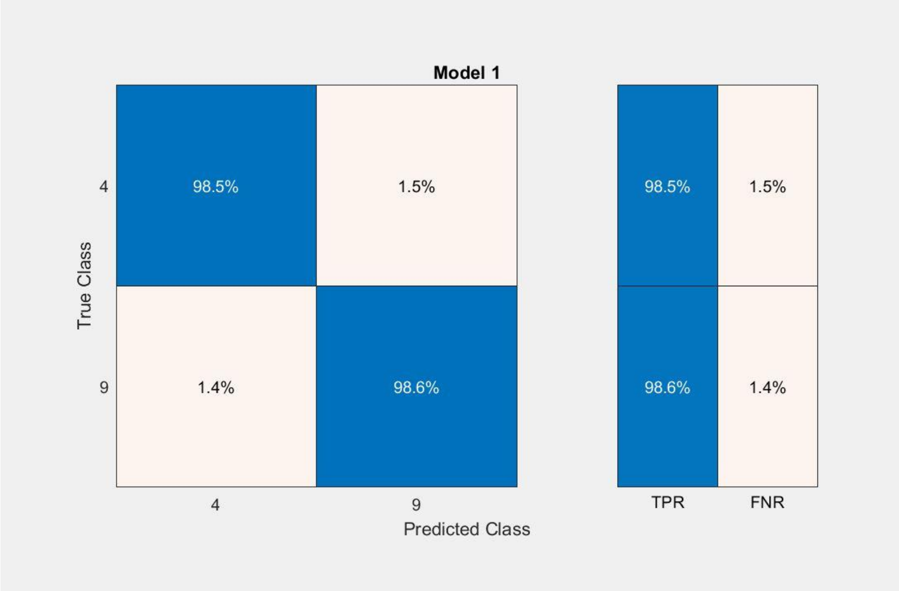

Figure C.i: Confusion Matrix of one feature selected by Hybrid feature selection algorithm

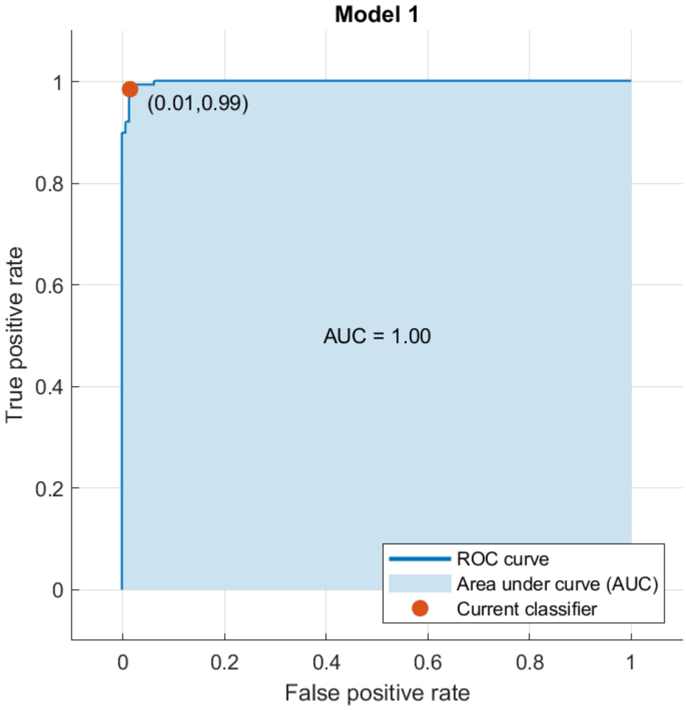

Figure C.ii: ROC Curve of one feature selected by Hybrid feature selection algorithm

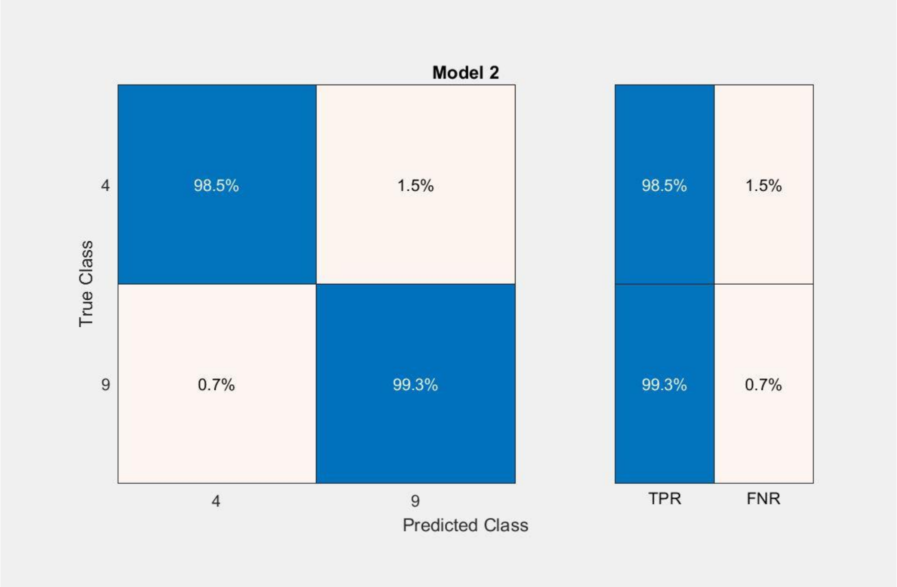

Figure C.iii: Confusion Matrix of two features selected by Hybrid feature selection algorithm

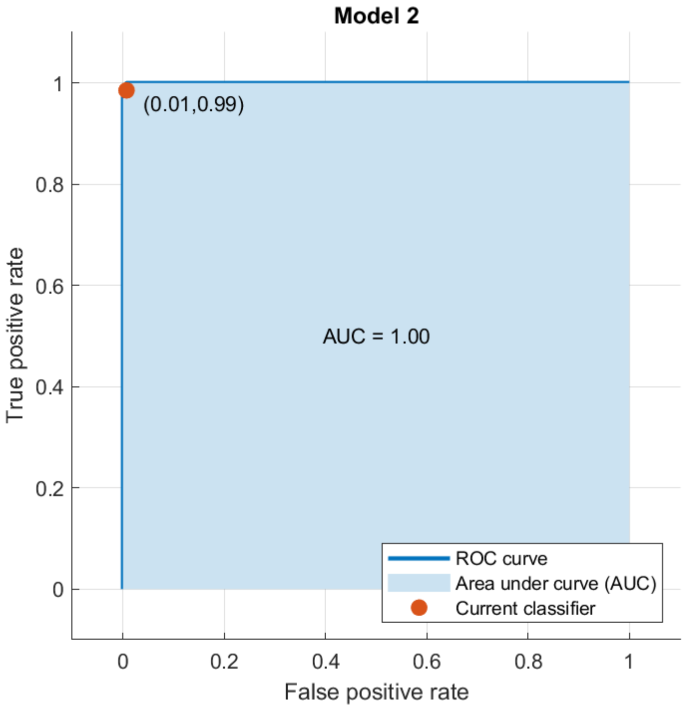

Figure C.iv: ROC Curve of two features selected by Hybrid feature selection algorithm

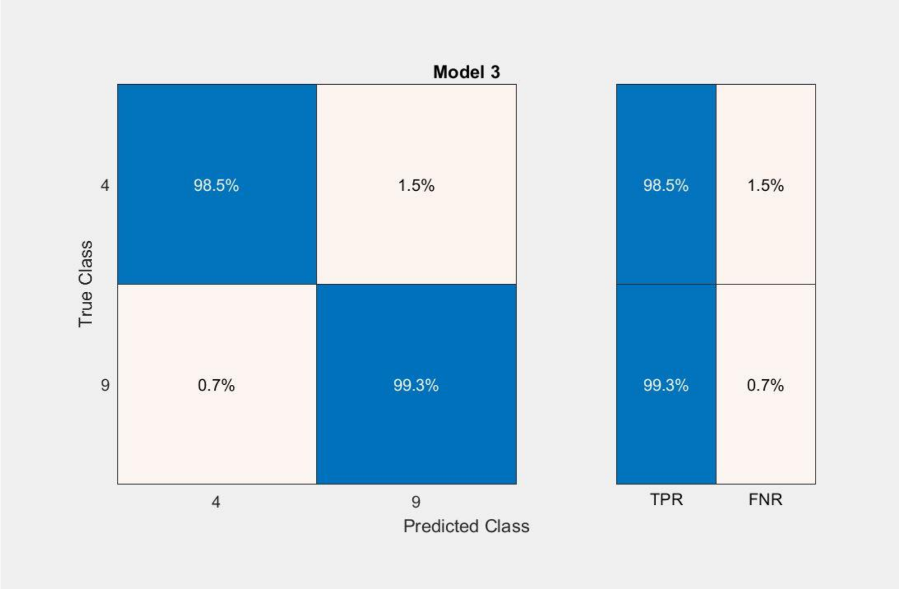

Figure C.v: Confusion Matrix of three features selected by Hybrid feature selection algorithm

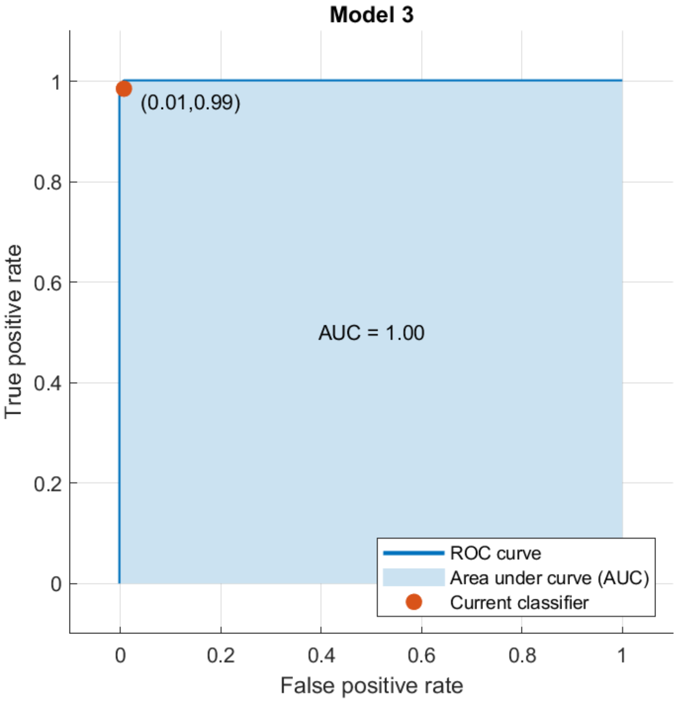

Figure C.vi: ROC Curve of three features selected by Hybrid feature selection algorithm

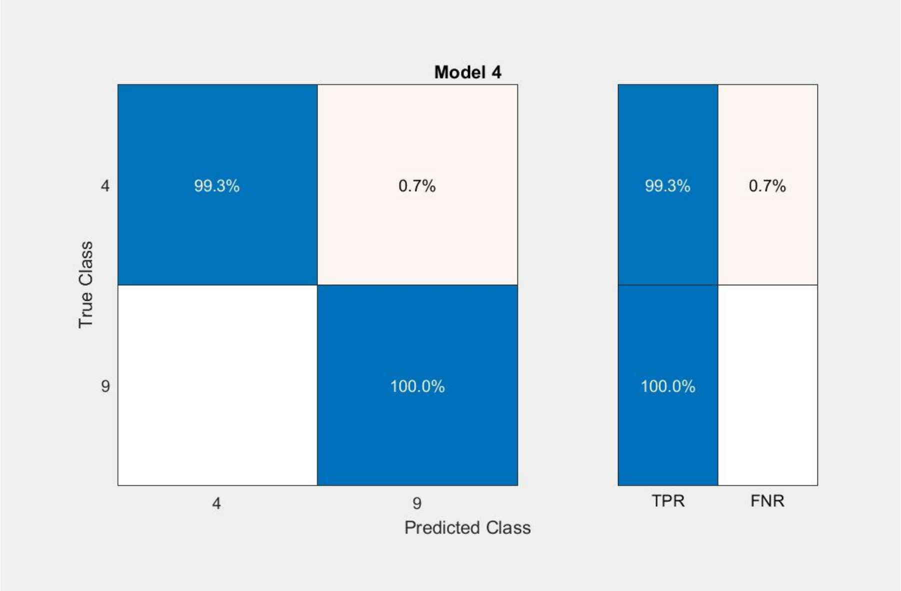

Figure C.vii: Confusion Matrix of four features selected by Hybrid feature selection algorithm

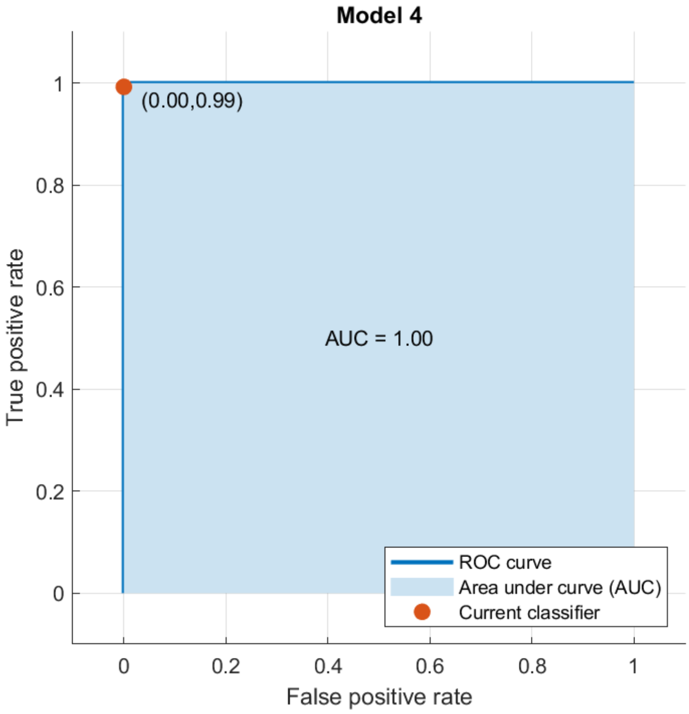

Figure C.viii: ROC Curve of four features selected by Hybrid feature selection algorithm

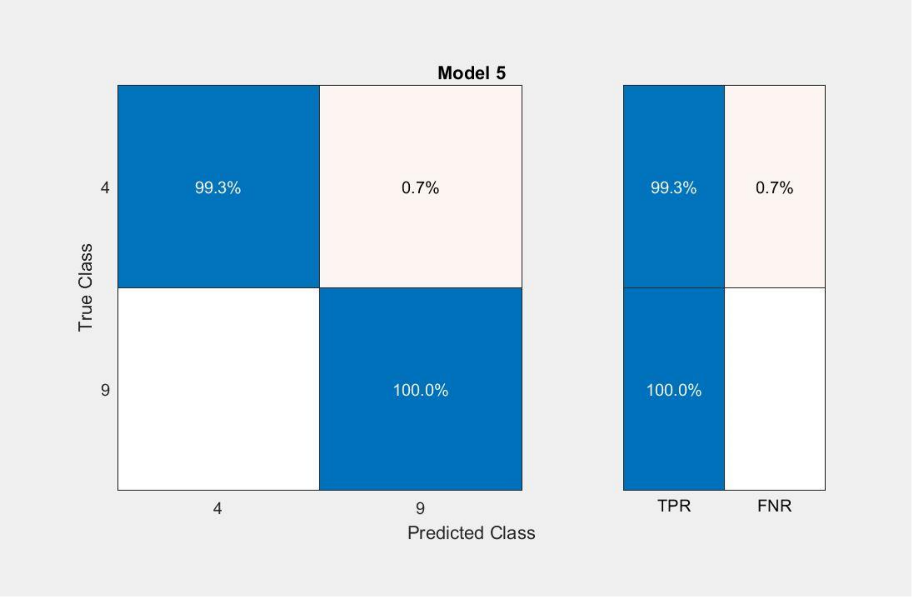

Figure C.ix: Confusion Matrix of five features selected by Hybrid feature selection algorithm

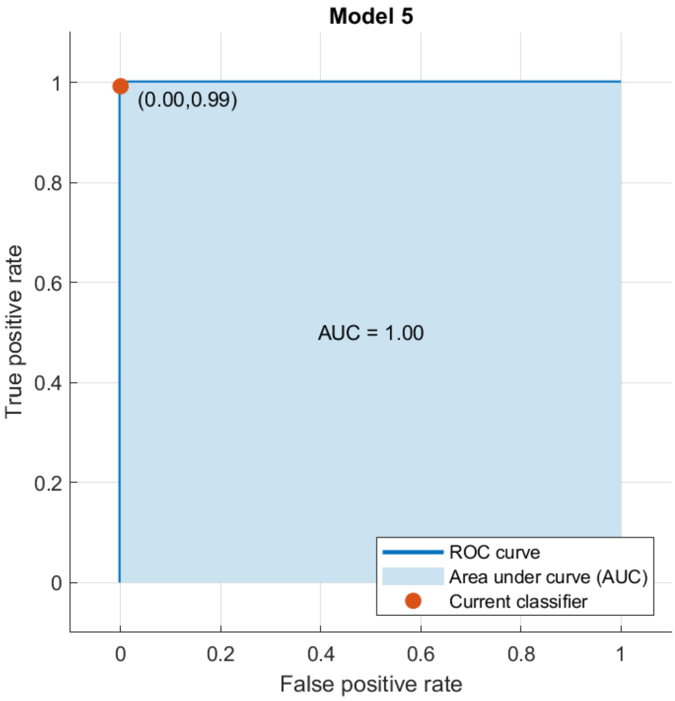

Figure C.x: ROC Curve of five features selected by Hybrid feature selection algorithm

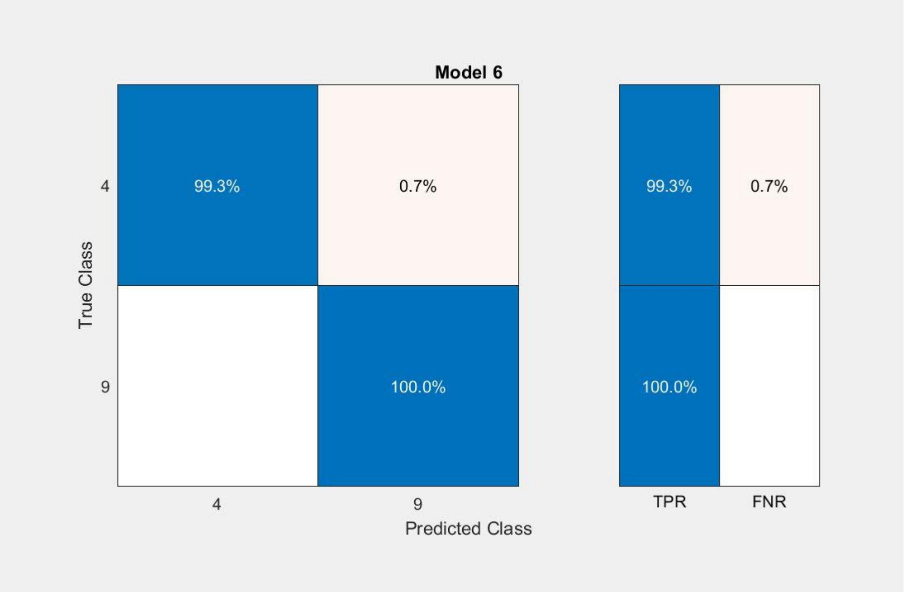

Figure C.xi: Confusion Matrix of six features selected by Hybrid feature selection algorithm

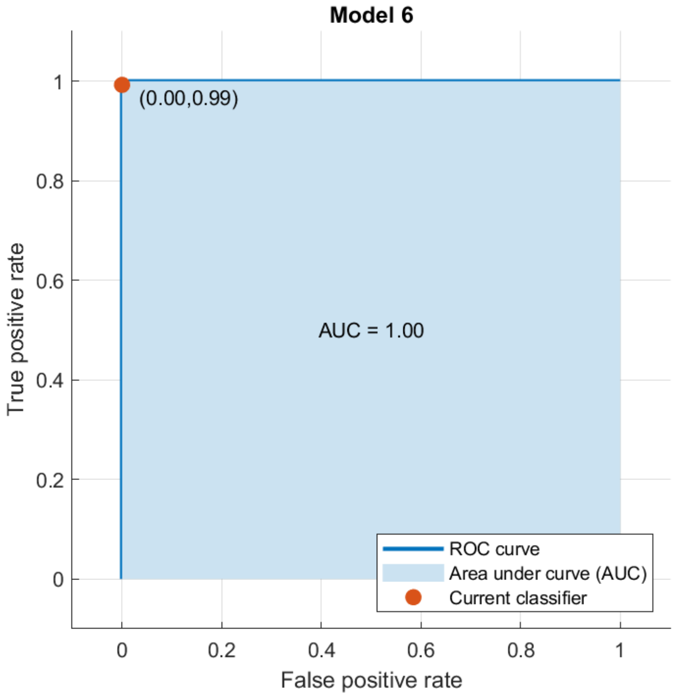

Figure C.xii: ROC Curve of six features selected by Hybrid feature selection algorithm

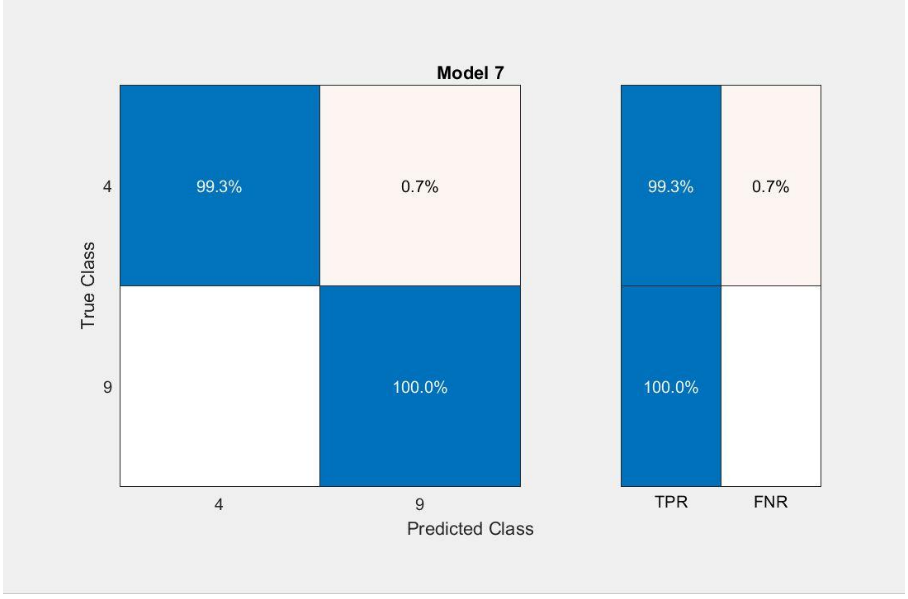

Figure C.xiii: Confusion Matrix of seven features selected by Hybrid feature selection algorithm

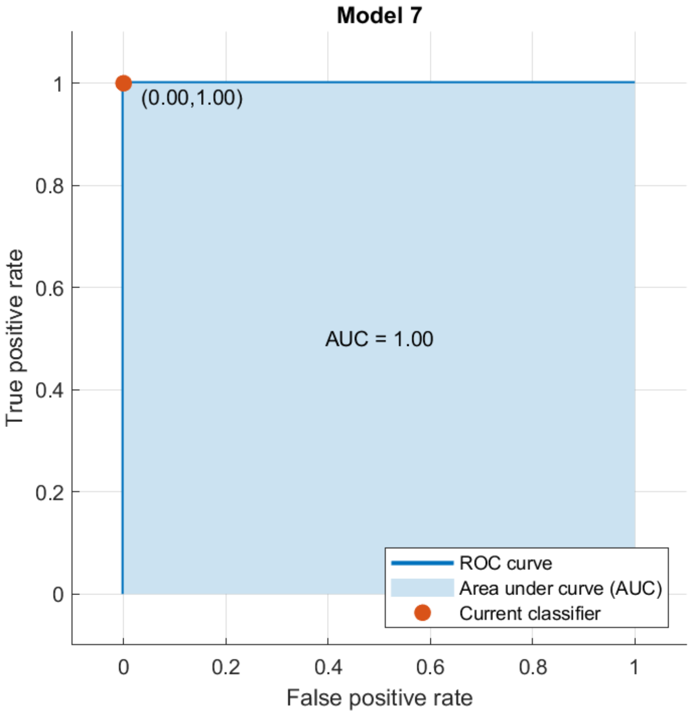

Figure C.xiv: ROC Curve of seven features selected by Hybrid feature selection algorithm

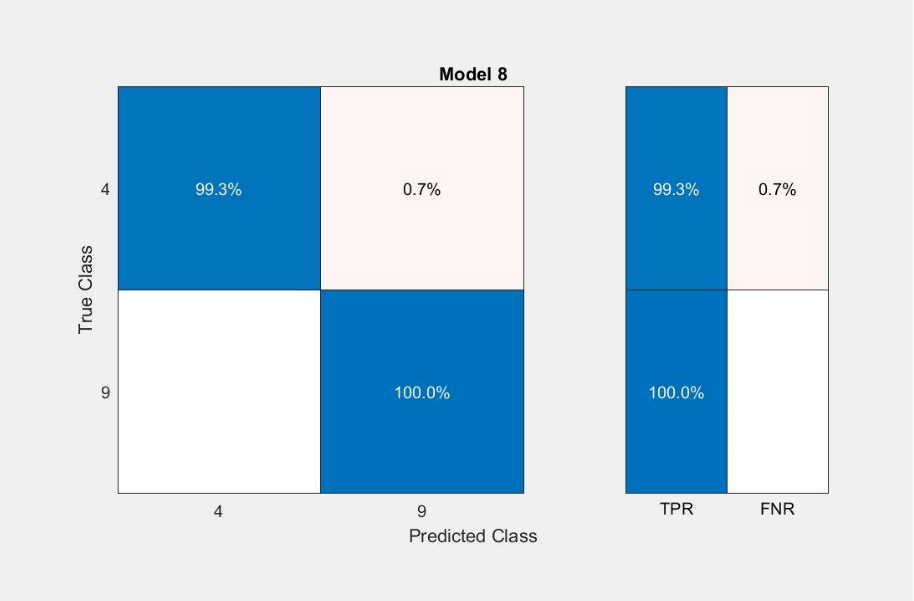

Figure C.xv: Confusion Matrix of eight features selected by Hybrid feature selection algorithm

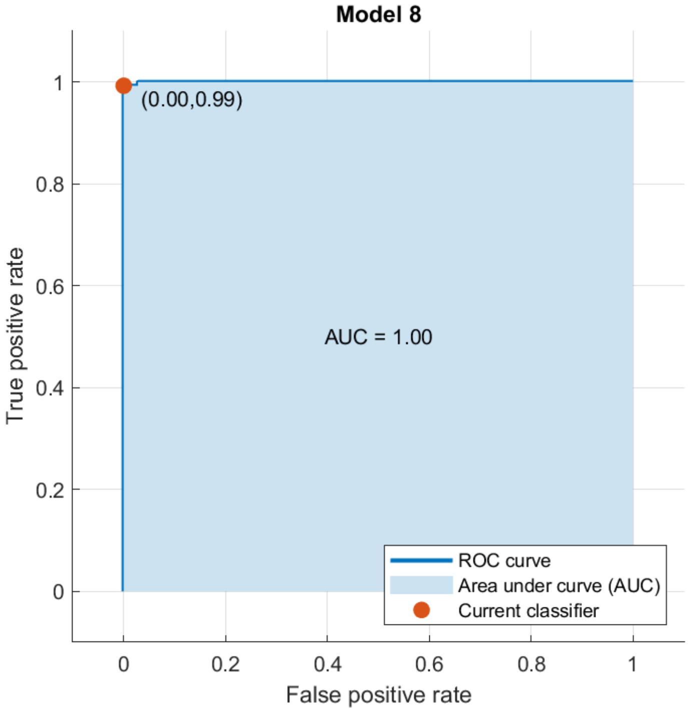

Figure C.xvi: ROC Curve of eight features selected by Hybrid feature selection algorithm

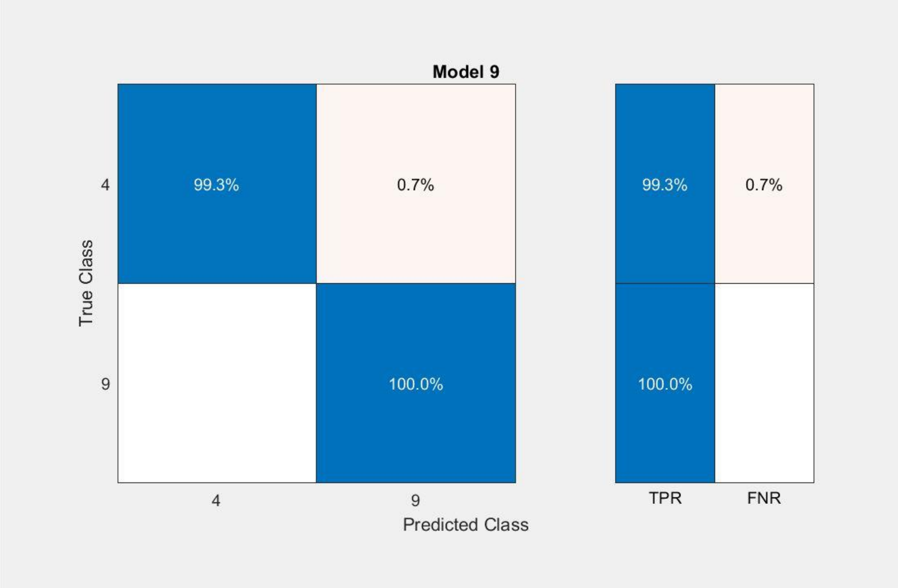

Figure C.xvii: Confusion Matrix of nine features selected by Hybrid feature selection algorithm

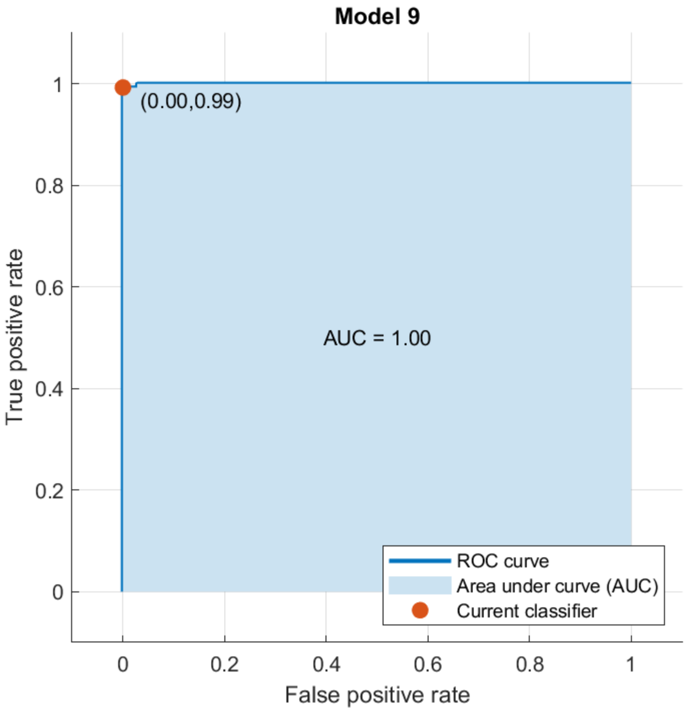

Figure C.xviii: ROC Curve of nine features selected by Hybrid feature selection algorithm

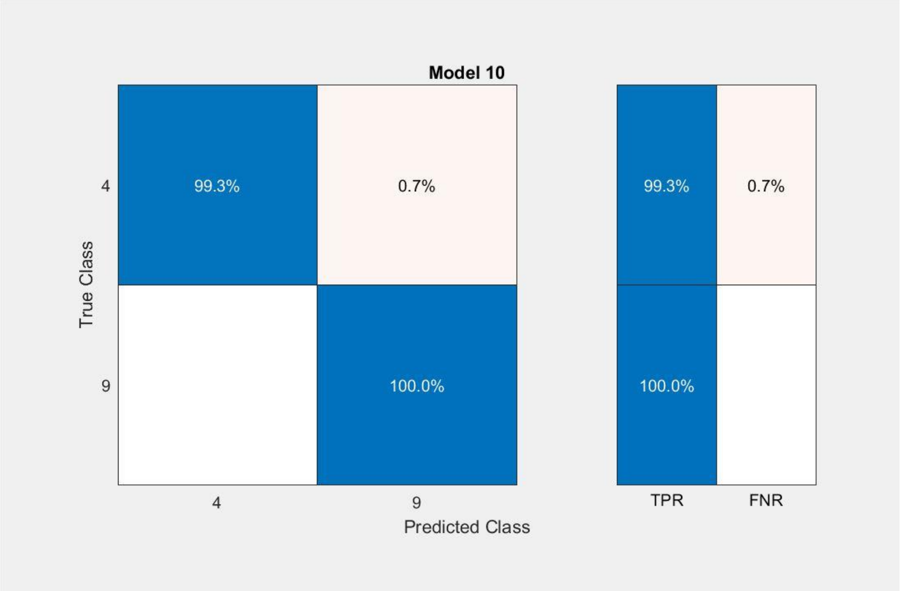

Figure C.xvix: Confusion Matrix of ten features selected by Hybrid feature selection algorithm

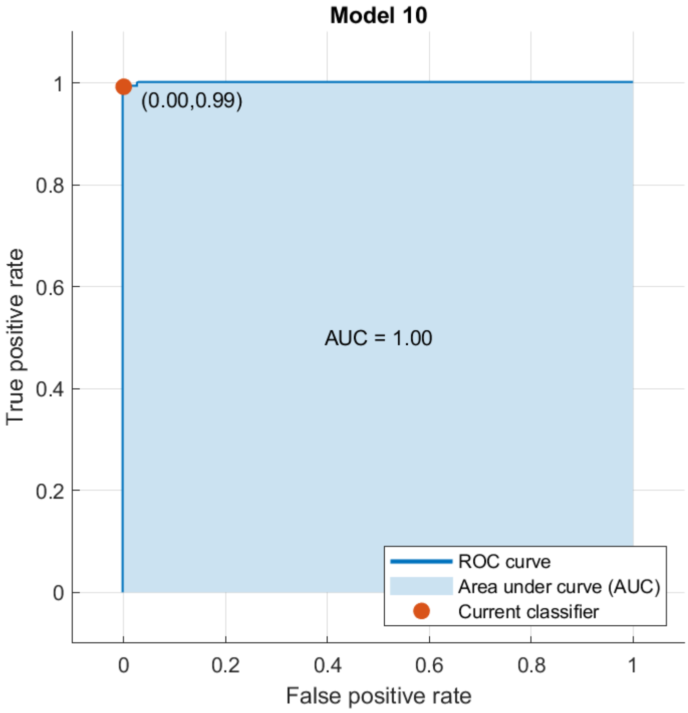

Figure C.xx: ROC Curve of ten features selected by Hybrid feature selection algorithm

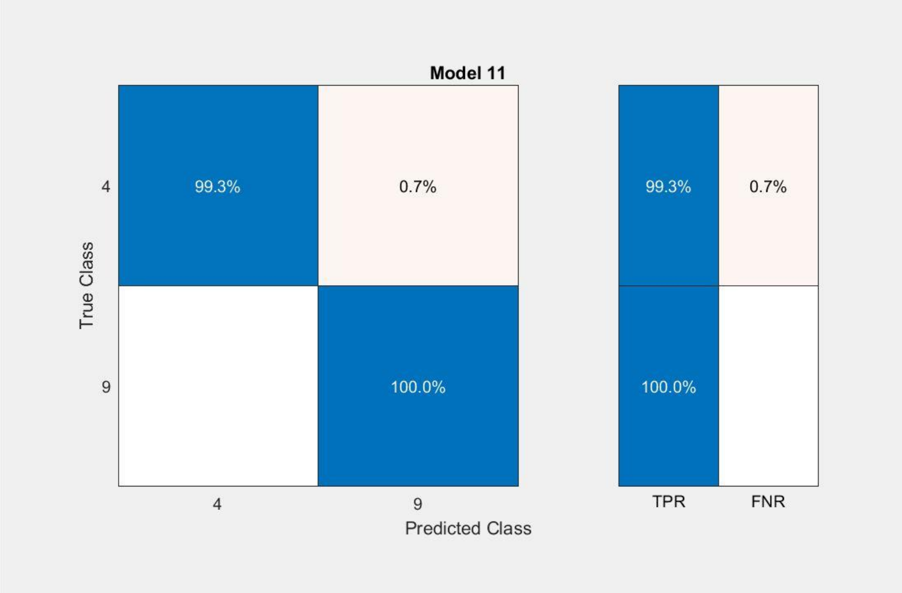

Figure C.xxi: Confusion Matrix of eleven features selected by Hybrid feature selection algorithm

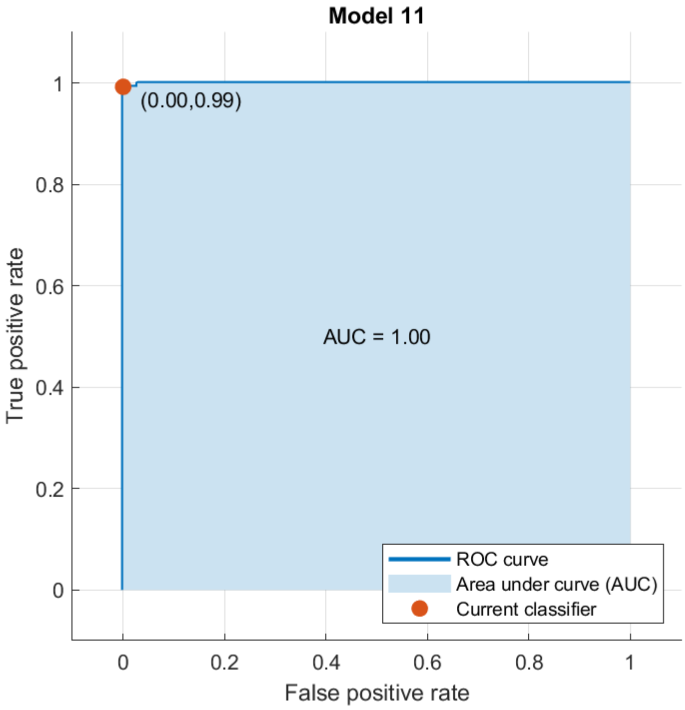

Figure C.xxii: ROC Curve of eleven features selected by Hybrid feature selection algorithm

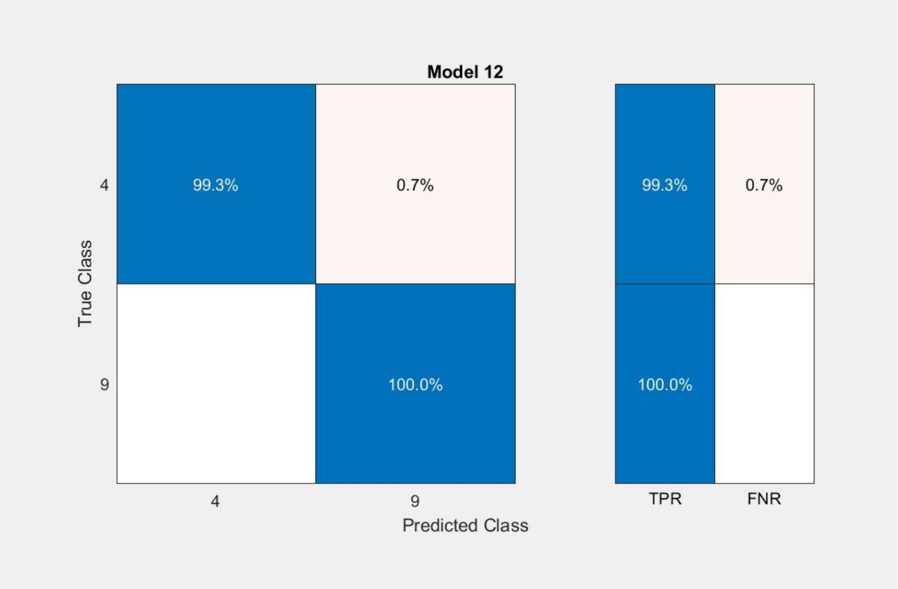

Figure C.xxiii: Confusion Matrix of twelve features selected by Hybrid feature selection algorithm

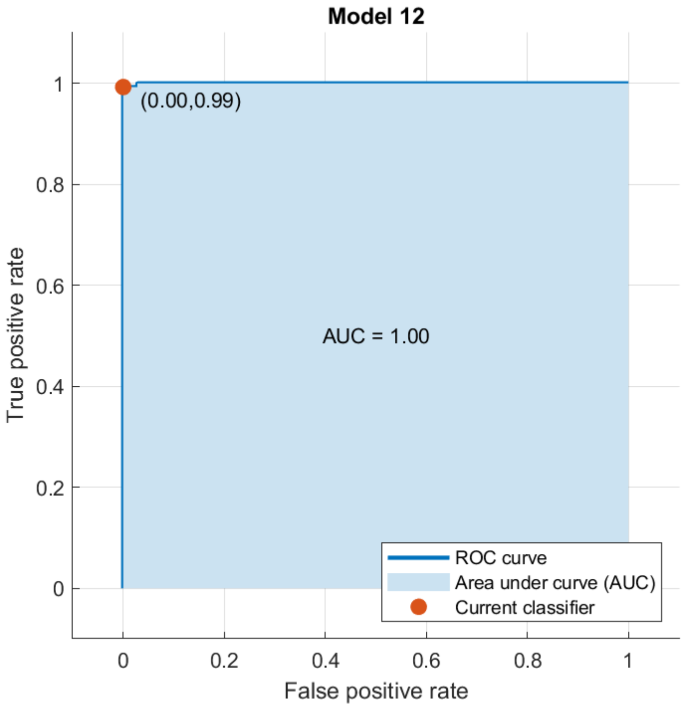

Figure C.xxiv: ROC Curve of twelve features selected by Hybrid feature selection algorithm

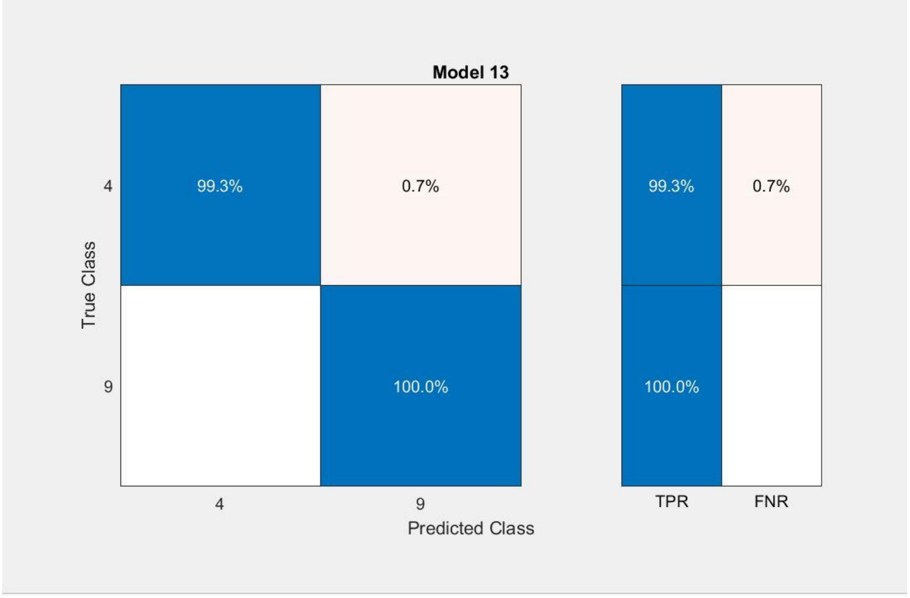

Figure C.xxv: Confusion Matrix of thirteen features selected by Hybrid feature selection algorithm

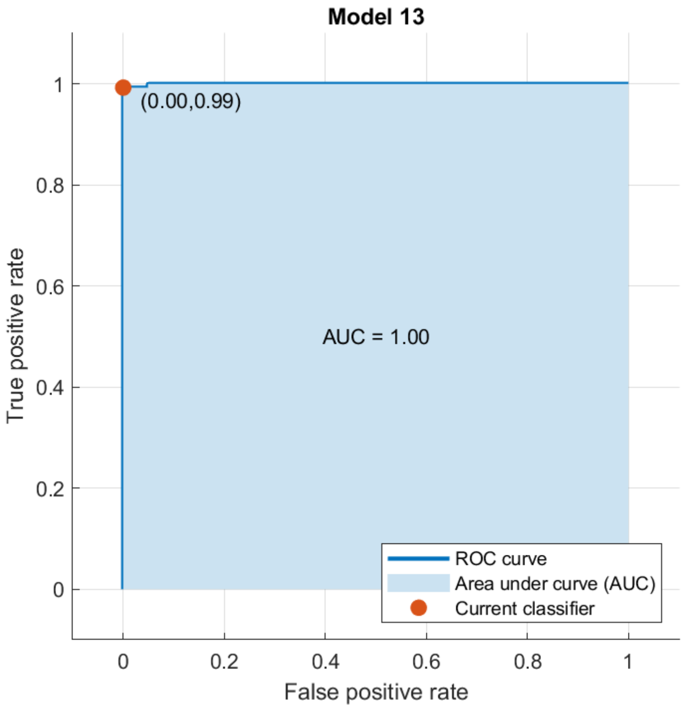

Figure C.xxvi: ROC Curve of thirteen features selected by Hybrid feature selection algorithm

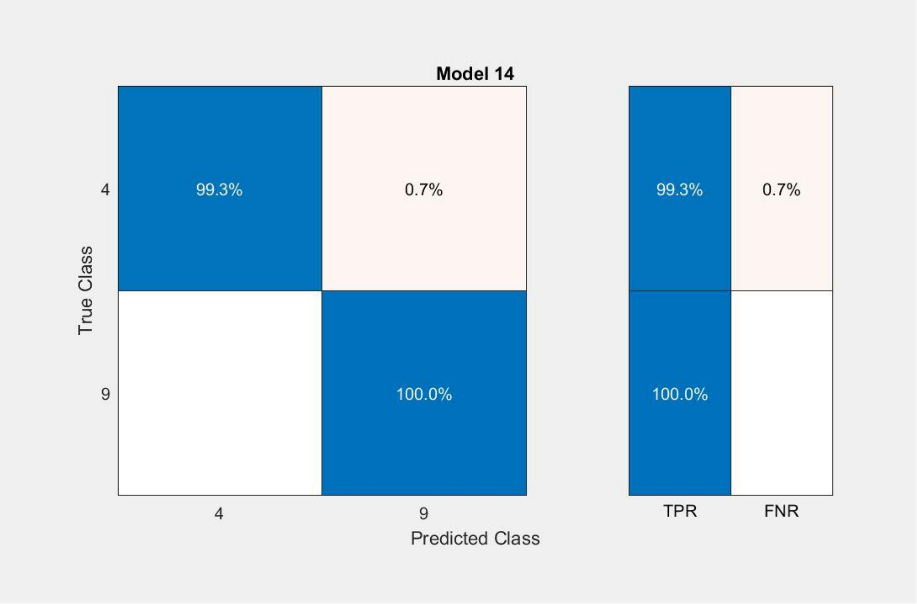

Figure C.xxvii: Confusion Matrix of fourteen features selected by Hybrid feature selection algorithm

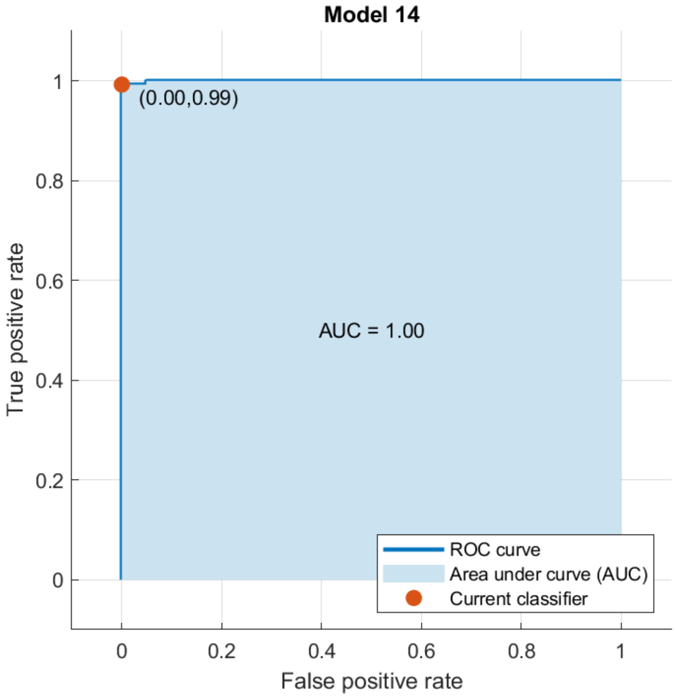

Figure C.xxviii: ROC Curve of fourteen features selected by Hybrid feature selection algorithm

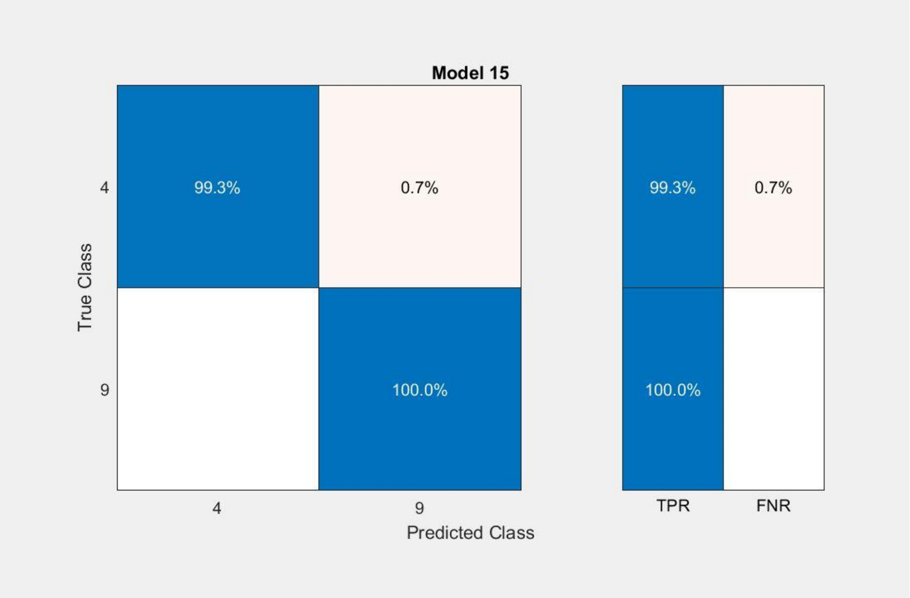

Figure C.xxvix: Confusion Matrix of fifteen features selected by Hybrid feature selection algorithm

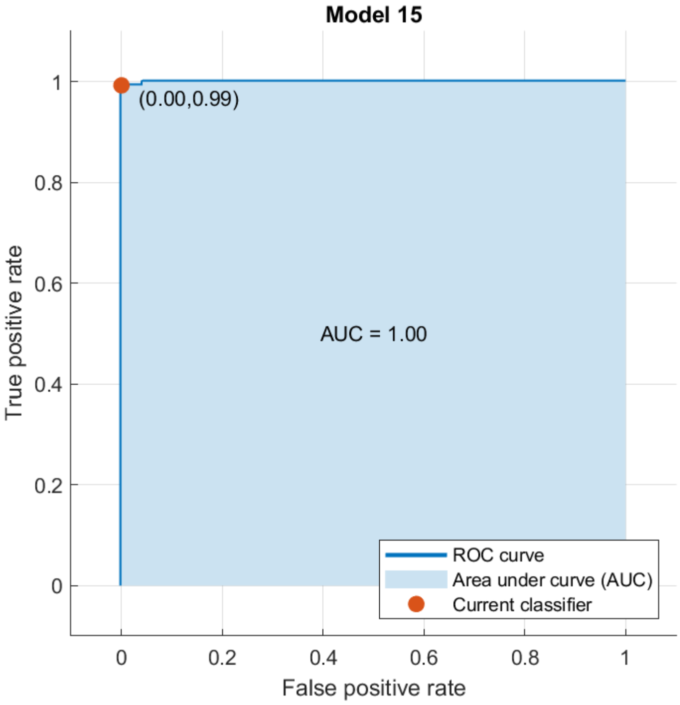

Figure C.xxx: ROC Curve of fifteen features selected by Hybrid feature selection algorithm