THOMPSON RIVERS UNIVERSITY

Credit Risk Modeling

A Comparative Analysis of Artificial and Deep Neural Networks

by

Marriappan Vasudevan

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF BUSINESS ADMINISTRATION

KAMLOOPS, BRITISH COLUMBIA

APRIL, 2020

Supervisor: Dr. Mohammad Mahbobi

Committee Members

Dr. Salman Kimiagiri

Dr. Li Zhang

Dr. Jabed Tomal

© Marriappan Vasudevan, 2020

Abstract

Credit risk assessment plays a major role in the banks and financial institutions to prevent counterparty risk failure. One of the primary capabilities of a robust risk management system must be detecting the risks earlier, though many of the bank systems today lack this key capability which leads to further losses (MGI, 2017). In searching for an improved methodology to detect such credit risk and increasing the lacking capabilities earlier, a comparative analysis between Deep Neural Network (DNN) and machine learning techniques such as Support Vector Machines (SVM), K-Nearest Neighbours (KNN) and Artificial Neural Network (ANN) were conducted. The Deep Neural Network used in this study consists of six layers of neurons. Further, sampling techniques such as SMOTE, SVM-SMOTE, RUS, and All-KNN to make the imbalanced dataset a balanced one were also applied. Using supervised learning techniques, the proposed DNN model was able to achieve an accuracy of 82.18% with a ROC score of 0.706 using the RUS sampling technique. The All-KNN sampling technique was capable of achieving the maximum true positives in two different models. Using the proposed approach, banks and credit check institutions can help prevent major losses occurring due to counterparty risk failure.

Keywords: Credit Risk, Deep Neural Network, Artificial Neural Network, Support Vector Machines, Sampling techniques

Acknowledgement

I would like to thank Dr. Mohammad Mahbobi for providing me guidance during my research and supervising my thesis. I would also like to thank my thesis committee members Dr. Salman Kimiagari, Dr. Li Zhang from School of Business and Economics, and Dr. Jabed Tomal from Faculty of Science, and Dr. Maryam Darvish from Department of Operations and Decision Systems, University of Laval for providing their thoughtful insights, reviewing my work and providing thoughtful comments on my thesis.

I would like to take this opportunity to thank Thompson Rivers University and School of Business and Economics for providing me with this opportunity to pursue my Master of Business Administration with the thesis. I would like to thank Heidi Milovick, Catherine Dallaire, Monica Macaulay, and Shelley Lee for their continued support and guidance throughout my MBA journey at TRU.

I would also like to thank all the professors at TRU with whom I have taken classes and who have been instrumental in helping me complete this program.

I'm humbled and blessed with so many friends that I have made during the time I have spent in TRU and Kamloops who have always been on cheer and supported for me at every stage of my career and this journey

Lastly, I would like to thank my parents for their love and support throughout my program journey.

Table of contents

List of tables

List of figures

List of Abbreviations

ANN.....Artificial Neural Network

AI.....Artificial Intelligence

DL.....Deep Learning

DNN.....Deep Neural Network

G-Mean.....Geometric Mean

KNN.....K-Nearest Neighbour

ML.....Machine Learning

RUS.....Random Under Sampling

ROC.....Receiver Operating Characteristics

ReLU.....Rectified Linear Unit

SVM.....Support Vector Machines

SMOTE.....Synthetic Minority Oversampling Technique

1.0 Introduction

1.1 Background

Credit risk is known as the probability of an organization or a consumer of financial credit instruments defaulting on the debt payment obligation, i.e. counterparty failure risk (Basel I, p.8). There are numerous standardized ways identified by the Basel Committee and Bank of International Settlements through which the member central banks and regional banks across the world can mitigate this risk. These techniques include collateralized transactions (Basel II, p.40), On Balance Sheet Netting (Basel II, p.42), Guarantees and Credit Derivatives (Basel II, p.42), Maturity Mismatch (Basel II, p.42) and other approaches like collateral against the debt obligations. Basel Accord II recommends forming credit risk control units (Basel II p.102), a team internal to the banking operations which can help in maintaining the ratings of the consumer and thereby maintaining oversight on the overall exposure of the bank to credit risk. These teams are likely to produce the internal ratings for a given credit approval request thereby which the banking officials can decisively take actions for the approval of debt or any kind of financial credit instruments. Although banks do implement these techniques in their credit risk management procedures, but by predicting these risks during the application process or prior to the customer request, banks can avert any sort of counterparty failure.

The financial credit instrument that we have used in this study are credit cards which have become a common form of payment in the last decade for a range of financial transactions. As per the report published by the Payments Canada (2019) on Canadian Payment Methods and Trends, of the total payment transactions that took place in 2018, 28% of the transactions were carried out by credit cards, an increase of 52% from 2017. Data released by the Canadian Bankers Association on credit card statistics (2018) indicated that the total net dollar value of transactions carried out by VISA and MasterCard holders exceeded CAD $547 billion in 2018. There were 75.8 Million cards in circulation for the year of which 0.8% of the card holder's were delinquent in credit card payment resulting in more than 600,000 credit card delinquency cases in 2018 alone (CBA, 2018). As the per the Global Payment reports (2019) published by JP Morgan Chase on the United States, US has a credit card penetration of 2.01 per capita which are enabled for e-commerce transactions. US Federal Reserve Bank's Economics

Research published the delinquency rate at 2.59% for the Q1 2019 which has been steadily increasing for the past two years from 2.42% in Q1 2017.

Given the growing trend in payments through credit cards, it can be assumed that the delinquency rate may increase over the coming years in terms of credit card payments. The major reason for the increase in the delinquency rate as per St Louis Federal Reserve (2019) has been due to increased user base of credit cards especially between age group of 18 to 29 years. The delinquency rate among these users in 2019 alone has been 8.05% as per St Louis Federal Reserve. In order to understand the delinquency, we must take a look at the definition of default used by banks across the globe. As per the Basel accord II, the definition of default is as follows (Basel II, p.104,105):

"A default is considered to have occurred with regard to a particular obligor when either or both of the two following events have taken place.

The bank considers that the obligor is unlikely to pay its credit obligations to the banking group in full, without recourse by the bank to actions such as realizing security (if held).

The obligor is past due more than 90 days on any material credit obligation to the banking group."

Following the definition of default, the delinquency rate for credit card payment obligations is calculated as defaulters who fail to pay the obligations for more than 90 days. Due to the limitations in the dataset the complete definition of delinquency may not be implemented in this study. However, for conducting this study since the credit instruments used has been credit card, default is considered when the clients fail to make any payment in the next month by due date. By predicting and identifying credit card customers who might be defaulting in the payments, banks can avoid major losses occurring due to the credit card defaulters. As per the Canadian Bankers Association Data on Credit Card Delinquency, the net annualized loss rate for 2019 alone has been 3.45%.

1.2 Rationale for the Study

According to McKinsey Global Institute (2016) implementing adequate measures with advanced analytics to detect credit risk and averting further losses, portfolios can reduce up to

50% of the cost in the credit risk operations of the business. One of the primary capabilities of a robust risk management system must be detecting the risks earlier, though many of the bank systems today lack this key capability which leads to further losses (MGI,2017). By implementing and placing a system to check defaulters, banks can avoid losses which will help save the bank millions of dollars. With reference to our study, these losses would be occurring due to credit card default payments. This leads us to the rationale behind the study of developing a model using deep neural network (DNN) architecture which can efficiently help the banks in identifying the defaulters and thereby helping them save millions of dollars. Identifying and classifying credit card defaulters using machine learning and advanced analytics can help banks and financial institutions detect their risk early in the transactions or in a client's portfolio based on the data available in the system. This will allow banks and financial institutions to implement appropriate measures and help them in targeted risk-based pricing, faster client service without sacrifice in risk levels, and more effective management of existing portfolios (Bahillo et.al, 2016).

Our major objective is to develop a robust and efficient DNN model with a combination of specific sampling algorithms based on machine learning techniques. This thesis would then conduct a comparative study with the already established machine learning techniques like Support Vector Machines (SVM), K-Nearest Neighbours (KNN) and Artificial Neural Network (ANN) used in credit risk assessment and the respective literature. These models have been developed from the understanding of the current literature and techniques already in place for credit risk identification and classification. To undertake and complete this research we plan to use datasets that include open-source data sets offered by the University of California, Irvine database (https://archive.ics.uci.edu/ml/datasets, 2019) available for conducting researches and developing such models.

Our inspiration for research is based on the recent advancements in the use of artificial intelligence and machine learning techniques to solve the problems faced by the financial industry. The probability of default and classification of the defaulters in credit risk assessment has been widely studied with machine learning techniques but limited with regards to deep learning techniques. In this thesis, we propose a 6 Layer-DNN Model to study credit risk assessment. We will be comparing it with techniques like ANN, SVM and KNNs which are

some of the widely used models to and predict study credit risk assessment. This thesis will also include the study of sampling techniques like SMOTE, RUS, SVM-SMOTE, and All-KNN to be used along with the imbalanced dataset and the models.

1.3 Organization of Thesis

The organization of the thesis is as follows. Chapter 2 describes the current classification techniques used in credit risk research and models used in this study. Established machine learning models used for the comparative study are explained in detail. DNN architectures along with our model proposed for this study is introduced in this chapter. Chapter 3 presents the literature on credit risk assessment along with specific techniques or models used in those studies. Chapter 4 outlines the methodology used in this study and the process carried out while conducting the study. The performance evaluation, robustness, and sensitivity analysis carried out for the models are discussed in detail in this chapter. Chapter 5 presents a comparative study between the performances of the different models using performance metrics, confusion matrix, and ROC curve. Chapter 6 concludes the thesis by presenting key results of the evaluations, further discussion into the policy implications of using the models in real-world application and future work in incorporating a combination of techniques for credit risk classification.

2.0 Classification Techniques and Approaches

Post-Great Recession (2008-2009), credit risk identification and prevention have received great importance from managers of the financial institution for issuing debt and line of credit (Harris 2013). Regulatory developments post-global financial crisis has mandated to perform complete due diligence on the credit history of the companies and candidates requesting for the credit line. These regulations have initiated the development of a variety of techniques under the credit risk scoring model (e.g. Basel III). Financial firm and investment banks heavily rely on these scoring techniques to identify defaulters so that credit lines are offered to the most legit ones. One of the earliest risks scoring statistical techniques discriminant analysis (DA) was developed based on the Fisher's linear discriminant model (1936) and his seminal paper published on the topic of quantitative techniques to classify between "good" and "bad" applicants.

In the past few decades, data mining techniques based on supervised learning and unsupervised learning algorithms have been implemented for classification and default identification. Data mining is a process of analyzing data using different techniques and by different dimensions which can then be used in the process of decision making to cut costs, to identify risk, to improve customer service and to involve many more applications. It ideally involves finding the relationship between the dependent variable and set of independent variables or features involved in a given dataset. In this chapter, we take a deeper look into the established techniques used in the study and introduce our DNN model.

2.1 Supervised and Unsupervised Learning

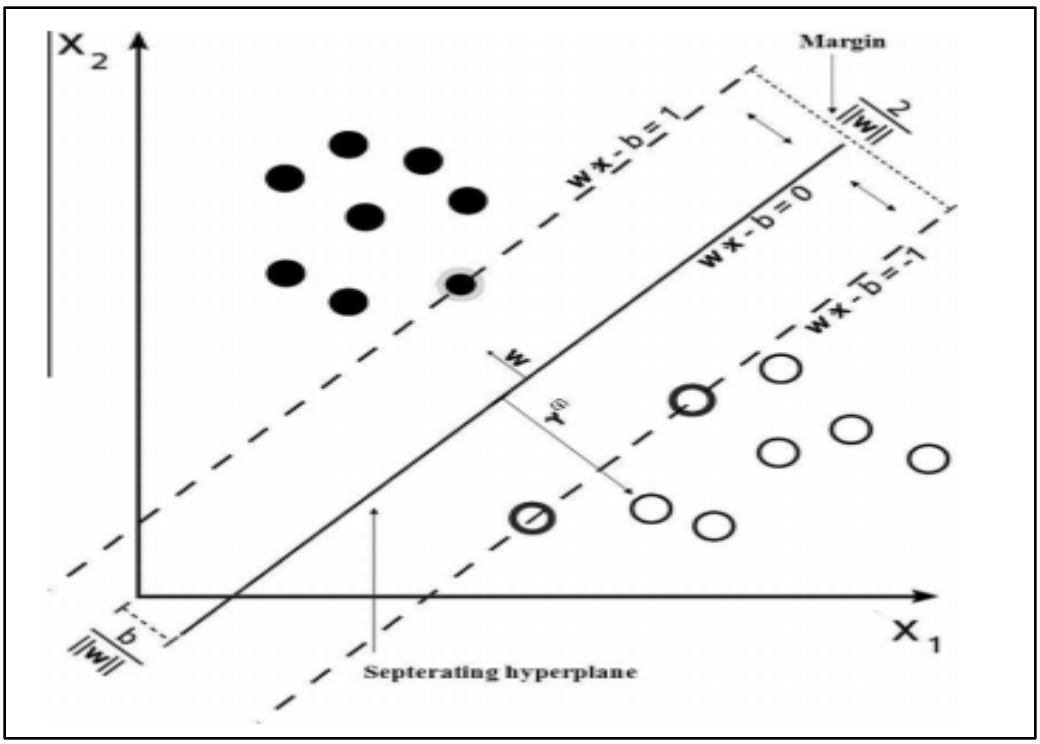
Data Mining techniques can be classified as supervised learning and unsupervised learning techniques. The primary difference between supervised learning and unsupervised learning is that in supervised learning the models are trained using a partial dataset ideally 80% of the dataset and post which these models are used for prediction and classification problems. In unsupervised learning however, the step to train the model is skipped and these models are directly used for solving the problems. Unsupervised learning techniques are much more complex as compared to the supervised learning techniques given the nature of the problem in hand.

Supervised learning techniques involves the process of modifying and optimizing the systems so that the desired outputs or targets are detected for a given range of inputs (Reed & Marks, 1999). It involves training the model which can also be termed as the process of adaptation through which the models can learn the relationship between the inputs and outputs. It involves an external entity termed as an external "teacher" (Reed & Marks, 1999) which helps in specifying the output for a given set of input variables. In some machine learning literature directed data mining is termed as supervised learning which involves classifications, prediction, and estimation (Hamori, 2014) whereas undirected data mining techniques are termed as unsupervised learning which involves affinity grouping and clustering (Schmidhuber, 2014). This thesis utilizes supervised learning techniques. These techniques include SVM with RBF Kernel, KNNs, ANN and DNNs.

2.2 Support Vector Machines with Sigmoid and RBF Kernel

Support Vector Machines (SVM) are one of the prominent binary classification machine learning models utilized to resolve the problem of classification especially if the dataset consists of binary features (T. Harris, 2013). Support Vector Machines. SVM were first developed by Vapnik & Cortes in 1995 which attempts to find the optimal separating hyperplane between the classes by maximizing the class margin (T. Harris, 2013). The model can be depicted as in Figure 1.0. The points lying on the boundaries of the hyperplane are called support vectors. The optimal hyperplane is found by maximizing the width of the margin. Figure 1.0 shows the margin as the distance between the separating hyperplane between the positive class and the negative class.

The optimization function in the SVMs for finding the optimal hyperplane is carried out by functions called kernel functions. These functions play a similar role in finding an optimized solution similar to an optimization problem. For this thesis, the Radial Basis Function (RBF) is used as a kernel function. RBF reflect SVMs with exponential functions whereas Sigmoid functions are taken as a function of the tangent to the input parameters.



Source: T. Harris, 2013

Figure 1.0 Illustration of SVM

Table 1.0 indicates the functional form of SVM involved in the study along with parameters and default values.

Table 1.0 Functions and Parameters of SVM used in this study

Kernel Functions	**Functional Form**	**Parameters**	**Default Values**
Radial Basis Function	K(xi.xj)=exp(-γ||xi-xj^2)	$\gamma \in R$	$\gamma = 1$

Source: Khemakhem & Boujelbène, 2015

SVM works on the optimization of the margin between the hyperplane. For a set of training instances say $\{(x_1, y_1), \ldots\ldots\ldots\ldots (x_n, y_n)\}$ where $x \in R^n$, $y \in \{-1,1\}$ where y is the class label for the dependent feature in a binary classification problem as in this study. In a binary classification problem, SVM attempts in finding a classifier $f(x)$ which in turn minimizes the misclassification rate. The $f(x)$ is the hyperplane which can be represented as $f(x) = sgn(w^T x + b)$. This function in training results in the convex quadratic optimization problem.

The convex optimization problem can be rewritten in a dual quadratic programming problem form using the Lagrangian functions as below.

$$minimize\ W(\alpha) = 1/2 \sum_{i=1}^{n} \sum_{j=1}^{n} yiyj\alpha i\alpha jK(x_i, x_j) \quad (1)$$

$$subject\ \ to$$

$$\sum_{i=1}^{n} yi\alpha i = 0\ \forall i : 0 \leq \alpha i \leq C$$

Here α is the Lagrange multipliers and C is the tradeoff between the maximum margin and misclassification error. The term $K(x_i, x_j)$ represents the kernel functions which are used to map linearly non-separable instances into a higher dimensional space. The kernel used in the study is represented in Table 1.0.

2.3 K- Nearest Neighbours

Nearest Neighbour algorithm has been one of the majorly studied algorithms with respect to classification problem. The algorithm was first introduced by Fix & Hodge in 1951 with their seminal paper on 'Discriminatory analysis, nonparametric discrimination'. The researchers were the first ones to establish the rules of the Nearest Neighbour and how the algorithm identifies the nearest neighbors using Euclidean distance. Cover & Hart introduced the nearest neighbor algorithm for pattern classification in 1967 and identified how the K-NN algorithm can fit into a broader applications of classification problems.

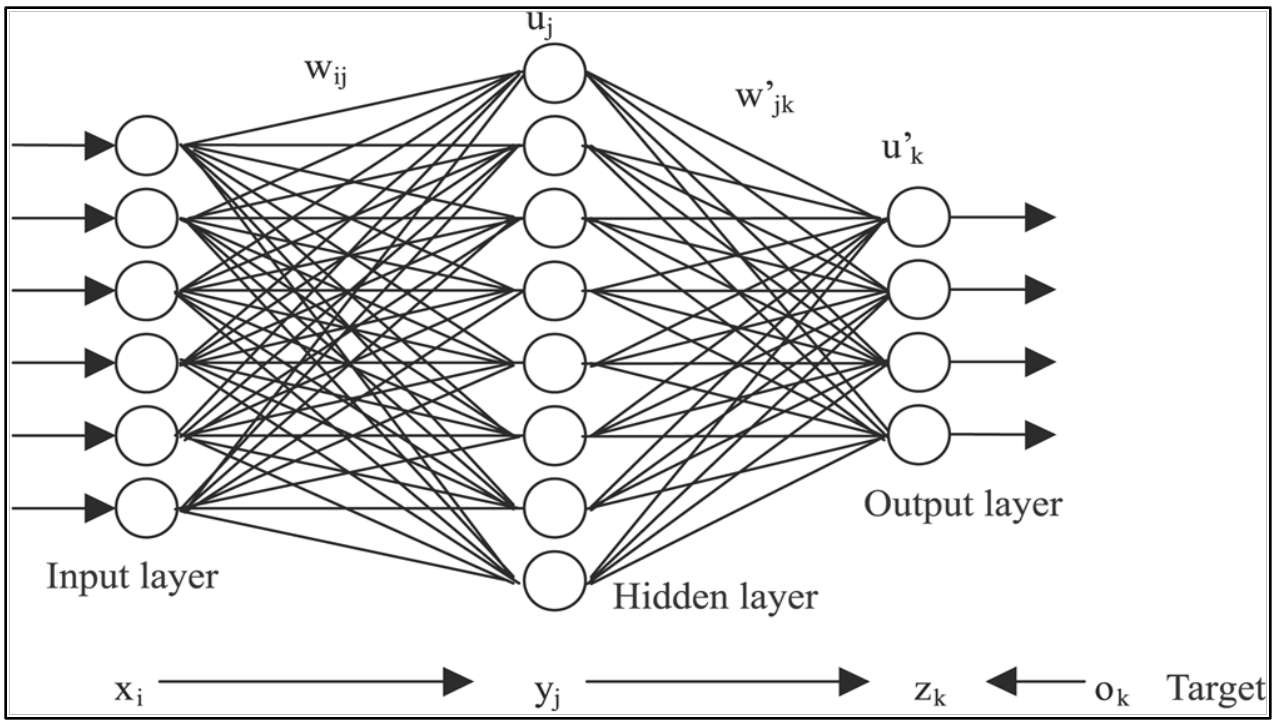
KNN was introduced by Altman N. S in 1992 as a nonparametric method for pattern recognition and classification. This algorithm also belongs to the class of supervised learning techniques as the algorithm requires to be trained before the actual application of the algorithm on a give set of independent features. It is also one of the standard machine learning methods which can be extended and applied for large scale data mining problems (Nadkarni, 2016). The algorithm uses the common principle that in a given dataset, similar objects or features exist within the proximity of one another.

Being a non-parametric classification technique, KNNs can be used for non-linear datasets like credit risk assessment. In this thesis, the K-NN algorithm is used as a classification technique to identify the default payments in the dataset. Parameter tuning is key relative to the K-NN model. One of the most important parameters to be identified for K-NN is the

number of nearest neighbors. Using the trial and error method, we have tuned our nearest neighbor to be 10 based on the understanding of overfitting and underfitting the model. Overfitting the model means using excessive data points to fit the data onto the model which results in plain memorization of the datapoints by the model (Massaron & Boschetti, 2016, p.94) and thereby can provide incorrect measurements for the model prediction. Underfitting on the other hand indicates use of less datapoints or information to fit the model thereby not utilizing the complete information for training the model accurately.

2.4 Artificial Neural Network (ANN)

ANNs consist of neurons that are similar to those of human neurons. These neurons form a single functional unit in the layer of networks. The ANN can consist of one to many layers making them easily programmable algorithms to be studied in the field of computer science. The mathematical model of a neuron was proposed by McCulloch & Pitts in 1943. The neuron proposed by McCulloch & Pitts in 1943 consisted of binary input, binary output, and single activation function. Stacking multiple neurons with a given set of input variables and connecting them with different weights and activation functions provides us with ANNs or simply neural networks. The most common form of the neural network is known as the feed-forward network where the information from the input variables is carried forward linearly through cross-connected neurons as the middle layers and finally towards the desired output layer. These networks are termed as “feed-forward” as the information flow in only one direction without any feedback loops or back into the hidden layers



Source: Retrieved from https://www.extremetech.com

Figure 2.0 Illustration of feed-forward neural network

Over the past few years with the help of advanced programming languages, neural network research has led to several other architectures like error back-propagation neural networks, recurrent neural networks and convolutional neural networks which is a widely implemented neural network in the image processing and image recognition technologies. The ANN in this study has been influenced by the work of Khemakhem & Boujelbene, 2017 where they used an ANN to conduct a credit risk assessment. The ANN used in this thesis consists of 4 layers which are as follows:

Layer 1: Input Layer consisting of 10 neurons representing the 10 input variables

Layer 2: A hidden layer consisting of 16 neurons

Layer 3: A hidden layer consisting of 10 neurons

Layer 4: An output layer consisting of a single neuron.

This thesis uses Rectified Linear Unit (ReLU) as the activation function for the neurons with a feed-forward neural architecture as explained above. The hidden layer neurons were

optimized throughout this study for better accuracy and classification results through the trial and error method. The choice of neurons in the hidden layer were decided by a common assumption to form a tunnel architecture in the network topology of the neural networks as to reduce the error rates in the neural networks. Combined with this assumption and using multiple trials for avoiding overfitting of the models the neurons were appropriated at 16 and 10 for the hidden layers in the ANN architecture. Similar method was carried out for finalizing the architecture of the DNN model. We have used the binary_crossentropy as our loss function and Stochastic Gradient Descent as our optimizer for the neural network model.

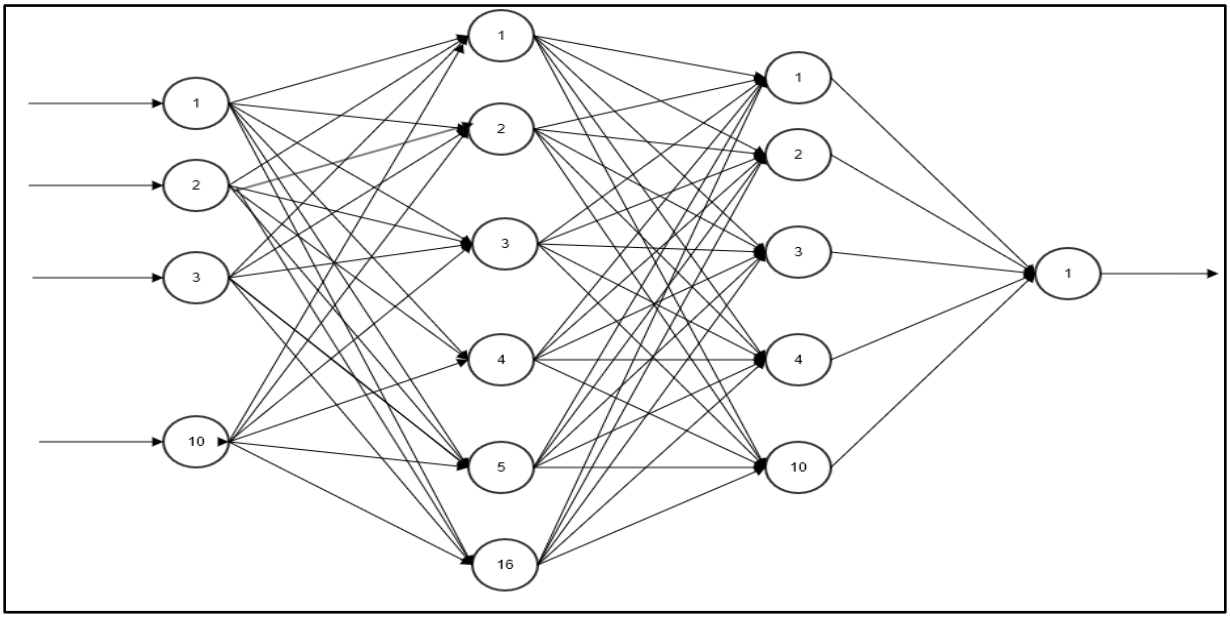
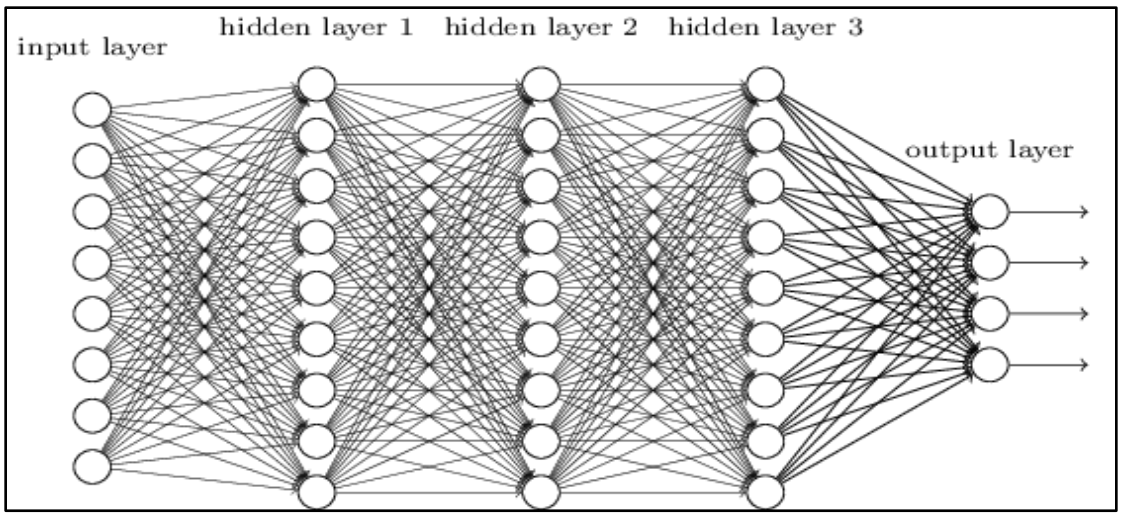


Figure 3.0 Architecture of the feed-forward network used in the study

2.5 Deep Learning Architectures

DNNs consists of multiple layers of neural networks and works on a similar line of ANN. They form a part of the larger family of deep learning architectures which also consists of Deep Recurrent Neural Network, Deep Belief Network, and Deep Convolutional Neural Networks. Figure 4.0 presents an idea of a DNN with 3 hidden layers. DNN architectures for broader applications can include N-different hidden layers depending upon the optimization of the model and problem being solved using DNN.



Source: Retrieved from http://neuralnetworksanddeeplearning.com/chap5.html

Figure 4.0 Illustration of DNN with Feed-Forward propagation

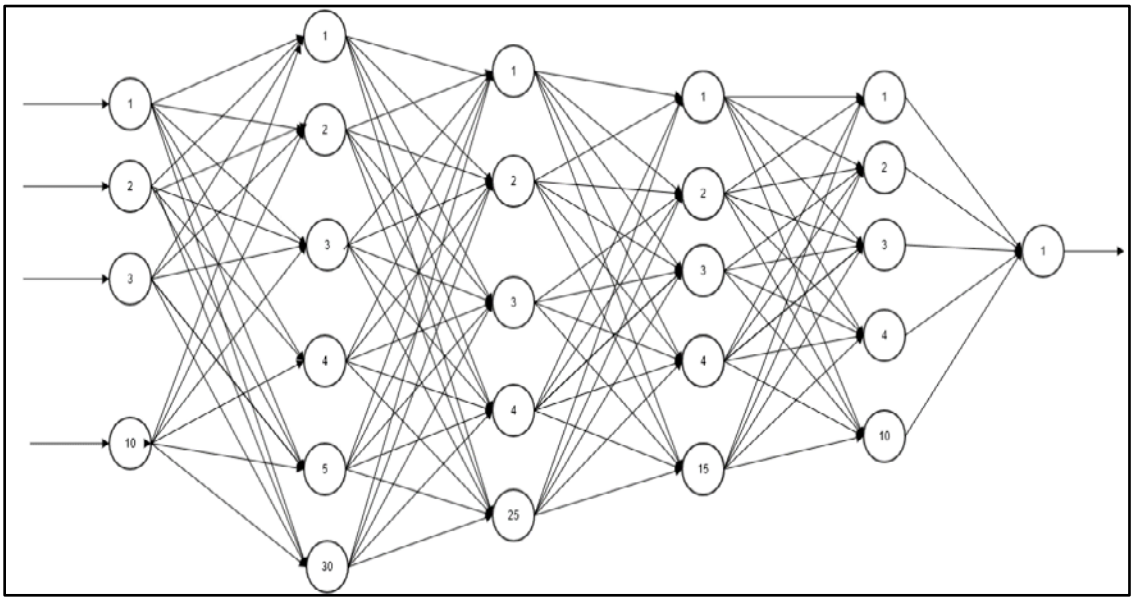


Figure 5.0 Architecture of DNN used in this study

The DNN used in this thesis consists of 6 layers which are as follows:

Layer 1: Input Layer consisting of 10 neurons representing the 10 input variables

Layer 2: A hidden layer consisting of 30 neurons

Layer 3: A hidden layer consisting of 25 neurons

Layer 4: A hidden layer consisting of 15 neurons

Layer 5: A hidden layer consisting of 10 neurons

Layer 6: An output layer consisting of a single neuron.

This thesis uses Rectified Linear Unit (ReLU) as the activation function for the neurons with a feed-forward neural architecture as explained above. The hidden layer neurons were optimized throughout this study for better accuracy and classification results through the trial and error method. To reduce the loss function, we have used the binary_crossentropy and we have used Stochastic Gradient Descent as our optimizer for the DNN model.

3.0 Literature Review

In this chapter, a detailed literature review of the studies in the field of credit risk assessment is presented. Section 3.1 outlines the studies conducted with SVM and comparison with other methods. Section 3.2 discusses in detail the studies conducted with ANN. The following section 3.3 discusses the latest research in the credit card default detection techniques and outlines literature on DNNs.

One of the earliest risks scoring statistical techniques, discriminant analysis (DA) was developed based on the Fisher's linear discriminant model (1936) and his seminal paper published on the topic of quantitative techniques to classify between "good" and "bad" applicants. Post-1980, the DA techniques were replaced by statistical techniques like linear regression, logistic regression and early stage base classifier likes nearest neighbours, decision trees that provided significant results provided the data were linearly separable, however, if the data sets are not linearly separable then these techniques have proved to be insufficient for credit risk analysis (S. Chen et al, 2011). In the past decade, researchers and analyst have shifted their focus on ANNs and machine learning techniques to classify the defaulters from non-defaulters where the datasets are not linearly separable. Some of the non-linear numerical methods for classification included ANN, SVM and maximum likelihood model proposed by Standard & Poor's Risk Solutions Group (S. Chen et al, 2011). Khemakhem & Boujelbène (2015) studied the difference between DA and ANN on Tunisian companies and established the fact that neural network (NN) models are more accurate in terms of predictability. They criticized NN models for being less robust and less well-founded terming them a black box of unknown operating rules as NN models are unable to explain the results provided by them.

3.1 Credit Risk Assessment with SVM

In the past few years, kernel-based vector algorithms derived from the statistical learning theory by Vapnik (1998) have come into a wide variety of research for classification problems and one of them is SVM. SVM are one of the latest machine learning techniques used in the finance industry to classify defaulters and non-defaulters based on their credit and financial history. SVM falls under the category of supervised machine learning techniques

which can be used for classification or regression problems but often these techniques are used for classification problems.

L. Yu et al (2010) studied credit risk evaluation using SVM with a multiagent ensemble learning system They used credit card applicants from British financial service companies and increased the bad applicants to match the level of good applicants. This allowed them to perform their study on the balanced dataset. As per L. Yu et al (2010) Multiagent system with SVM outperformed Logistic Regression, Quadratic Discriminant Analysis, and Feed-forward neural network but lagged with Multiagent Feedforward Neural Network model. Their study did not include any kind of sensitivity analysis or robustness test with the model which would have helped in understanding the application of the models. Their study also lacked in explaining the implications of using such models on credit risk evaluation and future applications.

S. Chen et al (2011) studied the bankruptcy of German firms using SVM with a Gaussian Kernel. They used 28 different financial ratios for the firms that went bankrupt between 1996 to 2002 and used these ratios as features for the algorithm. S. Chen et al (2011) identified that SVM outperforms logit in terms of classification problems especially in the case of linearly non-separable datasets. Their datasets consisted of 20,000 solvent firms and 1,000 solvent whose financial statements were extracted from the database Creditreform. S. Chen et al (2011) did perform sensitivity analysis using the parameters of the SVM but overlooked the imbalanced dataset they used for the study.

J.-H. Trustorff et al (2011) conducted a similar study using least squares SVM and logistic regression models. They chose 5 debt ratios to identify the credit risk of the companies and in total studied 78.000 companies using these ratios. One of the major outcomes of this study was that SVM perform well under small training samples with high variance in the input data (J.-H. Trustorff et al, 2011). Both J.-H. Trustorff et al (2011) and S. Chen et al (2011) have overlooked the imbalanced dataset they used in the study. To overcome this problem in our study we have used over-sampling and under-sampling techniques which will be explained in detail in the next chapter.

Wang & Ma (2012) used a hybrid ensemble approach for detecting enterprise credit risk assessment. They used financial records of 239 companies provided by the Industrial and

Commercial Bank of China. The method involved Bagging and Boosting techniques along with Linear and Polynomial SVM kernel. However, the dataset used in this study was much smaller in comparison to other datasets used in the study. Lack of applications of the methodologies to a large dataset was one of the shortcomings of this research.

Harris studied credit risk assessment in 2013 and in 2015 which is of particular interest to us. These two studies involve the use of SVM in credit risk assessment. T. Harris (2013) conducted a study on credit risk assessment based on default definitions as given by the Bank of International Settlements and Base Committee. His study argued that using "narrow" and "Broad" definitions of defaults based on the number of days past due payments, credit risk evaluations could be improved using quantitative credit risk models. His methodologies, however, lacked in providing clear applications of the credit risk models along with any sensitivity analysis of the models. His study in 2015 involved the application of clustered SVM proposed by Gu and Han (2013) and compared it with techniques like logistic regression, decision trees and a combination of other techniques. In this study, he used German Credit Dataset provided by UCI Machine learning repository and Barbados credit union dataset.

Cao et.al (2013) proposed a novel model-based of cost-sensitive SVM (CS-SVM) enhanced by particle swarm optimization technique (PSO) for loan default discrimination. Their research improved the SVM model integrating with cost sensitivity and PSO increasing the accuracy of the output but their model was applied as a binary classification technique to a specific bank data thereby limiting the application of the model for a wider dataset. Limitation of the model application on the wider dataset places the question of efficiency and scalability on the model used by Cao et al (2013) and suggested for further research on multi-class multi-feature classification clustering models for shortcomings in their research.

Paulius Danenas & Gintautas Garsva have studied the application of SVM in credit risk assessment in different scenarios and using different combinations of kernel functions. One of their recent research (Danenas & Garsva, 2015) on credit risk assessment was completed by SVM with particle swarm optimization as used by Cao et al (2013). They also utilized financial ratios as the input features for the credit risk assessment. In their research, they used the Zmijewksi score (Z-score) as a binary output feature with companies scoring greater than zero i.e. $Z > 0$ to be labeled as bankrupt. They compared the measurements of the model with

logistic regression and RBF based network classifiers. Limitations on the stability of particle swarm optimization-based SVM were one of the major lacking points of their research. The model didn't outperform linear SVM models as used by other researchers in the credit risk assessment.

Based on the literature presented above, SVM has been one of the prominently studied models in credit risk assessment. This makes it one of the ideal models to be involved in the study and conduct comparative research with the DNN Model presented in this study.

3.2 KNN in Credit Risk Assessment

Henley & Hand (1996) studied K-nearest neighbor as a classifier for credit risk scoring techniques by considering the bad risk rate as part of their research. The authors identified that K-NN performed well in identifying the bad risk rate and was able to perform well in comparison to decision trees, logistic and linear regression. The dataset used by Henley & Hand (1996) was fairly balanced with over 54% of the dataset consisting of credit risk and involved 16 features. The researchers were able to reduce the bad risk rate up to 40%. Although the research was carried in the early developmental stages of machine learning techniques, the researchers didn't give a detailed performance metrics of the models studied and further application of the model in the credit risk assessment. Post their study as per our knowledge based on the research for literature review, K-NN's application was not studied until the early 2000s.

Marinakis et. al (2008) studied the nearest neighbor classifier using metaheuristic algorithms for credit risk assessment using loan portfolios of 1411 firms from Greek Commercial Bank. The authors used 16 different financial ratios including profitability, solvency and managerial performance ratios. The dataset used had 218 firms with default class whereas 1193 firms were non-default class (Marinakis et. al, 2008) making it an imbalanced dataset but their research didn't involve any techniques to make the imbalanced dataset a balanced one. Using the metaheuristics algorithms some of the models were able to achieve more than 98% accuracy with an overall average of between 94% to 97% percent.

Abdelmoula (2015) studied the Tunisian bank credit risk using the K-NN algorithm with 3 nearest neighbor parameters. The dataset consisted of 924 credit records between 2003 to 2006 held by a Tunisian commercial bank (Abdelmoula, 2015). Abdelmoula (2015) was able to

obtain an accuracy of 88.63% with over 95% in terms of ROC score. The author used over 24 financial and non-financial ratios as features of the study with cash flow and non-cash flow models. Abdelmoula (2015) also used Type 1 and Type 2 error rates as credit risk and commercial risk to identify whether the models are able to cover these error rates which would help the banks in making efficient risk management decisions. Type 1 error rate indicates the rate of default customers being categorized as non-default customers and Type 2 error indicates the rate of non default customers being categorized as default customers(Abdelmoula, 2015).With respect to methodology, although the author used ROC as the main performance metric, there was no discussion regarding the dataset's imbalanced nature. It would have been highly possible that the dataset involved may have been imbalanced and thereby the research lacked any techniques to improve the dataset. Being said that to the best of our knowledge while conducting this research Abdelmoula's (2015) research is one of the high-quality researches in the use of K-NN with respect to credit risk assessment.

Although K-NN is one of the base classifiers and highly popular machine learning techniques, there hasn't been much application of different types of K-NN in the credit risk assessment. This knowledge discovery comes as a collateral finding as a part of this research.

3.3 Artificial Neural Networks in Credit Risk Assessment

Khashman (2010) built a credit risk evaluation system using three different neural network models using 24 numerical attributes and implemented it with nine different learning schemes. From 27 different learning models, he chooses 3 learning models which provided an error rate of less than 0.008 which does indicate that efficient models require iterative regression procedures to deliver accurate risk evaluation techniques. These three models delivered an overall accuracy rate of 83.6% but the research lacked in multiple points like feature selection procedures as in how the clients were chosen for the training and validation procedures. All three models used only one hidden layer in their design whereas the latest research focuses more on multiple hidden layers to enhance the results and achieve better accuracy.

Cimpoeru (2011) introduced the concepts of neural calculus and studied the concept of error backpropagation techniques. The author of this research focused on multiple models like feedforward networks with multiple layers, adaptive networks based on fuzzy algorithms and SVM's. Cimpoeru (2011) conducted a study on Romanian small-medium enterprises whose

turnover was between EUR 700,000 and EUR 3,755,000. The research was conducted on 2% of the total population as sample and input variables were financial ratios based on the data available. Although the research conducted was extensive but the research lacked clearly outlining the application of these models in real-time datasets and what can be done to improve the efficiency of the models.

Karaa et.al (2012) conducted a similar study by comparing SVM and NN models and established the superiority of NN models over SVM. The researchers focused mainly on the historical datasets of the companies and their financial ratios. The authors didn't mention if the dataset was imbalanced and any use of sampling techniques in the research. They achieved accuracy of 90.2% accuracy with NN model and Type 1 error rate at 18.55%. They also indicated their comparative results between DA and logistic regression techniques which proved that logistic regression is a better model in resolving classification problems.

Oreski et.al (2012) investigated the extent of the impact that total data from a single bank has on the genetic algorithms based neural network (GA-NN) for credit risk assessment. Their primary study was based on the subject of feature engineering and feature selection through hybrid models of genetic algorithms which helps in better feature selection for data processing and evaluation as compared to other models. Using the same hybrid models in both places i.e. in feature extraction and in the data-processing has allowed the researchers to get better accuracy as compared to using different models in different places. Although the research was carried out and performed with far better accuracy genetic algorithm-based neural network (GA-NN) are computationally intensive techniques as per the researchers and the feature selection process takes a longer duration of time to complete. Implementing this technique in the banks will definitely require optimization of the models and the internal parameters as well because each bank uses a different set of ratios to determine the credit risk assessment of the clients. Even though the accuracy rate of 82.30% was achieved using these models it can be improved using some of the advanced artificial intelligence techniques like SVM and DNN Models. Moreover, the limited application of this model due to technology-intensive requirement necessitates the study to be improved and provide better models for real-world applications.

Khemakhem & Boujelbène (2015) studied the difference between DA and ANNs on Tunisian companies and established the fact that neural network (NN) models are more accurate in terms of predictability but they criticized NN models being less robust and less well-founded terming them black-box operating rules as NN models are unable to explain the results provided by models used in the study of Tunisian companies. ANN although in many cases provided better results (Oreski et.al 2012, Khemakhem & Boujelbène 2015) as compared to linear models in classification, it has been criticized for being vulnerable to multiple minima problems as OLS and MLE were (S. Chen et al 2011). The major reason cited behind this vulnerability was due to the principle of minimizing empirical risks leading to the poor classification of sample data sets (Haykin 1998, S. Chen et al 2011). Several researchers in the past years have done comparative analysis between different models of ANN and ML techniques to understand the shortcomings and learning to improve the efficiency of such models. Khashman (2010), Cimpoeru (2011) and Karaa et.al (2012) conducted this kind of research by comparing different models to understand their impact on the data and the output.

3.4 Deep Learning Models in Credit Risk Assessment

With the advancements in machine learning, development of software languages and faster processing capabilities of computers, DNN and Deep Learning Architectures have taken center stage in the study of applications relative to predictions and classifications. Sun & Vasarhelyi (2018) studied the application of DNN on credit card delinquencies, one of the major influencers for conducting this study. The credit card applicants from one of the largest banks in Brazil with over 700,000 credit card applicants and found out that deep learning actually improves the accuracy of prediction in case of a large dataset. Although they used a novel approach but lacked in terms of sensitivity analysis and overlooked the imbalanced dataset they used in the study and did not incorporate any kind of sampling techniques that might have helped in overcoming the imbalanced dataset.

Hamori et al (2018) studied credit card delinquency using the same dataset as we have used in this study. Their study involved a comparison of ensemble learning methods along with Neural Networks and DNNs with Tanh and ReLU activations functions. They identified that the dataset used was imbalanced and used the approach of normalization rather than sampling techniques with the dataset. Secondly, their DNN model consisted of only two-layer which

ideally falls under the category of neural network and did not include higher number neurons or layers of neurons as is the case with DNN.

Zhu et al (2018) introduced the use of Relief Algorithm based Convolutional Neural Network (CNN) in the consumer credit scoring. The researchers used consumer credit data from a Chinese consumer finance company which consisted of 24,387 data points and over 570 numeric attributes. Out of these 570 numeric attributes, they used 50 attributes concerned with the consumer credit (Zhu et al, 2018). They compared the results with logistic regression and random forest which are two completely different sets of statistical techniques and machine learning algorithms respectively. Their study only included AUC and F1- Measure which indicates that the dataset used was highly imbalanced whereas their methodology did not include any data normalization or sampling techniques with the neural network.

H. Kvamme et al (2018) used a convolutional neural network to predict mortgage defaults from the consumer's account balance. They used a dataset from a Norwegian Bank, DNB consisting of 20,989 data points with a time series from 2012 to 2016. Their neural network consisted of 3 hidden layers with ReLU Activation functions with one output layer with a SoftMax activation function. For overcoming imbalanced dataset problem and overfitting of the model, they used data augmentation and regularization on the both the CNN models they used in this research. One of the major critiques of this research would on the selection of data features and use of only consumer account dataset, not financial transactions data which the customers carry out in the day to day life.

Bayraci & Susuz (2019) studied DNN-based classification models in credit risk assessment of Tunisian financial institutions in two separate datasets. For the datasets pertaining to credit card applicants, to avoid the imbalanced nature of the dependent variable, the researchers used a random selection of the major and minor classes. They identified that DNN works well with complex datasets. However, their research lacked in presenting sufficient evaluations of DNN Models in terms of F1- Measure and AUC, instead they chose to use the Weighted Average Accuracy rate. Secondly, the researchers didn’t quite specify the activation functions or the number of layers used in the DNN model used in the research.

From the literature review, it can be observed that several gaps could be outlined. Previous research on DNN Model has majorly overlooked the sampling techniques that could be

implemented along with these models. The evaluation of the models has been limited to accuracy whereas in the case of the imbalanced dataset it is recommended to use other measures like F1- Score, G-Mean and AUC – ROC Curve. Limited research has been completed on comparing the established scoring techniques like SVM with DNN models which could help us in understand whether DNN models have an advantage or not. Previous researches have been limited to presenting the outcomes of the models in terms of their performance, however limited discussion has been presented on the policy implication for the use of such models in financial institutions.

This thesis aims at filling the gaps in the literature as highlighted in Table 2.0, Table 3.0 and Table 4.0. The methodologies presented in the next chapter will outlay the sampling techniques that are used to overcome the imbalanced nature of the dataset. Evaluations techniques like F1-Score, G-Mean along with accuracy, sensitivity, and specificity for the imbalanced datasets are also presented. This thesis also intends on presenting some of the latest policies that are formulated or are in place for the use of such models for credit risk prediction and what could be done better in terms of adopting these models into real-life applications.

Table 2.0 Literature Review Gap - SVM and KNN

Author/Authors	**Models Used**	**Dataset**	**Sampling Techniques**	**Gap in the Literature**	**Literature Gap Filled by this study**
L. Yu et al (2010)	SVM with Ensemble learning, LogitR, FeedForward Neural Network	Balanced by increasing bad applicants	No	Sampling techniques	Sampling techniques
S. Chen et al (2011)	SVM with Gaussian Kernel	Imbalanced	No	Sampling techniques	Sampling techniques
Trustorff et al (2011)	SVM with Least Squares	Imbalanced	No	Sampling techniques	Sampling techniques
Wang and Ma (2012)	SVM with the hybrid ensemble	smaller - 239 instances	No	Smaller dataset	30,000 instances used in this study
Harris (2013 and 2015)	SVM	Smaller - 1000 instances	No	Smaller dataset, Sampling techniques	30,000 instances used in this study, Sampling techniques
Danenas & Garsva (2015)	PSO-SVM, SVM	Imbalanced, 24000 instances	No	Measurements for imbalanced dataset, ROC, Sampling techniques	Sampling techniques and better performance measurement techniques
Henley & Hand (1996)	KNN	Balanced	No	Performance measurements	Performance measurements under imbalanced dataset
Marinakis et. al (2008)	KNN	Imbalanced, 1411 instances	No	Sampling techniques, Smaller Dataset	Sampling techniques
Abdelmoula (2015)	KNN	N/A, 924 instances	No	No discussion on imbalanced dataset	Sampling techniques

Table 3.0 Literature Review Gap - ANN and DNN

Author/Authors	**Models Used**	**Dataset**	**Sampling Techniques**	**Gap in the Literature**	**Literature Gap Filled by this study**
Khashman (2010)	ANN	24 Attributes of Financial ratios	No	Performance measurements like ROC, F-Measure	Performance measurements under imbalanced dataset
Cimpoeru (2011)	ANN - Neural calculus, Error Back propagation techniques	Financial Ratios, No discussion on dataset's nature	No	N/A	N/A
Oreski et.al (2012)	GA- NN	Financial Ratios, No discussion on dataset's nature	No	Technology intensive	DNN model used in this study (Able to run on any laptop with 8 GB ram)
Khemakhem & Boujelbène (2015)	ANN	Financial Ratios of Tunisian Companies	No	Less Robust Model	Consistent results obtained by ANN and DNN model used in this study
Sun & Vasarhelyi (2018)	DNN (Layers not mentioned)	Credit Card delinquencies- 700,000 instances	No	Overlooked Imbalanced dataset	Sampling techniques along with DNN Model

Author/Authors	Models Used	Dataset	Sampling Techniques	Gap in the Literature	Literature Gap Filled by this study
Hamori et al (2018)	DNN - 2 Layers	Same Dataset as used in this study	No	Sampling techniques	Sampling techniques, DNN model used in this study has 4 hidden layers
Zhu et al (2018)	CNN	Chinese Consumer Finance Company, 24387 instances, Imbalanced Dataset	No	Sampling techniques, No model comparison	Sampling techniques, 4 different models used in this study
H. Kvamme et al (2018)	CNN	DNB Bank, 20989 instances, Augmentation and Regularization for imbalanced nature of the dataset	No	Sampling techniques	Sampling techniques
Bayraci & Susuz (2019)	DNN (Layers not mentioned)	Tunisian Financial Institutions, Random selection of major and minor classes	No	Sampling techniques	Sampling techniques

Table 4.0 Literature Review Gap - Deep Neural Network

4.0 Methodology

In this chapter, the methodologies used in the study are discussed in depth. Section 4.1 outlines the software used and how the models are constructed. Section 4.2 outlines the dataset used in the study. Section 4.3 discusses the sampling techniques to over the imbalanced datasets as we discussed in the literature review. Section 4.4 describes the evaluation techniques used to determine the performance of the models. Section 4.5 discusses the overall framework used in the study.

4.1 Software Used

LIBLINEAR: It is an open-source library developed by National Taiwan University in 2008, used primarily for large scale classifications (Fan et al, 2008). This software package primarily supports logistic regression and linear SVM. The package allows developers and common users with limited knowledge in programming to implement and research about the impact of classification techniques in several fields.

SCI-KIT LEARN: SciKit is a python based open-source software library distributed under the BSD licenses. The major focus of the developers of this package has been on the implementation of the models (Pedregosa et al, 2011). It provides a range of in-build algorithms for classification, regression, and clustering such as SVM, random forests, gradient boosting, and KNN along with sampling algorithms.

KERAS: Keras is described as one of the widely used python based deep learning library. This software package is capable of running along with other higher-end software packages in deep learning like TensorFlow, Theano and CNTK. Keras supports two kinds of models within its packages, one consisting of the sequential model which reflects the feed-forward neural architecture and the second one through functional API (Application programming interface) for complex models.

TensorFlow: TensorFlow was developed at Google as a part of the research project by Abadi et al (2016). TensorFlow packages were also developed in python programming language and released as an open-source software package. As per the white paper published by Abadi et al (2016), TensorFlow was primarily developed to operate at large scale computing systems and

in heterogeneous environments. Over the past years, TensorFlow has gained a lot of traction in the machine learning research community due to the ease of implementation and advanced machine learning algorithms

ECLIPSE: For conducting this study, Eclipse has been used as an integrated development environment and software packages of sci-kit-learn, TensorFlow, Keras has been integrated into the environment through PyDev-Plugins (Python Development Environment). These plugins allow for the integration of python-based software packages like sci-kit-learn, TensorFlow, Keras into Eclipse which based on Java programming language.

The models used in the study were developed by using the above-mentioned packages. The following table outlines the software package used for the corresponding models.

Table 5.0 Software used for models in this study

Models	**Software Used**
SVM – RBF Kernel	SciKit-Learn, LIBLINEAR
KNN	SciKit-Learn
Two-layer – ANN	Keras, TensorFlow, SciKit -Learn
DNN	Keras, TensorFlow, SciKit-Learn
Sampling Techniques	SciKit-Learn

4.2 Dataset

The data utilized for the research has been obtained from the University of California, Irvine Machine Learning repository which is one of the leading databases for research datasets in artificial intelligence and machine learning. The dataset contains over 30,000 rows of individual client credit cards with 23 explanatory features. These 23 explanatory features are outlined in Table 6.0. The explanatory features are based on the 30,000 client's credit card transaction that happened between April to September 2005. The response variable or the dependent variable is 'default payment next month' which indicates that the client will fail in paying any amount to the financial institution in the next month, thereby defaulting in the credit card payment.

For training and testing the models, this study uses a ratio of 80:20 for splitting the entire dataset randomly using the software package Sklearn. 80% of the dataset has been used for training the models whereas 20% of the dataset was used for testing the models. The preliminary analysis of the dataset has been explained in detail in Chapter 5. To particularly identify the explanatory features contributing towards the probability of default, the dataset has been kept consistent throughout with the ratios of testing and training datasets. The following table defines the 24 features of the dataset:

Features	Type	Explanation
LIMIT_BAL	Quantitative	Amount of the given credit (NT dollar): it includes both the individual consumer credit and his/her family (supplementary) credit.
SEX	Qualitative	Gender (1 = male; 2 = female).
EDUCATION	Qualitative	Education (0=No Education, 1 = graduate school; 2 = university; 3 = high school; 4,5,6 = others).
MARRIAGE	Qualitative	Marital status (1 = married; 2 = single; 3,0 = others).
AGE	Qualitative	Age (year)
History of past payment		The measurement scale for the repayment status is: -2=No payment required as BILL_AMT =0, -1 = pay duly; 1 = payment delay for one month; 2 = payment delay for two months; . . .; 8 = payment delay for eight months; 9 = payment delay for nine months and above.
PAY_0	Quantitative	The repayment status in September: 2005.
PAY_2	Quantitative	The repayment status in August: 2005.
PAY_3	Quantitative	The repayment status in July: 2005.
PAY_4	Quantitative	The repayment status in June: 2005.
PAY_5	Quantitative	The repayment status in May: 2005.
PAY_6	Quantitative	The repayment status in April: 2005.

Amount of bill statement		
BILL_AMT1	Quantitative	Amount of bill statement in September: 2005
BILL_AMT2	Quantitative	Amount of bill statement in August: 2005
BILL_AMT3	Quantitative	Amount of bill statement in July: 2005
BILL_AMT4	Quantitative	Amount of bill statement in June: 2005
BILL_AMT5	Quantitative	Amount of bill statement in May: 2005
BILL_AMT6	Quantitative	Amount of bill statement in April: 2005
Amount of previous payment (NT dollar)		
PAY_AMT1	Quantitative	Amount paid in September: 2005
PAY_AMT2	Quantitative	Amount paid in August: 2005
PAY_AMT3	Quantitative	Amount paid in July: 2005
PAY_AMT4	Quantitative	Amount paid in June: 2005
PAY_AMT5	Quantitative	Amount paid in May:2005
PAY_AMT6	Quantitative	Amount paid in April: 2005
default payment next month	Quantitative	Output variable/ Response Variable/Dependent variable

Source: University of California, Irvine, Retrieved from https://archive.ics.uci.edu/ml/datasets/default+of+credit+card+clients, 2019

Table 6.0 Features of the dataset used in this study

4.3 Sampling techniques

As discussed in Chapter 3, one of the gaps in the literature has been the use of sampling techniques along with the models implemented to study credit risk assessment. Sampling techniques are used to overcome the problem of an imbalanced dataset and minimize the impact of such datasets on the final outcome provided by the models. These sampling techniques can be divided into two namely, over-sampling and under-sampling techniques. Oversampling techniques helps increasing the minority class to match the majority class thereby providing balanced dataset. Under-sampling techniques helps in reducing the majority class to match the minority class.

For this study, the following oversampling and under-sampling have been used for further analysis of dataset along with models in credit risk assessment.

Over-Sampling techniques:

SMOTE – Synthetic Minority Over-Sampling Technique

SMOTE was first proposed by Chawla et al (2002) in their seminal paper on the technique. Based on google scholar's estimation over 9000 papers have cited this research, indicating the review of this technique over the past two decades. SMOTE is implemented by over-sampling the minority class and by under-sampling the majority class (Chawla et al, 2002). In this study, the minority class would be the segment of data with credit card clients defaulted in their payment and the majority class would be vice versa.

SVM – SMOTE

It is a variant of the SMOTE Algorithm which uses the SVM kernel algorithm for detecting samples and generating new synthetic samples (Karaa, Cooper and Kamei,2009). Based on our literature review, SVM-SMOTE has not been used in the literature of credit risk assessment as researchers prefer to use SMOTE as a form of oversampling and conduct a further comparison. By using one more method in this study, a comparison between these two oversampling methods can also be established.

Under-Sampling techniques:

RUS – Random Under-Sampling

Random Under Sampling has been one of the widely used under-sampling techniques in the literature we have reviewed. This technique under-samples the majority class by randomly picking samples with or without replacement.

All-KNN

All – K Nearest Neighbour (All- KNN) uses a K-Nearest neighbor algorithm to carry out the under-sampling. This technique has been developed based on the paper published by Tomek (1976). Based on our literature review, All-KNN under-sampling technique has not been previously employed to study the effect of this technique on the respective models used in this study. Using this technique in this study will allow us to establish a comparison between the two under-sampling techniques which will be used for further analysis.

Although oversampling and under sampling techniques both help in creating a balanced dataset. These two techniques have their own advantages and disadvantages while used in conjunction with the machine learning techniques. Oversampling techniques tends to become computationally intensive due to increase in the datapoints whereas with under sampling its vice versa. Oversampling helps in increasing the datapoint of the class or dependent variable which are less in the original dataset also called as minority class. Under sampling techniques results in loss of information the datapoints of major dependent variable are reduced to match the minority class where as Oversampling techniques helps in increasing the information at hand.

4.4 Performance Evaluation

To understand the model's performance with respect to each other we have outlined the following metrics for all of them. Since we identify that our dataset may be imbalanced in nature, we have also included metrics for understanding the performances of the models under such conditions.

Confusion Matrix:

Confusion matrix has been used widely to understand the segregation of true positives, false positives, true negatives and false-negative within the study of classification models. For this study, the confusion matrix defines the default payments and payments occurring in time. Following tables illustrates the confusion matrix used in this study.

Table 7.0 Confusion Matrix as used in this study

Actual Y	Predicted Y	
	Default Payment(Y=1)	Payment on Time (Y=0)
Default Payment(Y=1)	True Positive (TP)	False Negative (FN)
Payment on Time (Y=0)	False Positive (FP)	True Negative (TN)

Accuracy:

Accuracy of the classification model is the proportion of correct predictions to the total number of instances or data points used in the prediction. It is given by formula as below. The values for the accuracy range from 0 to 1 where 0 indicates the least accuracy and 1 indicates the highest accuracy of classification for positive and negative values.

$$\text{Accuracy} = \frac{(TP+TN)}{(TP+TN+FP+FN)} \quad (2)$$

Where TP stands for True positives, TN for True Negative, FP for False Positives and FN for False Negatives.

Sensitivity:

Sensitivity is known as the true positive rate is the proportion of true positives to the total number of positive instances or positive data points used in the prediction. The values for the sensitivity range from 0 to 1 where 0 indicates the least sensitivity and 1 indicates the highest sensitivity and the model is geared towards classifying positive values better.

$$\text{Sensitivity} = \frac{TP}{(TP+FN)} \quad (3)$$

Specificity:

Specificity is known as the true negative rate is the proportion of true negative to the total number of negative instances used in the study.

$$\text{Specificity} = \frac{TN}{(TN+FP)} \quad (4)$$

The above metrics are generally used among all the machine learning and neural network model evaluation and performance.

In the case of imbalanced datasets or skewed datasets, it is ideal that more appropriate metrics are used for comparison. The following metrics used in this study will allow for such comparisons.

Balanced Accuracy:

Balanced accuracy is most commonly used when dealt with imbalanced datasets. It is the arithmetic mean of sensitivity and specificity for a given model. The values for the balanced accuracy range from 0 to 1 where 0 indicates the least accuracy and 1 indicates the highest accuracy of classification for positive and negative values.

$$\text{Balanced Accuracy} = \frac{\text{Specificity}+\text{Sensitivity}}{2} \quad (5)$$

Geometric Mean:

Geometric Mean or G- mean in this context is defined as the square root of sensitivity and specificity. The values for the geometric mean range from 0 to 1 where 0 indicates the least value for Geometric mean and 1 indicates the highest value for geometric mean.

$$\text{Geometric Mean} = \sqrt{\text{Specificity x Sensitivity}} \quad (6)$$

F1-Score or Balanced F-Score or F- measure:

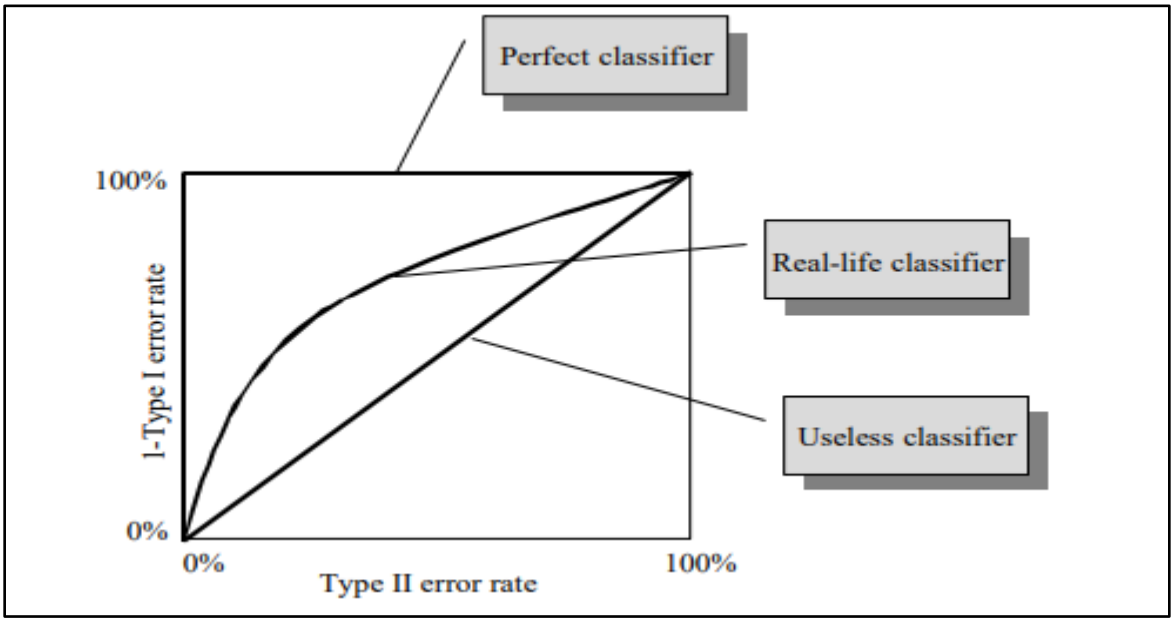
F1- Score is defined as the harmonic mean of precision and recall characteristics of the model. The best value is 1 and the worst value is 0. It is given by the below formula.

$$F1=\frac{2\ X\ Precison\ X\ Recall}{(Precision+Recall)} \quad (7)$$

Precision is the ratio of true positives to total positives including both true and false positives where recall is the ratio of true positives to true positives and false negatives.

Area Under the Curve (AUC):

The area under the curve is the measurement of the Receiver Operating Characteristic (ROC) of the model which is calculated from prediction scores. Figure 6.0 portrays an example of a ROC curve for a classifier. Any classifier which follows the 45-degree line is considered as a useless classifier. A perfect classifier classifies a default payment as "default" 100% of the time whereas real-life classifier's performance lies in between useless and perfect classifiers.



Source: Yang, 2002

Figure 6.0 Illustration of Receiver Operating Characteristics

4.5 Overall Framework

In this study, we have implemented 4 different models using 2 oversampling techniques and 2 under-sampling techniques as described in the previous sections. Before applying the models to the dataset, preprocessing of the dataset was undertaken to perform preliminary analysis and the feature selection procedure was carried out. To understand the feature importance and use them in further analysis we used logistic regression which is one of the

widely used techniques in the feature selection in the literature reviewed. Once the set number of features is selected based on the output from the logistic regression, the cleaned dataset was then passed through all the models along with sampling techniques. The following flowchart presents an outline of the overall framework used in this study.

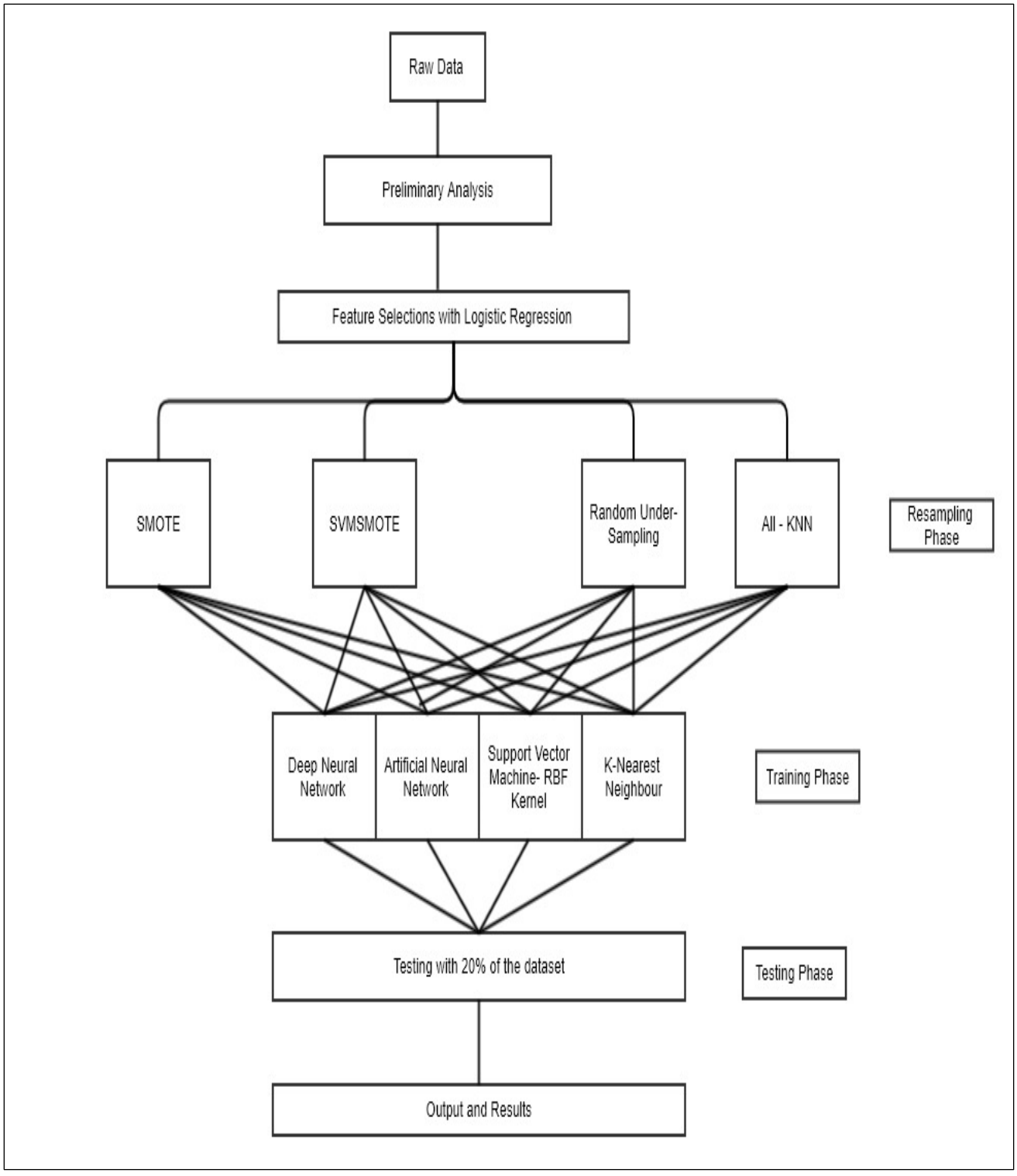


Figure 7.0 Overall Framework used in this study

5.0 Results and Analysis

In this chapter analysis of the models' output and their performance are discussed along with results from different sampling techniques used in this study. Section 5.1 outlines the preliminary analysis of the raw dataset. Section 5.2 discusses the selection of features using logistic regression. Section 5.3 outlines the model analysis using a confusion matrix for each of the models and the sampling techniques. Section 5.4 discusses the results of each model based on the performance metrics outlined in Chapter 4. Section 5.5 showcases the ROC curve achieved under each of the models and sampling techniques.

5.1 Preliminary Analysis

To understand the dataset better, a preliminary analysis was conducted on the raw dataset and several of the descriptive statistics were identified. The descriptive statistics are listed as shown in the below table. Table 8.0 shows how the dataset is distributed between default and non – default datapoints. Out of 30,000 records of clients in the dataset, 6,636 have defaulted in their payments. The percentage of the default records to total records in the dataset used to conduct this study is at 22.12 %, making the dataset an imbalanced dataset.

Table 8.0 Imbalanced Dataset

Total dataset	30000
default payments	6636
Percentage of default payments	22.12%

Table 9.0 Descriptive Statistics - Age, Sex, Education and Marriage

	SEX	**EDUCATION**	**MARRIAGE**	**AGE**
Count	30000	30000	30000	30000
Mean	1.6037	1.8531	1.5519	35.4855
Std Dev.	0.4891	0.7903	0.5220	9.2179
min	1	0	0	21
25% Conf. Int	1	1	1	28
50% Conf. Int	2	2	2	34
75% Conf. Int	2	2	2	41
Max	2	6	3	79

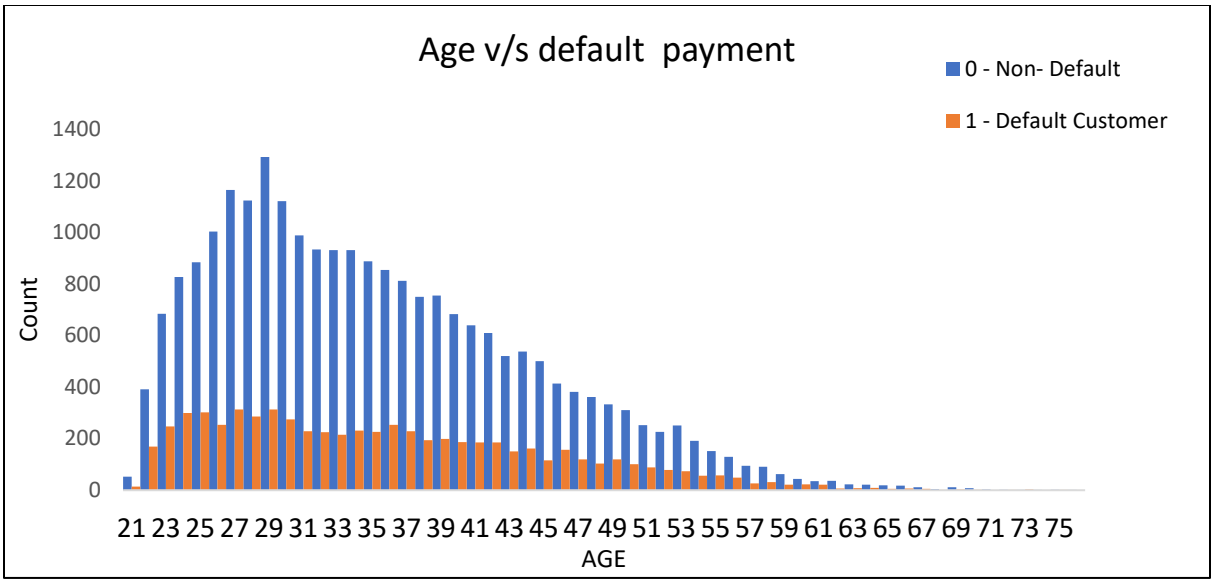


Figure 8.0 Age versus default Payment

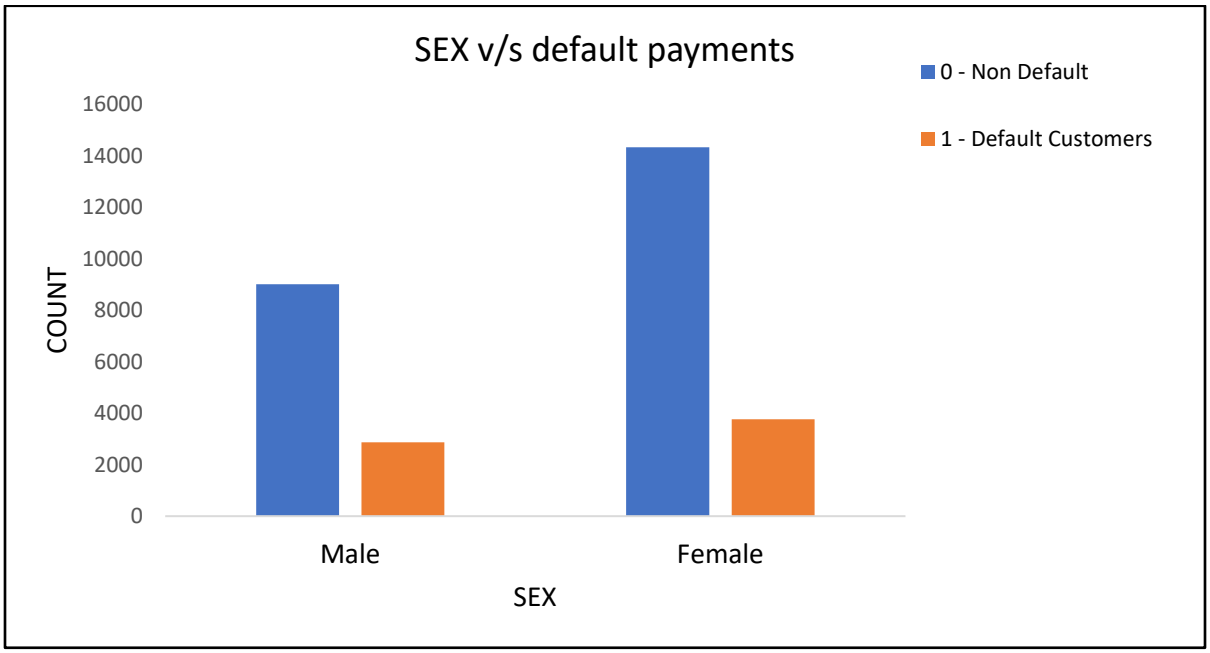


Figure 9.0 Sex versus default payment

Table 9.0 highlights the statistics of the clients regarding their age, sex, education, and marriage. The average age of the client is over 35 years with the minimum age being 21 and maximum age at 79, indicating the use of credit cards across different generations. The average education of clients is more than 1 indicating most of the clients having at least school level education. Figures 8.0, 9.0, 10.0 and 11.0 depict the count of each category against the default

payments which portrays a clearer picture of the different categories in the dataset. In these figures 0 define the non default customers represented by the blue colour and 1 defines the default customer represented by orange colour.

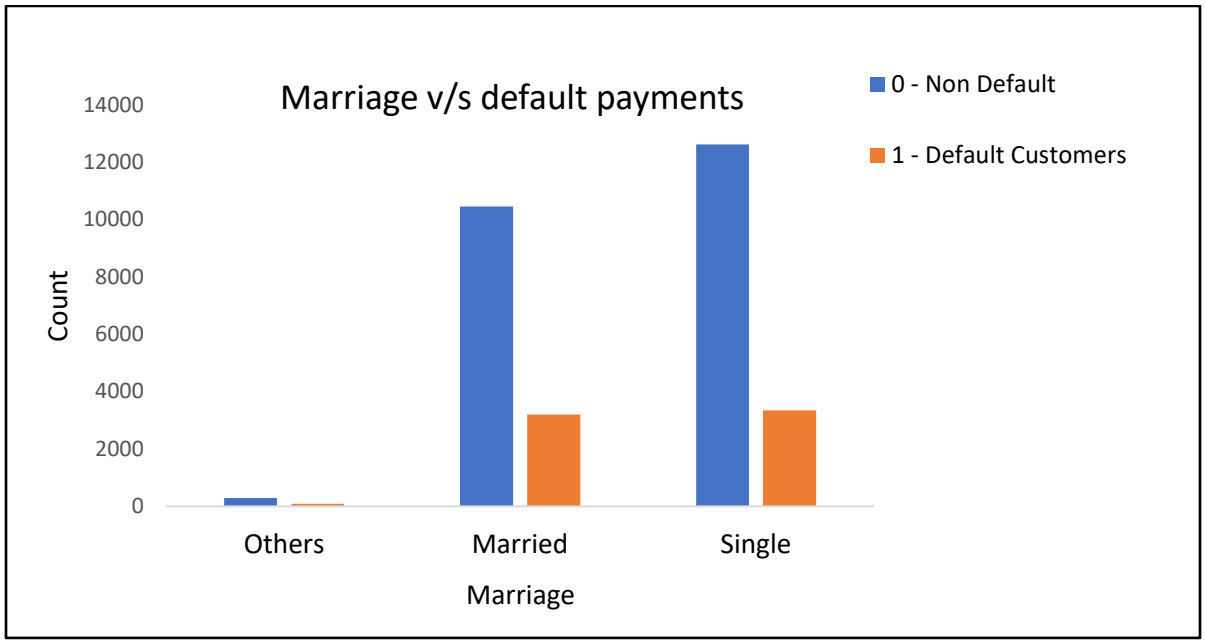


Figure 10.0 Marriage versus default Payment

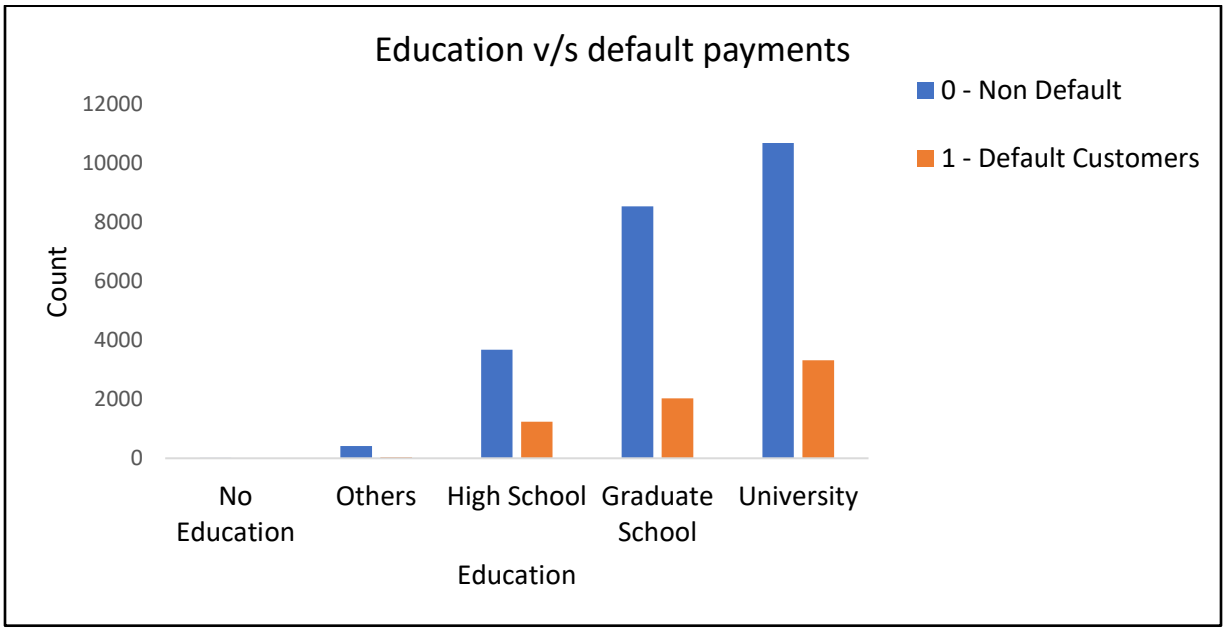


Figure 11.0 Education versus default payments

Table 10.0 Descriptive Statistics - Payment status of six months

	PAY_0	PAY_2	PAY_3	PAY_4	PAY_5	PAY_6
Count	30000	30000	30000	30000	30000	30000
Mean	-0.0167	-0.1338	-0.1662	-0.2207	-0.2662	-0.2911
Std Dev.	1.1238	1.19719	1.19687	1.16914	1.13319	1.14999
Min	-2	-2	-2	-2	-2	-2
25% Conf. Int	-1	-1	-1	-1	-1	-1
50% Conf. Int	0	0	0	0	0	0
75% Conf. Int	0	0	0	0	0	0
Max	8	8	8	8	8	8

Table 11.0 Descriptive Statistics - Amount of Bill Statements over 6 months

	BILL_AMT1	BILL_AMT2	BILL_AMT3	BILL_AMT4	BILL_AMT5	BILL_AMT6
Count	30000	30000	30000	30000	30000	30000
Mean	51223.33	49179.08	47013.15	43262.95	40311.40	38871.76
Std Dev.	73635.86	71173.77	69349.39	64332.86	60797.16	59554.11
min	-165580	-69777	-157264	-170000	-81334	-339603
25% Conf. Int	3558.75	2984.75	2666.25	2326.75	1763	1256
50% Conf. Int	22381.5	21200	20088.5	19052	18104.5	17071
75% Conf. Int	67091	64006.25	60164.75	54506	50190.5	49198.25
max	964511	983931	1664089	891586	927171	961664

Table 10.0 highlights the status of the payments of the clients over the past 6 months and how much they have delayed in payments of the credit card statements. The lowest value being -2 and highest value 8 indicating some defaulters haven't paid the bills for over 8 months. The mean across the payments holds a negative sign indicating customers who have defaulted for

a month or two may have paid the bills as well. This indicates the data consists of different combinations of the client with the payment status

Table 11.0 indicates the bill statements of the clients over the past 6 months. The average bill statements across the 6 months have been over $40,000 NT dollars indicating the expenditures and payments occurring through the credit cards. The maximum bills statements have been over $90,000 NT dollars highlighting the use of credit cards for expenses.

Table 12.0 Descriptive Statistics - Payment Amounts over 6 months

	PAY_AMT1	**PAY_AMT2**	**PAY_AMT3**	**PAY_AMT4**	**PAY_AMT5**	**PAY_AMT6**
Count	30000	30000	30000	30000	30000	30000
Mean	5663.58	5921.16	5225.68	4826.08	4799.39	5215.50
Std Dev.	16563.28	23040.87	17606.96	15666.16	15278.31	17777.47
min	0	0	0	0	0	0
25% Conf. Int	1000	833	390	296	252.5	117.75
50% Conf. Int	2100	2009	1800	1500	1500	1500
75% Conf. Int	5006	5000	4505	4013.25	4031.5	4000
max	873552	1684259	896040	621000	426529	528666

Table 12.0 highlights the payments made by clients against their bill statements over the 6 months. The minimum amount paid was 0 indicating clients who have defaulted in payments and the maximum payments have been in a wide range depending on bills with an average of over $5000 NT dollars.

5.2 Feature Selections

To eliminate noise in the dataset and to further optimize the importance of the features on the output variable, we implemented logistic regression on the raw dataset and identified that out of the 23 features in the raw dataset only 10 features played an important role in the detection of default payment. Out of the 10 variables, 6 variables were PAY_0 to PAY_6 which indicates that past repayment status plays a major role in identifying whether the client will make any future payments. It could also be stated that these repayment statuses will be

correlated with the dependent variable. The logistic regression is given by the equation (8) which includes 23 independent variables and 1 dependent variable.

$$P(Y = 1|X_1, X_2, X_3 \dots, X_{23}) = \frac{e^{\beta_0+\beta_1X_1+\beta_2X_2+\beta_3X_3 \dots\dots\dots\dots\dots+\beta_{23}X_{23}+\varepsilon_t}}{1 + e^{\beta_0+\beta_1X_1+\beta_2X_2+\beta_3X_3 \dots\dots\dots\dots\dots+\beta_{23}X_{23}+\varepsilon_t}} \quad (8)$$

where β_0 is the constant, and $\beta_1, \beta_2, \dots \dots \dots \dots, \beta_{23}$ are Coefficients of independent variables.

The independent features are labelled as $X_1, X_2, X_3 \dots, X_{23}$, and Y has been defined as the binary response for the client to be at fault $Y = 1$ or non-default when$Y = 0$.

The independent variables are defined by the characteristics of the each of the client's data included in this study. These characteristics are outlined in detail in the Table 6.0. The choice of independent and dependent variable has been made based on these characteristics and by definition of default. Based on these definitions, in this study, the dependent variable will be the default payment and independent variables are remaining features as outlined in Table 6.0. Common types of regression analysis use Mean Squared Error (MSE) as loss function that gives a convex shape. A complete optimization can be done by finding its vertex as a global minimum. However, there is no such option for logistic regression. Since the dependent feature is not continuous, the hypothesis of MSE will result in a non-convex graph with local minimums. The appropriate loss function for logistic regression is known as Cross Entropy Loss Function for linear classification models as defined by (Murphy, 2012). Such loss function also ensures that as the probability of the correct answer is maximized, the probability of the incorrect answer is minimized; since the two sum to one, any increase in the probability of the correct answer is coming at the expense of the incorrect answer. The optimized Cross Entropy Loss Function is reported by MATLAB

Using the coefficients of dependent variables obtained from the logistic regression we plotted the graph of independent variables against their relative importance. The plot of the relative importance of the features can be seen in Figure 8.0. Table 13.0 displays the logistic regression results obtained with the variables as defined in equation 8. The pseudo R-square value of 0.1207 in the table reflects the McFadden's R-Square as per the documentation of the programming used for calculating the value of the logistic regression results. Assuming, L_0 be the value of the likelihood function for a model with no predictors, and let L_m be the likelihood for the model being estimated. McFadden's R-square is defined as

$$R^2 = 1 - \left(\frac{\ln(Lm)}{\ln(L0)}\right) \quad (9)$$

As per McFadden (1974) a small ratio of the log likelihood indicates that model being estimated is far better fit than the model with no predictors. Based on the results from this step, the dataset was reduced to only 10 features which played an important role and was used for further analysis of the models.

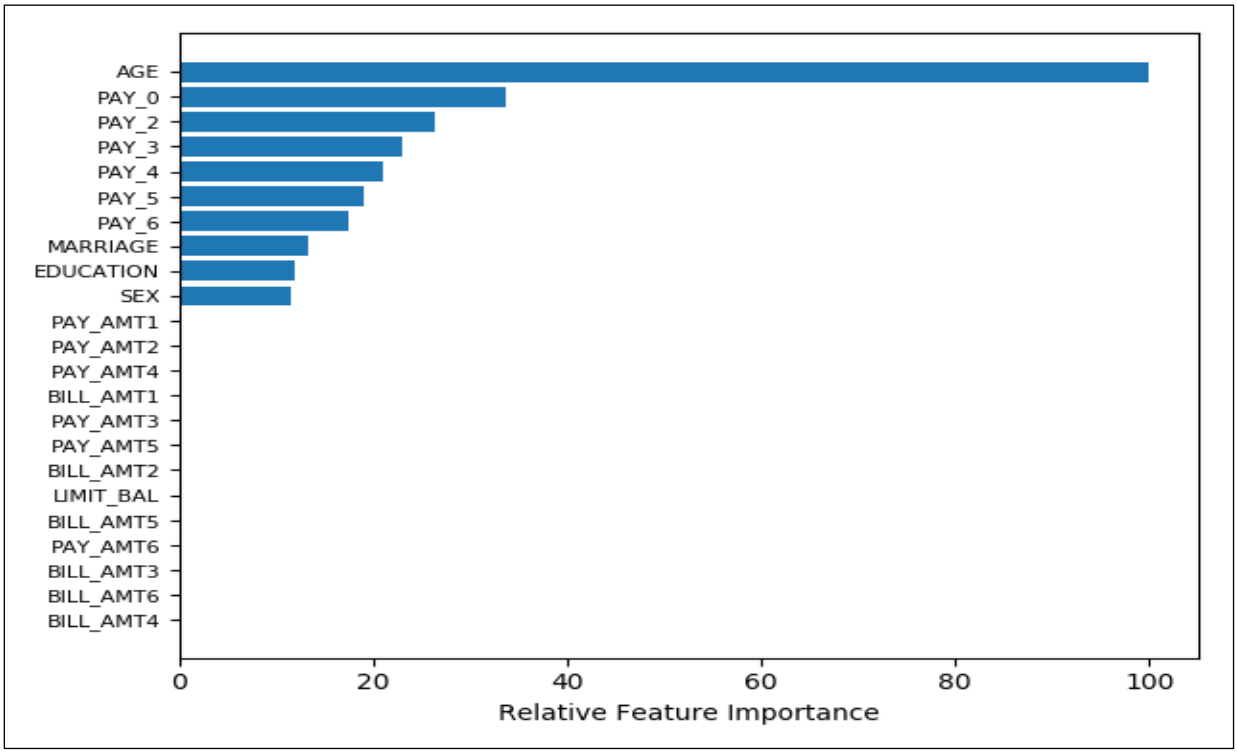


Figure 12.0 Plot of features and their relative importance using logistic regression

Table 13.0 Logistic Regression Results

Model:	**Logit**
Method:	MLE
Dep. Variable:	default payment next month
No. Observations:	30000
Df Residuals:	29976
Df Model:	23
Pseudo R-square:	0.1207
Log-Likelihood:	-13939
LL-Null:	-15853
LLR p-value:	0.00000

5.3 Model Analysis – Confusion Matrix with 10 features

The model analysis is presented with the help of metrics discussed in Chapter 4 Section 4.4. The following tables give a detailed confusion matrix of the models used in this study. Each model outlays the true positives, true negatives, false positives, and false negatives as discussed in the previous sections. These true positives, true negatives, false positives, and false negatives are generated by the models as we perform the tests on these models once the models are trained using the training dataset. For the dataset used in this study true positive detection indicates that the model was able to detect the default payment correctly, true negative indicates that the model was able to detect the non-default payment correctly, false-positive indicate that the model was not able to detect the non-default payment correctly and false-negative indicate that the model was not able to detect the default payment correctly. Table 14.0 gives a detailed confusion matrix for all the sampling techniques for the DNN model used in this study. All-KNN sampling technique has the highest true positives at 655 instances as compared to any other sampling technique with this model and SMOTE oversampling has the least true positives at 331 instances.

Table 14.0 Confusion Matrix - DNN

Model = DNN		**Predicted Y**	
Sampling	**Actual Y**	**Positive**	**Negative**
SMOTE	**TRUE**	331	4565
	FALSE	138	966
SVM SMOTE	**TRUE**	575	4291
	FALSE	412	722
RUS	**TRUE**	516	4415
	FALSE	288	781
ALLKNN	**TRUE**	655	4090
	FALSE	613	642

Table 15.0 outlays the detailed confusion matrix of ANNs with the sampling techniques. As it can be observed, the All-KNN technique has the highest number of true positives at 783

instances as compared to other techniques whereas Random Under Sampling has the least number of true positives at 496 instances.

Table 15.0 Confusion Matrix - ANN

Model = ANN		**Predicted Y**	
Sampling	**Actual Y**	**Positive**	**Negative**
SMOTE	**TRUE**	555	4266
	FALSE	437	742
SVM SMOTE	**TRUE**	558	4248
	FALSE	455	739
RUS	**TRUE**	496	4318
	FALSE	385	801
ALLKNN	**TRUE**	783	3488
	FALSE	1215	514

Table 16.0 outlays the detailed confusion matrix of SVM- RBF Kernel with the sampling techniques. As it can be observed, in this model RUS technique has the highest number of true positives at 775 instances as compared to other techniques whereas All-KNN has the least number of true positives at 450 instances.

Table 16.0 Confusion Matrix - SVM with RBF Kernel

Model = SVM - RBF Kernel		**Predicted Y**	
	Actual Y	**Positive**	**Negative**
SMOTE	**TRUE**	689	3931
	FALSE	772	608
SVM SMOTE	**TRUE**	684	4021
	FALSE	682	613
RUS	**TRUE**	775	38033
	FALSE	900	522
ALLKNN	**TRUE**	450	4476
	FALSE	227	847

Table 17.0 outlays the detailed confusion matrix of the KNN model with the sampling techniques. As it can be observed, in this model RUS technique has the highest number of true positives at 716 instances as compared to other techniques whereas SVM-SMOTE has the least number of true positives 418 instances.

Out of the 4 models, 2 models have shown the highest number of true positives and number of true negatives with All-KNN under-sampling techniques establishing that All KNN techniques detection capabilities are better than the other sampling techniques. KNN model has the highest number of true positives among all the other models indicating the model's capabilities to detect true positives among the models used in this study.

Table 17.0 Confusion Matrix - KNN

Model = KNN		**Predicted Y**	
	Actual Y	**Positive**	**Negative**
SMOTE	**TRUE**	711	3580
	FALSE	1123	586
SVM SMOTE	**TRUE**	706	3688
	FALSE	1015	591
RUS	**TRUE**	716	3761
	FALSE	942	581
ALLKNN	**TRUE**	418	4369
	FALSE	334	879

Table 18.0 provides the consolidated confusion matrix across all models and sampling techniques used in this study. The figures have been represented in percentage format to provide us with a better understanding of sampling techniques and their performance. As true positives indicate the default clients identified as default, we could observe that All -KNN technique has performed well with ANN and DNN whereas RUS has performed better with SVM and KNN in identifying true positives.

Confusion Matrix		DNN		ANN		SVM		KNN	
Sampling	**Actual Y**	**Positive**	**Negative**	**Positive**	**Negative**	**Positive**	**Negative**	**Positive**	**Negative**
SMOTE	**TRUE**	5.52%	76.08%	9.25%	71.10%	11.48%	65.52%	11.85%	59.67%
	FALSE	2.30%	16.10%	7.28%	12.37%	12.87%	10.13%	18.72%	9.77%
SVM SMOTE	**TRUE**	9.58%	71.52%	9.30%	70.80%	11.40%	67.02%	11.77%	61.47%
	FALSE	6.87%	12.03%	7.58%	12.32%	11.37%	10.22%	16.92%	9.85%
RUS	**TRUE**	8.60%	73.58%	8.27%	71.97%	**12.92%**	633.88%	**11.93%**	62.68%
	FALSE	4.80%	13.02%	6.42%	13.35%	15.00%	8.70%	15.70%	9.68%
ALLKNN	**TRUE**	**10.92%**	68.17%	**13.05%**	58.13%	7.50%	74.60%	6.97%	72.82%
	FALSE	10.22%	10.70%	20.25%	8.57%	3.78%	14.12%	5.57%	14.65%

Table 18.0 Consolidated Confusion Matrix

5.4 Performance Metrics Analysis

Table 18.0 outlays the performance metrics for each of the sampling technique with the DNN Model. As it can be observed, under most of the sampling technique DNN Model has been able to give an accuracy of 81% with the RUS-DNN model providing the highest accuracy at 82.18%. Based on balanced accuracy and G-Mean, All-KNN based DNN model has the highest performance metrics. All the sampling techniques under the DNN model were able to achieve an ROC score of 0.70 except for the SVMSMOTE technique which achieved 0.686 ROC score. The average accuracy for the techniques was at 81%, balanced accuracy at 66.16% and ROC score of 0.698.

Table 19.0 Performance Metrics - DNN

Model = DNN					
	SMOTE	**SVMSMOTE**	**RUS**	**All KNN**	**Average**
Accuracy	0.8160	0.8110	0.8218	0.7908	0.8099
Specificity	0.9707	0.9124	0.9388	0.8697	0.9229
Sensitivity	0.2552	0.4433	0.3978	0.5050	0.4003
Balanced Accuracy	0.6130	0.6779	0.6683	0.6874	0.6616
Geometric Mean	0.4977	0.6360	0.6111	0.6627	0.6019
Precision	0.7058	0.5826	0.6418	0.5166	0.6117
Recall	0.2552	0.4433	0.3978	0.5050	0.4003
F1	0.3749	0.5035	0.4912	0.5107	0.4701
Area Under the ROC Curve					
Training	0.700	0.699	0.705	0.698	0.701
Testing	0.701	0.686	0.706	0.698	0.698

Following figures show the ROC Curve for each of the techniques under the DNN Model

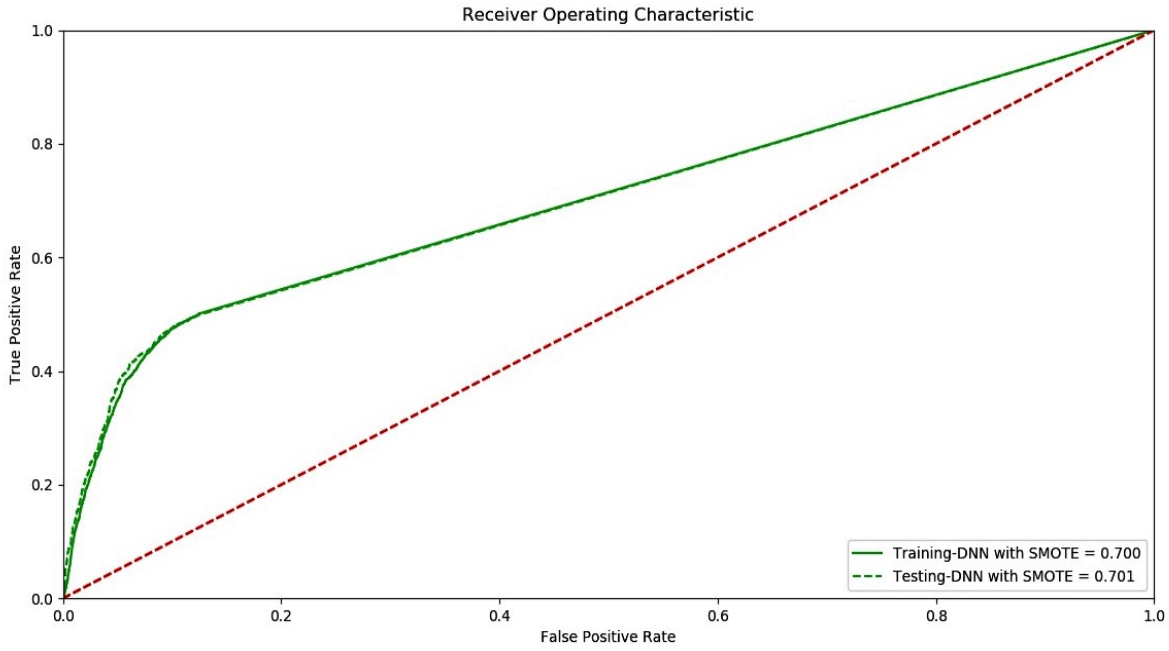


Figure 13.0 Receiver Operating Characteristics - DNN with SMOTE

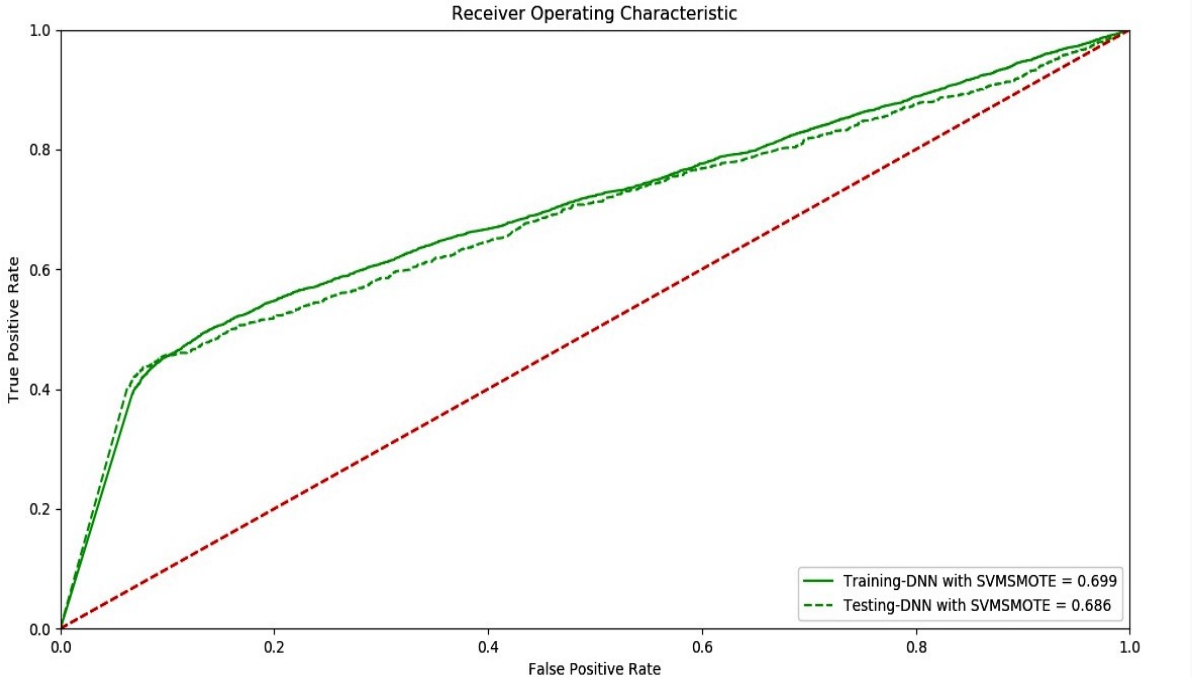


Figure 14.0 Receiver Operating Characteristics - DNN with SVM SMOTE

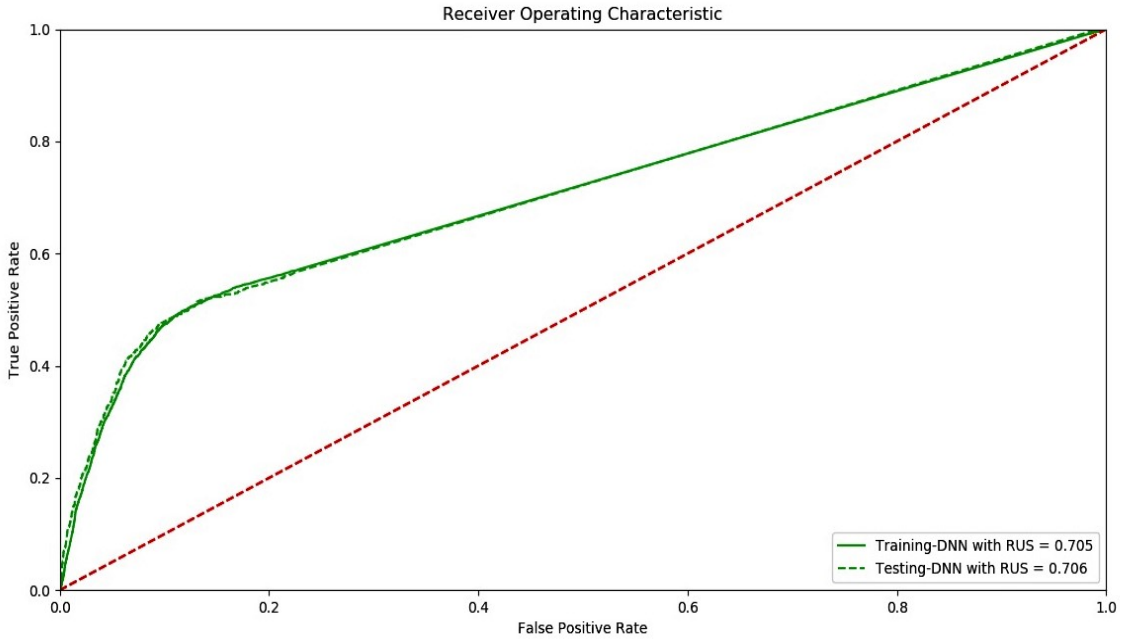


Figure 15.0 Receiver Operating Characteristics - DNN with RUS

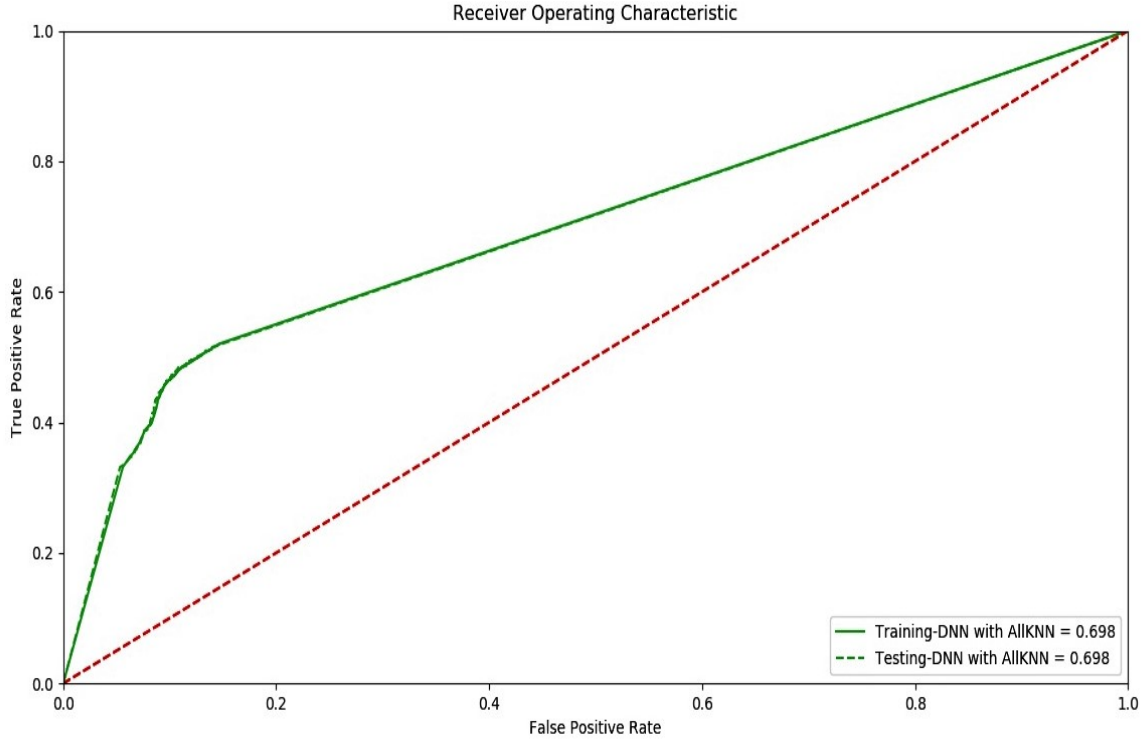


Figure 16.0 Receiver Operating Characteristics - DNN with All-KNN

Table 19.0 outlays the performance metrics for each of the sampling technique with the ANN Model. As it can be observed, for this model SMOTE, SVM-SMOTE and RUS techniques were able to give more than 80% accuracy whereas All-KNN lagged in accuracy. ANN Model has been able to give an accuracy of 80.10% with SVMSMOTE - technique. Based on balanced accuracy and G-Mean, AllKNN based ANN model has the highest performance metrics. All the sampling techniques under the ANN model were able to achieve an ROC score of 0.70 except for the RUS technique which achieved 0.691 ROC score. The average accuracy for the techniques was at 77.97%, balanced accuracy at 66.43% and ROC score of 0.703.

Table 20.0 Performance Metrics - ANN

Model = ANN					
	SMOTE	**SVMSMOTE**	**RUS**	**AllKNN**	**Average**
Accuracy	0.8035	0.8010	0.8023	0.7118	0.7797
Specificity	0.9071	0.9033	0.9181	0.7417	0.8676
Sensitivity	0.4279	0.4302	0.3824	0.6037	0.4611
Balanced Accuracy	0.6675	0.6668	0.6503	0.6727	0.6643
Geometric Mean	0.6230	0.6234	0.5925	0.6692	0.6270
Precision	0.5595	0.5508	0.5630	0.3919	0.5163
Recall	0.4279	0.4302	0.3824	0.6037	0.4611
F1	0.4849	0.4831	0.4555	0.4753	0.4747
Area Under the ROC Curve					
Training	0.707	0.708	0.686	0.708	0.702
Testing	0.706	0.707	0.691	0.706	0.703

Following figures show the ROC Curve for each of the techniques under the ANN Model

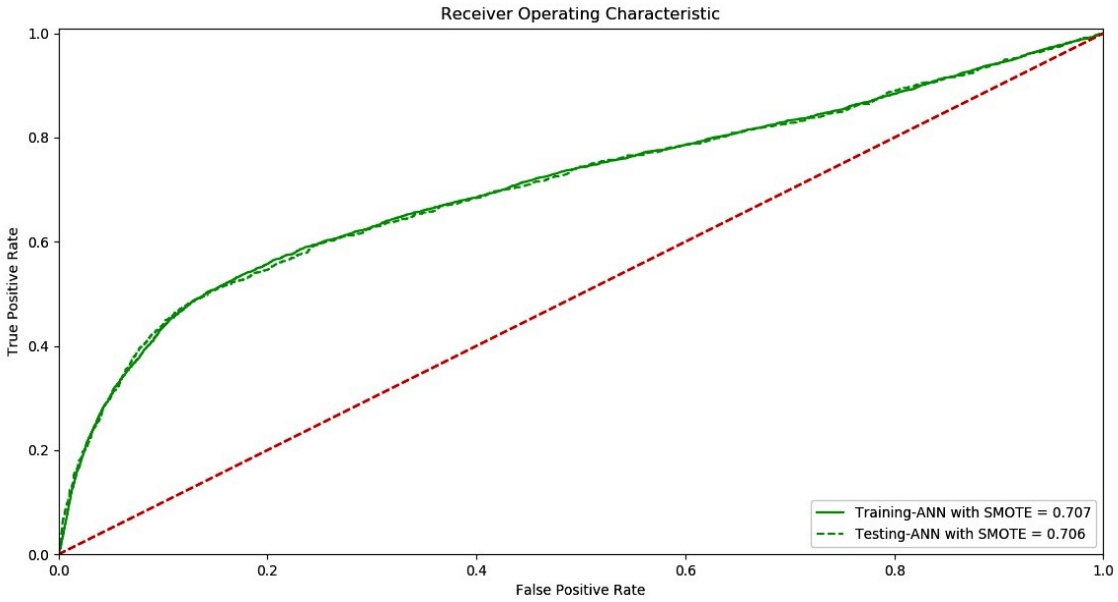


Figure 17.0 Receiver Operating Characteristics - ANN with SMOTE

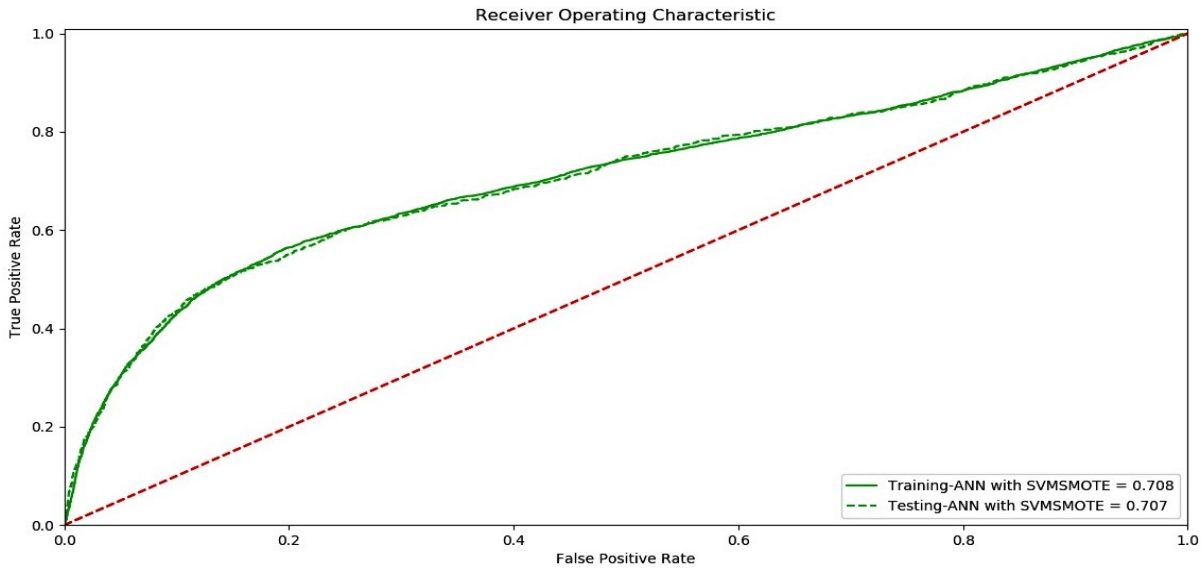


Figure 18.0 Receiver Operating Characteristics - ANN with SVM SMOTE

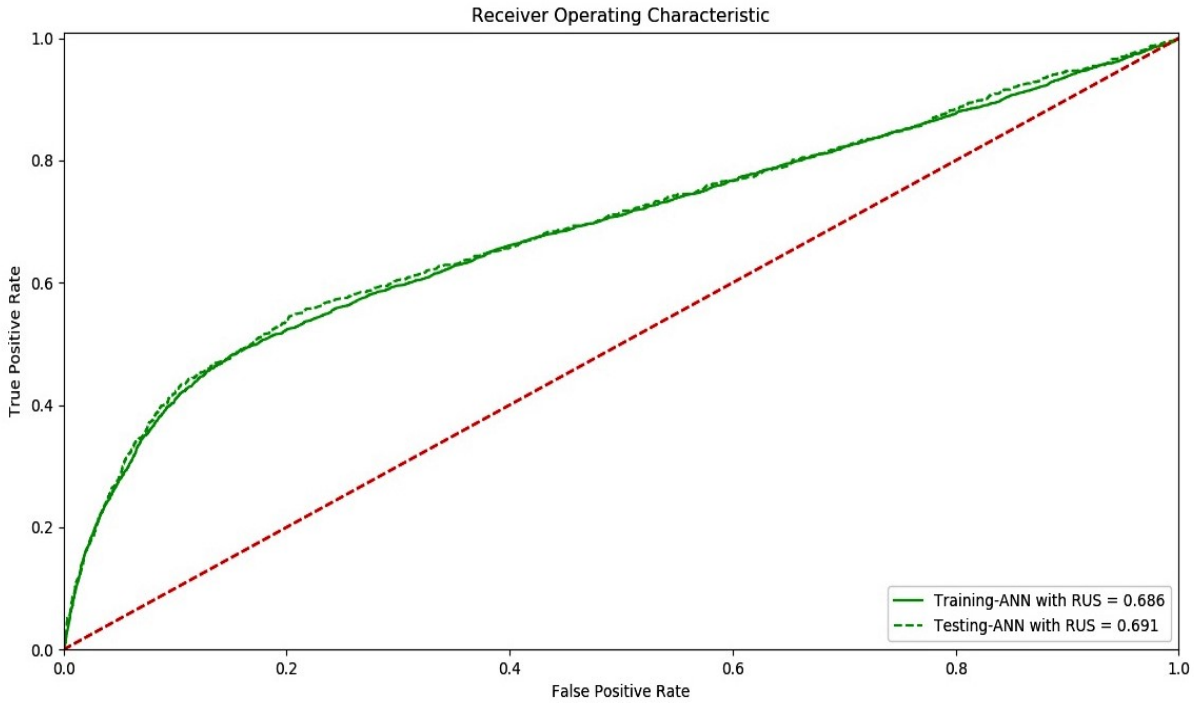


Figure 19.0 Receiver Operating Characteristics - ANN with RUS

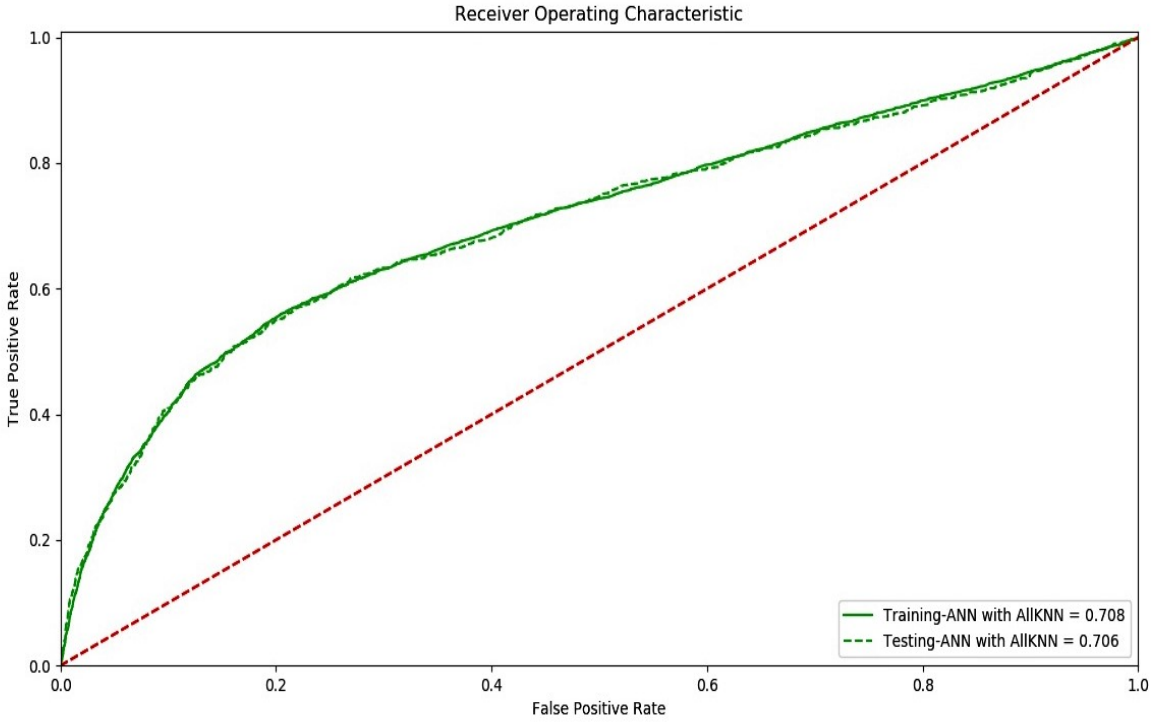


Figure 20.0 Receiver Operating Characteristics - ANN with All-KNN

Table 20.0 outlays the performance metrics for each of the sampling technique with the RBF Kernel-based Support Vector Machine. As it can be observed, under this model All-KNN has achieved more than 80% accuracy whereas SMOTE and RUS have achieved closer to 77% accuracy. Based on Balanced Accuracy and G-Mean, RUS has performed much better than other techniques with this model. All the techniques were able to achieve more than 0.69 of the Testing - ROC scores except the AllKNN technique which achieved 0.649. Taking the average on all the techniques, the model was able to achieve more than 78.46% accuracy, with a balanced accuracy of over 68.18% and 65.19% of G-Mean.

Table 21.0 Performance Metrics - SVM- RBF Kernel

Model = SVM-RBF					
	SMOTE	**SVMSMOTE**	**RUS**	**AllKNN**	**Average**
Accuracy	0.7700	0.7842	0.7630	0.8210	0.7846
Specificity	0.8358	0.8550	0.8086	0.9517	0.8628
Sensitivity	0.5312	0.5274	0.5975	0.3470	0.5008
Balanced Accuracy	0.6835	0.6912	0.7031	0.6494	0.6818
Geometric Mean	0.6663	0.6715	0.6951	0.5747	0.6519
Precision	0.4716	0.5007	0.4627	0.6647	0.5249
Recall	0.5312	0.5274	0.5975	0.3470	0.5008
F1	0.4996	0.5137	0.5215	0.4559	0.4977
Area Under the ROC Curve					
Training	0.733	0.730	0.721	0.653	0.709
Testing	0.684	0.691	0.703	0.649	0.682

Following figures show the ROC Curve for each of the techniques under the Support Vector Machine – RBF Kernel

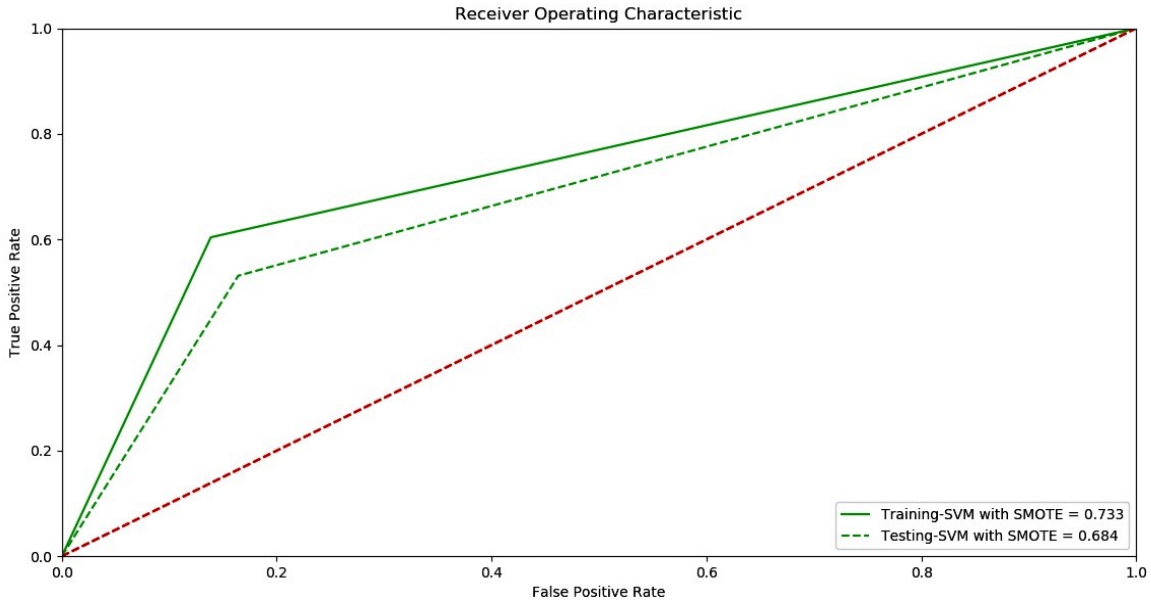


Figure 21.0 Receiver Operating Characteristics - Support Vector Machine - RBF Kernel with SMOTE

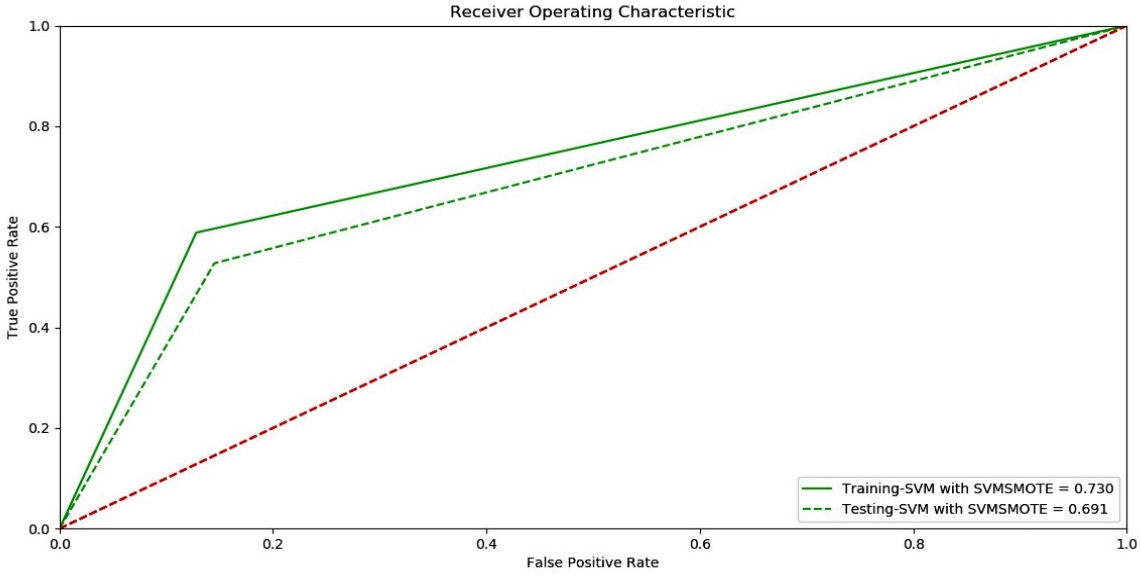


Figure 22.0 Receiver Operating Characteristics - Support Vector Machine - RBF Kernel with SVM SMOTE

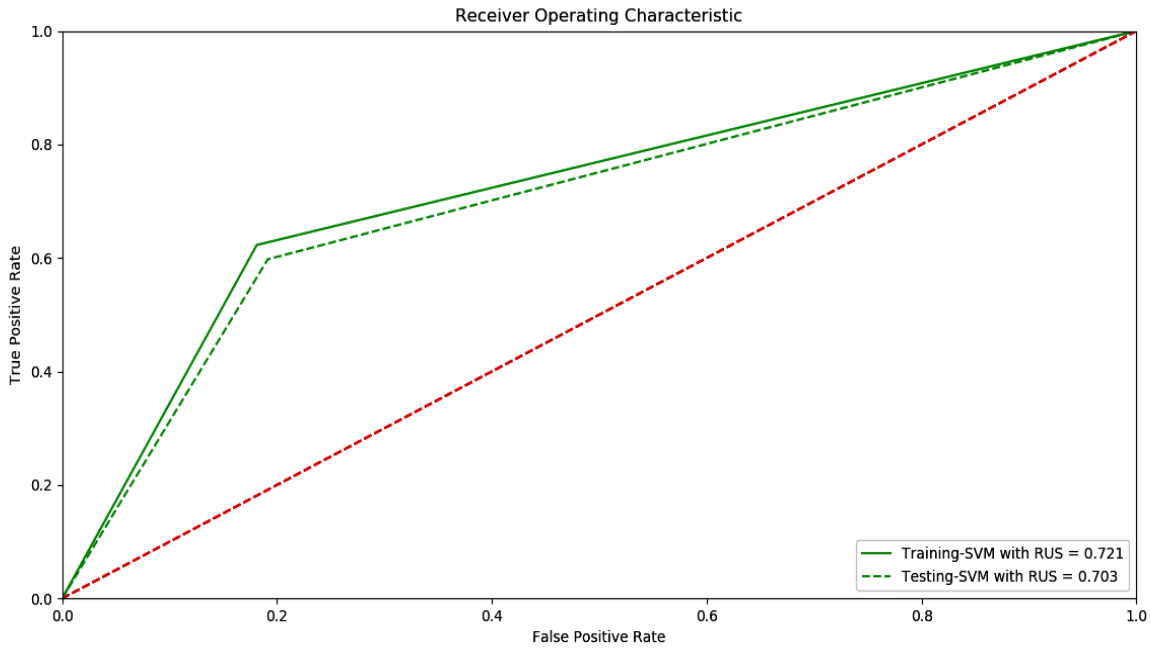


Figure 23.0 Receiver Operating Characteristics - Support Vector Machine - RBF Kernel with RUS

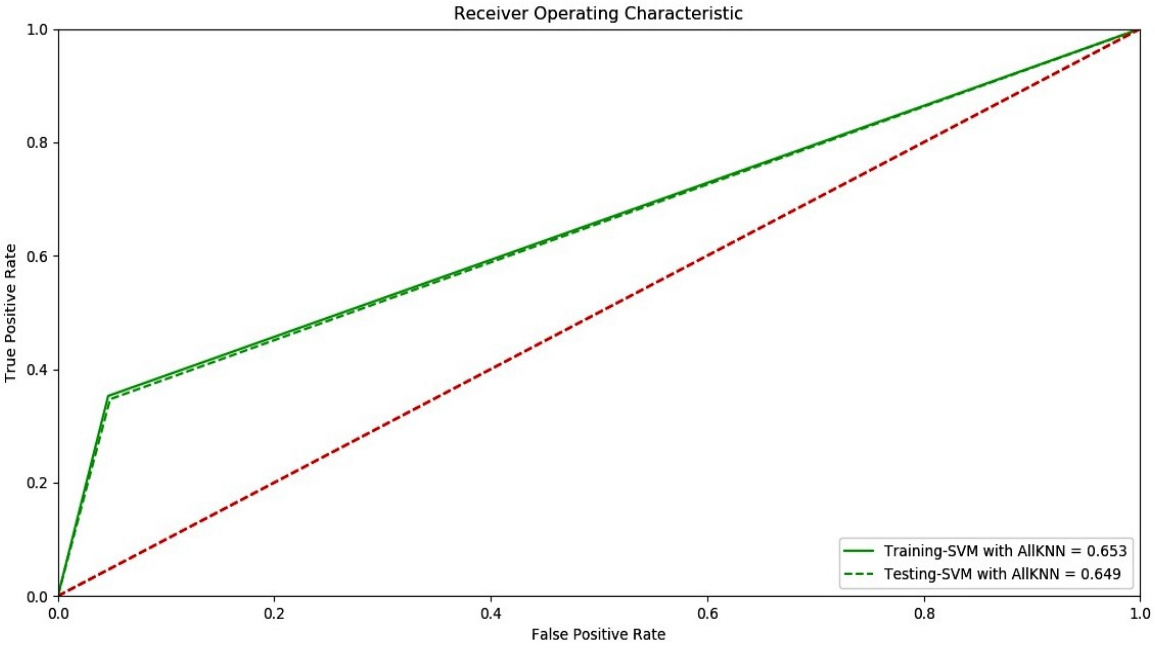


Figure 24.0 Receiver Operating Characteristics - Support Vector Machine - RBF Kernel with All-KNN

Table 21.0 outlays the performance metrics for each of the sampling technique with the KNN. KNN was able to achieve the least accuracy of all models, even in terms of balanced accuracy and G-Mean. Within the techniques, it could be observed that AllKNN has performed better than other sampling techniques with 79.78% and SMOTE has the least accuracy at 71.52%. On average of all the techniques, the KNN model was able to achieve 74.79% accuracy. All the techniques have performed differently in terms of the Testing-ROC score. Oversampling techniques have scored more than 0.65 whereas under-sampling techniques have scored more than 0.67 except the AllKNN technique with 0.626 ROC score.

Table 22.0 Performance Metrics - KNN

Model = KNN					
	SMOTE	**SVMSMOTE**	**RUS**	**AllKNN**	**Average**
Accuracy	0.7152	0.7323	0.7462	0.7978	0.7479
Specificity	0.7612	0.7842	0.7997	0.9290	0.8185
Sensitivity	0.5482	0.5443	0.5520	0.3223	0.4917
Balanced Accuracy	0.6547	0.6643	0.6759	0.6257	0.6551
Geometric Mean	0.6460	0.6533	0.6644	0.5472	0.6277
Precision	0.3877	0.4102	0.4318	0.5559	0.4464
Recall	0.5482	0.5443	0.5520	0.3223	0.4917
F1	0.4542	0.4679	0.4846	0.4080	0.4537
Area Under the ROC Curve					
Training	0.750	0.747	0.710	0.638	0.711
Testing	0.655	0.664	0.676	0.626	0.655

Following figures show the ROC Curve for each of the techniques under the KNN Model

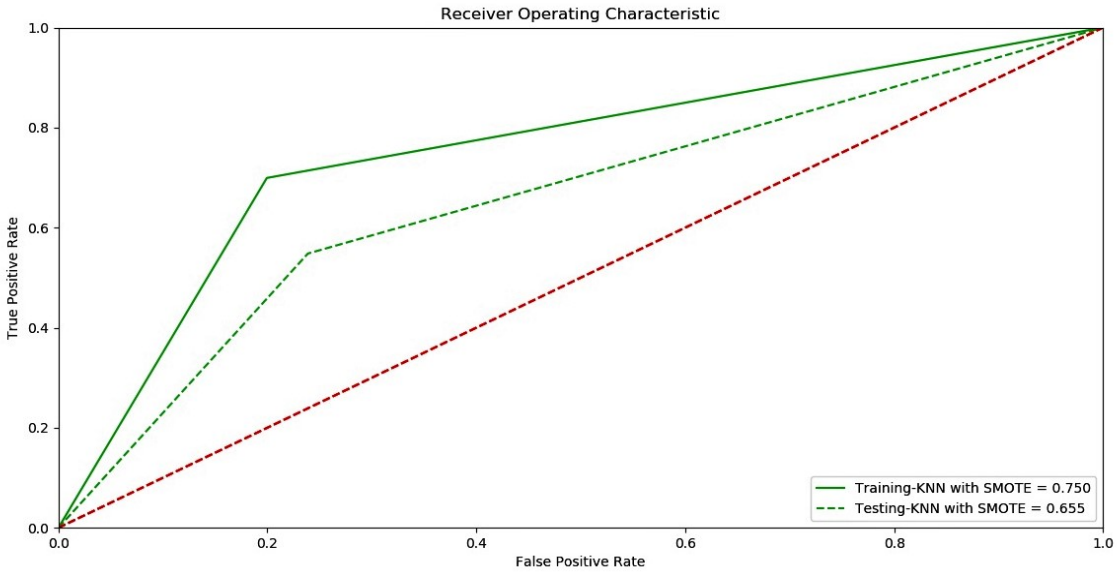


Figure 25.0 Receiver Operating Characteristics - KNN with SMOTE

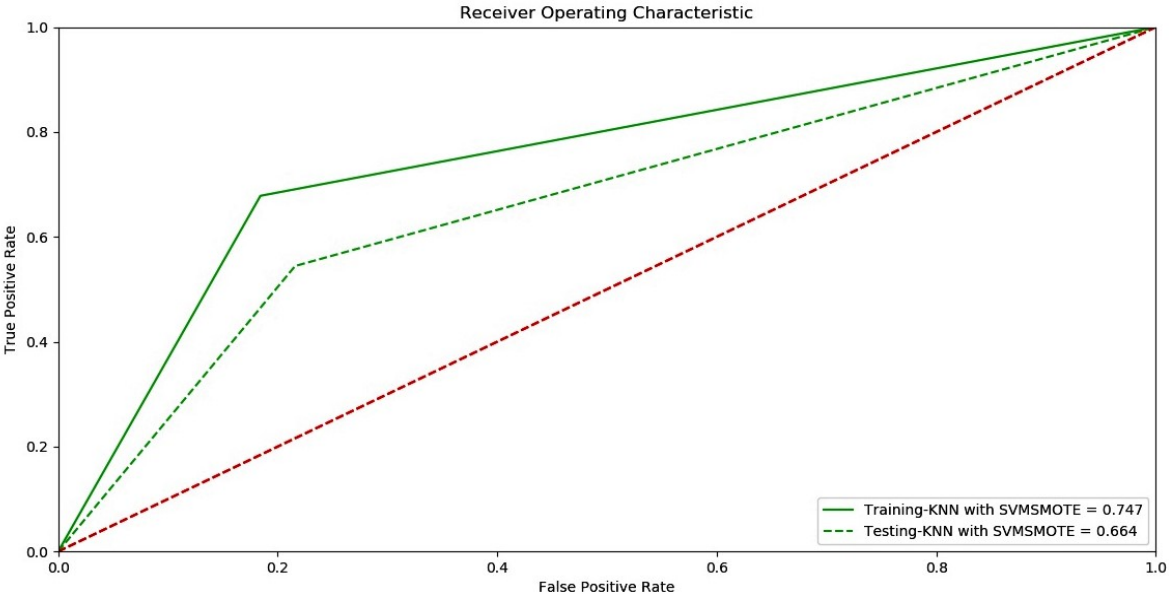


Figure 26.0 Receiver Operating Characteristics - KNN with SVM SMOTE

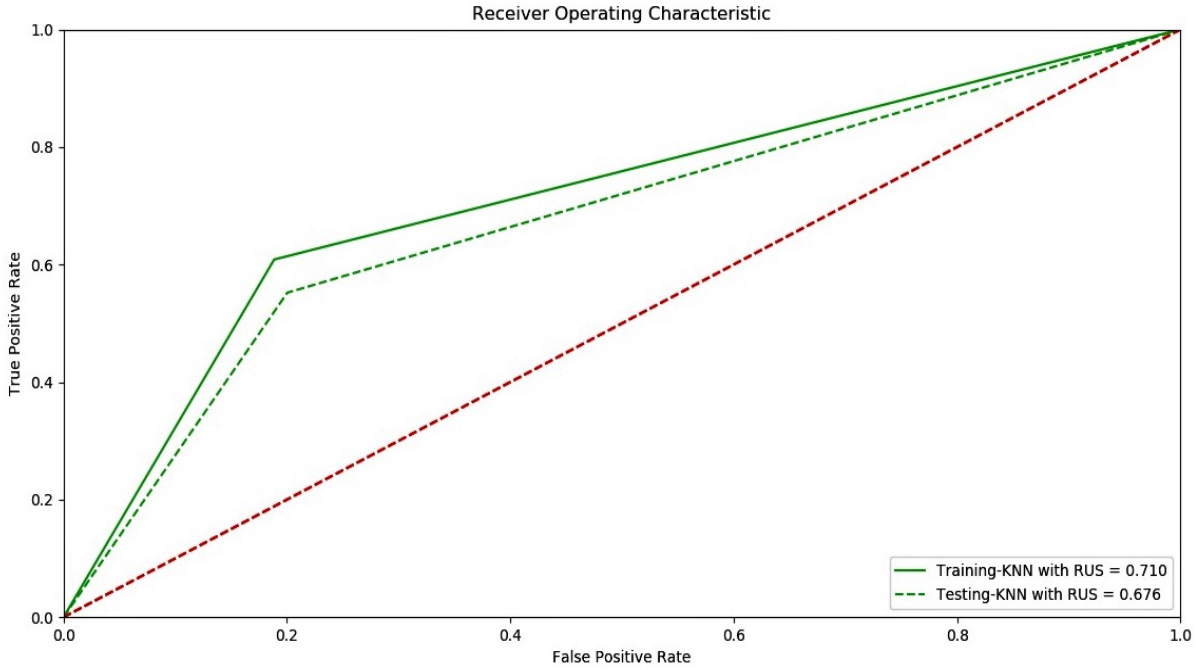


Figure 27.0 Receiver Operating Characteristics - KNN with RUS

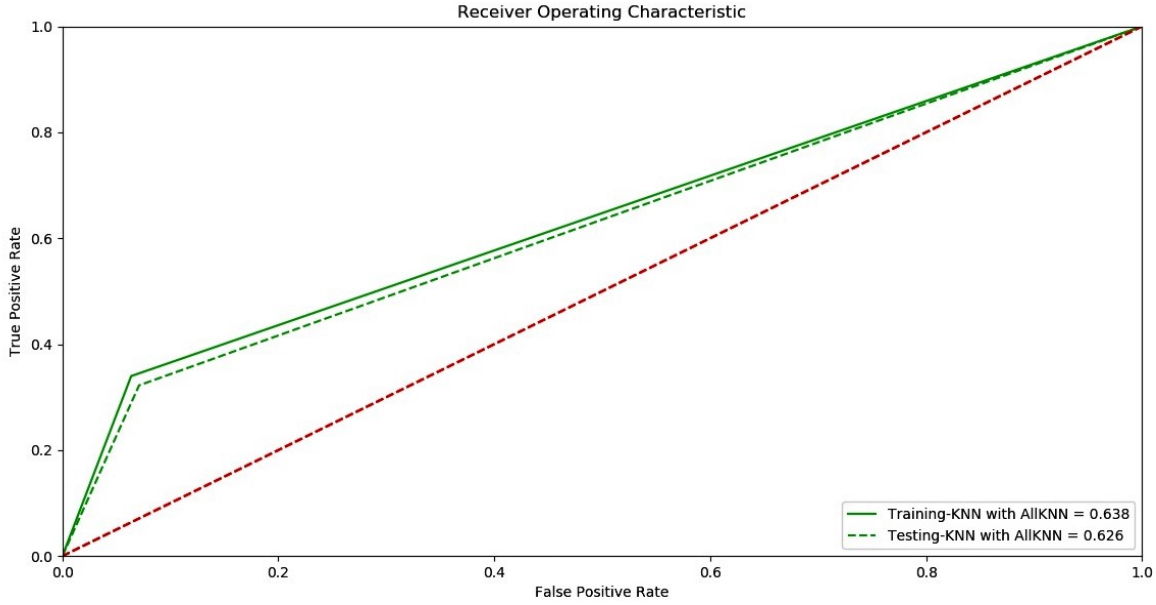


Figure 28.0 Receiver Operating Characteristics - KNN with All-KNN

Table 23.0 Consolidated Accuracies of the Models

Sampling	**SMOTE**	**SVMSMOTE**	**RUS**	**All KNN**	**Average**
DNN - Accuracy	0.8160	0.8110	**0.8218**	0.7908	**0.8099**
ANN - Accuracy	0.8035	0.8010	0.8023	0.7118	0.7797
SVM - Accuracy	0.7700	0.7842	0.7630	**0.8210**	0.7846
KNN - Accuracy	0.7152	0.7323	0.7462	0.7978	0.7479

Table 24.0 Consolidated Balanced Accuracies of the Models

Sampling	**SMOTE**	**SVMSMOTE**	**RUS**	**All KNN**	**Average**
DNN - BA	0.6130	0.6779	0.6683	0.6874	0.6616
ANN - BA	0.6675	0.6668	0.6503	0.6727	0.6643
SVM - BA	0.6835	0.6912	**0.7031**	0.6494	**0.6818**
KNN - BA	0.6547	0.6643	0.6759	0.6257	0.6551

Table 25.0 Consolidated ROC Scores of the Models

Model	**Sampling**	**SMOTE**	**SVMSMOTE**	**RUS**	**All KNN**	**Average**
DNN - ROC	Training	0.700	0.699	**0.705**	0.698	0.701
	Testing	0.701	0.686	**0.706**	0.698	0.698
ANN - ROC	Training	**0.707**	**0.708**	0.686	**0.708**	0.702
	Testing	**0.706**	**0.707**	0.691	**0.706**	0.703
SVM - ROC	Training	0.733	0.730	0.721	0.653	0.709
	Testing	0.684	0.691	0.703	0.649	0.682
KNN - ROC	Training	0.750	0.747	0.710	0.638	0.711
	Testing	0.655	0.664	0.676	0.626	0.655

Table 23.0, 24.0 and 25.0 provides the consolidated accuracies, balanced accuracies and ROC scores of the models and the sampling techniques. Based on these tables we could identify that in terms of accuracies DNN and ANN has performed better whereas SVM has performed better in terms of Balanced accuracy. ROC scores of ANN and DNN models are much better as compared to SVM and KNN. To understand the framework and to study the effect of the remaining features on the models we applied all the models and sampling techniques to the dataset with 23 independent features and 1 dependent variable. The following tables and figures will outlay the confusion matrices, performance metrics and ROC curves for each of the models and sampling techniques.

5.5 Confusion Matrices with 23 features

Table 26.0 gives a detailed confusion matrix for all the sampling techniques for the DNN with 24 features. As it can be observed that introducing the remaining features has introduced noise in the dataset increasing the loss functions in the DNN Model. The model was not able to recognize any true positives across all the sampling techniques used. As it can be seen consistently across all the sampling techniques, we can conclude that additional features have taken away the ability of the model to detect default payments accurately.

Table 26.0 Confusion Matrix with 23 features - DNN

Model = DNN		**Predicted Y**	
Sampling	**Actual Y**	**Positive**	**Negative**
SMOTE	**TRUE**	0	4703
	FALSE	0	1297
SVM SMOTE	**TRUE**	0	4703
	FALSE	0	1297
RUS	**TRUE**	0	4703
	FALSE	0	1297
ALLKNN	**TRUE**	0	4703
	FALSE	0	1297

Table 27.0 gives a detailed confusion matrix for all the sampling techniques for the ANN with 23 features. The results of the confusion matrix for both ANN Model and the DNN model

has been consistent across all sampling techniques. The model was not able to recognize any true positives across all the sampling techniques used. One of the reasons for these results may also be due to the use of the same activation functions in both the ANN and DNN Model. This will require further investigation into the activation and loss functions of both the models which are currently out of scope for this study.

Table 27.0 Confusion Matrix with 23 features - ANN

Model = ANN		**Predicted Y**	
Sampling	**Actual Y**	**Positive**	**Negative**
SMOTE	**TRUE**	0	4703
	FALSE	0	1297
SVM SMOTE	**TRUE**	0	4703
	FALSE	0	1297
RUS	**TRUE**	0	4703
	FALSE	0	1297
ALLKNN	**TRUE**	0	4703
	FALSE	0	1297

Table 28.0 outlays the detailed confusion matrix of SVM- RBF Kernel with the 24 features. SVM has been able to improve on the introduction of the features but was not successful as with 10 features. As it can be observed, in this model RUS technique has the highest number of true positives at 67 instances as compared to other techniques whereas All-KNN has the least number of true positives at 10 instances. All the sampling techniques have more 4600 true negatives which indicate the effect on the model due to features introduction.

Table 28.0 Confusion Matrix with 23 features - SVM with RBF Kernel

Model = SVM - RBF Kernel		**Predicted Y**	
	Actual Y	**Positive**	**Negative**
SMOTE	**TRUE**	52	4636
	FALSE	67	1245
SVM SMOTE	**TRUE**	48	4644
	FALSE	59	1249
RUS	**TRUE**	67	4617
	FALSE	86	1230
ALLKNN	**TRUE**	10	4695
	FALSE	8	1287

Table 29.0 outlays the detailed confusion matrix of KNNs with the 24 features. KNN has shown much better results in terms of the true positive detection as compared to the other models but was not as successful as with only 10 features. As it can be observed, in this model RUS technique has the highest number of true positives at 716 instances as compared to other techniques whereas All-KNN has the least number of true positives at 94 instances. All-KNN sampling technique under this model has the greatest number of true negatives at 4604 whereas other sampling techniques have more than 3000 true negatives. Out of the 4 models, DNN and ANN are the most affected models due to the introduction of additional features whereas KNN is the least affected model.

Table 29.0 Confusion Matrix with 23 features -KNN

Model = KNN		**Predicted Y**	
	Actual Y	**Positive**	**Negative**
SMOTE	**TRUE**	649	3089
	FALSE	1614	648
SVM SMOTE	**TRUE**	550	3458
	FALSE	1245	747
RUS	**TRUE**	716	3054
	FALSE	1649	581
ALLKNN	**TRUE**	94	4604
	FALSE	99	1203

Table 30.0 Consolidated Confusion Matrix - 23 features

Confusion Matrix		DNN		ANN		SVM		KNN	
Sampling	**Actual Y**	**Positive**	**Negative**	**Positive**	**Negative**	**Positive**	**Negative**	**Positive**	**Negative**
SMOTE	**TRUE**	0.00%	78.38%	0.00%	78.38%	0.87%	77.27%	10.82%	51.48%
	FALSE	0.00%	21.62%	0.00%	21.62%	1.12%	20.75%	26.90%	10.80%
SVM SMOTE	**TRUE**	0.00%	78.38%	0.00%	78.38%	0.80%	77.40%	9.17%	57.63%
	FALSE	0.00%	21.62%	0.00%	21.62%	0.98%	20.82%	20.75%	12.45%
RUS	**TRUE**	0.00%	78.38%	0.00%	78.38%	1.12%	76.95%	11.93%	50.90%
	FALSE	0.00%	21.62%	0.00%	21.62%	1.43%	20.50%	27.48%	9.68%
ALLKNN	**TRUE**	0.00%	78.38%	0.00%	78.38%	0.17%	78.25%	1.57%	76.73%
	FALSE	0.00%	21.62%	0.00%	21.62%	0.13%	21.45%	1.65%	20.05%

5.6 Performance Metrics with 23 features

Table 31.0 outlays the performance metrics for the DNN Model with 23 features. As it can be observed, we have received consistent accuracy of 78.38% mainly due to true negatives with sensitivity at 1.000 and balanced accuracy at 50%. ROC score for all the techniques has been flat 0.50 which indicates that with 23 features DNN Model is a useless classifier and cannot be used for further applications.

Table 31.0 Performance Metrics with 23 features - DNN

Model = DNN					
	SMOTE	**SVMSMOTE**	**RUS**	**AllKNN**	**Average**
Accuracy	0.7838	0.7838	0.7838	0.7838	0.7838
Specificity	1.0000	1.0000	1.0000	1.0000	1.0000
Sensitivity	0.0000	0.0000	0.0000	0.0000	0.0000
Balanced Accuracy	0.5000	0.5000	0.5000	0.5000	0.5000
Geometric Mean	0.0000	0.0000	0.0000	0.0000	0.0000
Precision	0.0000	0.0000	0.0000	0.0000	0.0000
Recall	0.0000	0.0000	0.0000	0.0000	0.0000
F1	0.0000	0.0000	0.0000	0.0000	0.0000
Area Under the ROC Curve					
Training	0.500	0.500	0.500	0.500	0.500
Testing	0.500	0.500	0.500	0.500	0.500

Following figures show the ROC Curve for each of the techniques under the DNN Model with 23 features

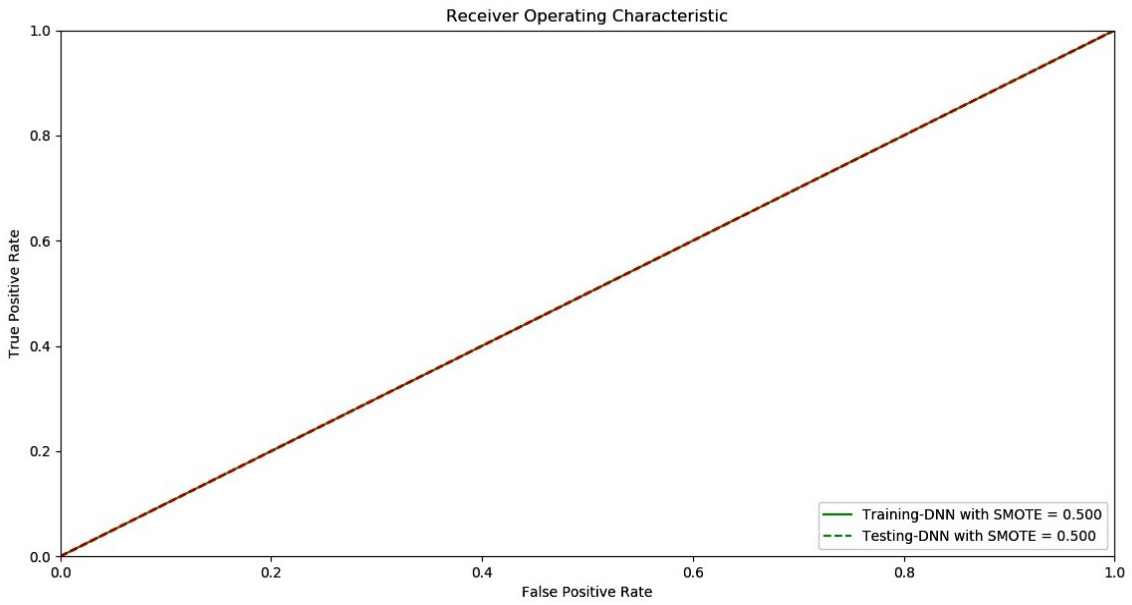


Figure 29.0 Receiver Operating Characteristics with 23 features - DNN with SMOTE

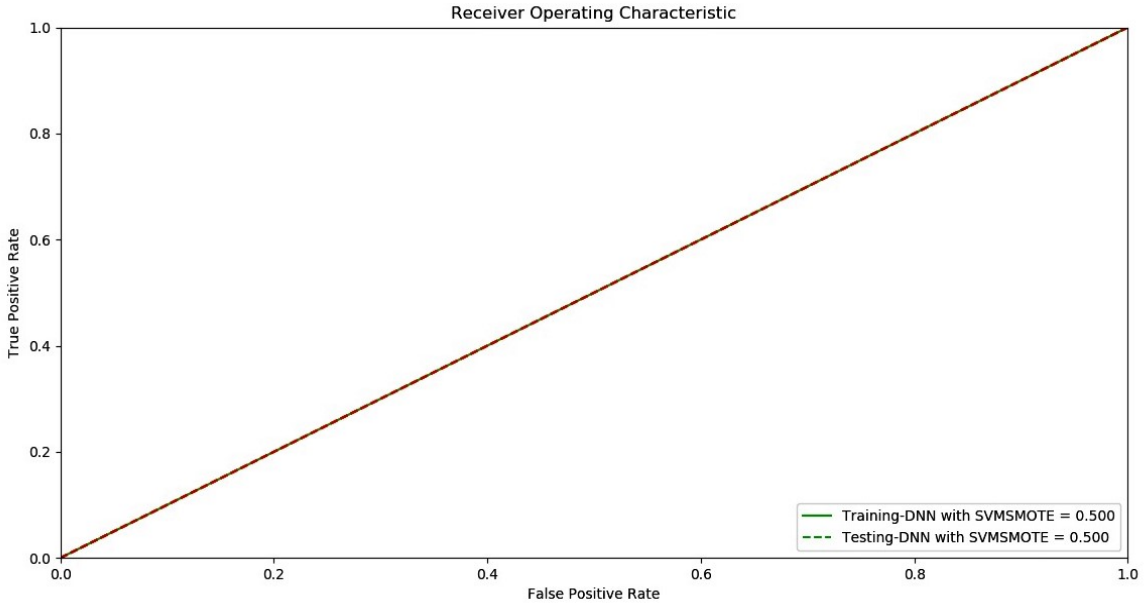


Figure 30.0 Receiver Operating Characteristics with 23 features - DNN with SVM SMOTE

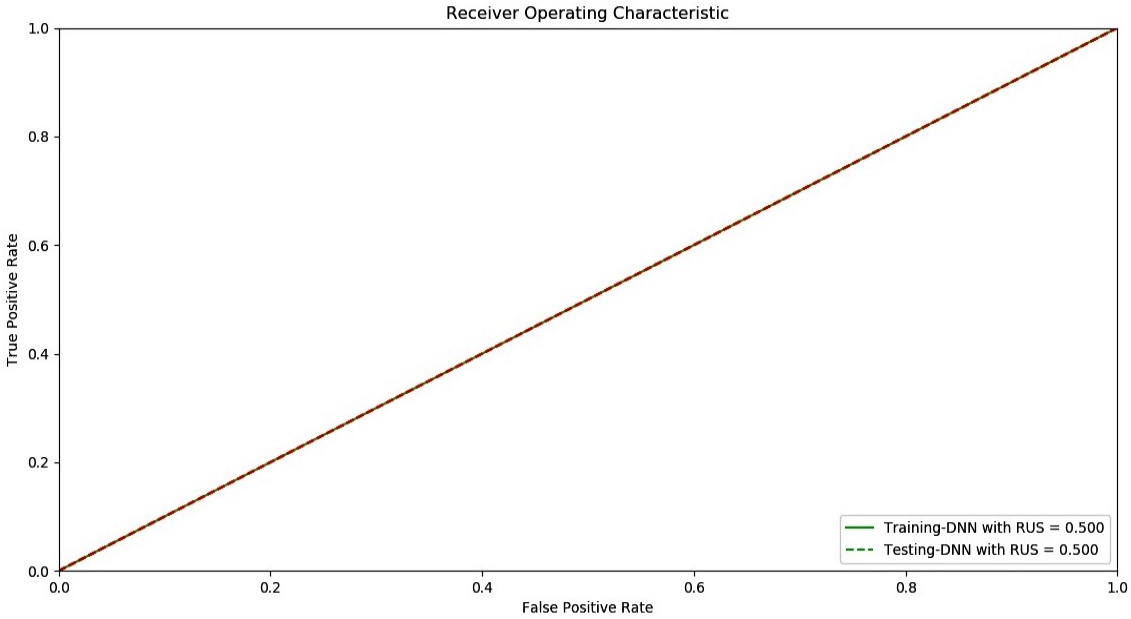


Figure 31.0 Receiver Operating Characteristics with 23 features - DNN with RUS

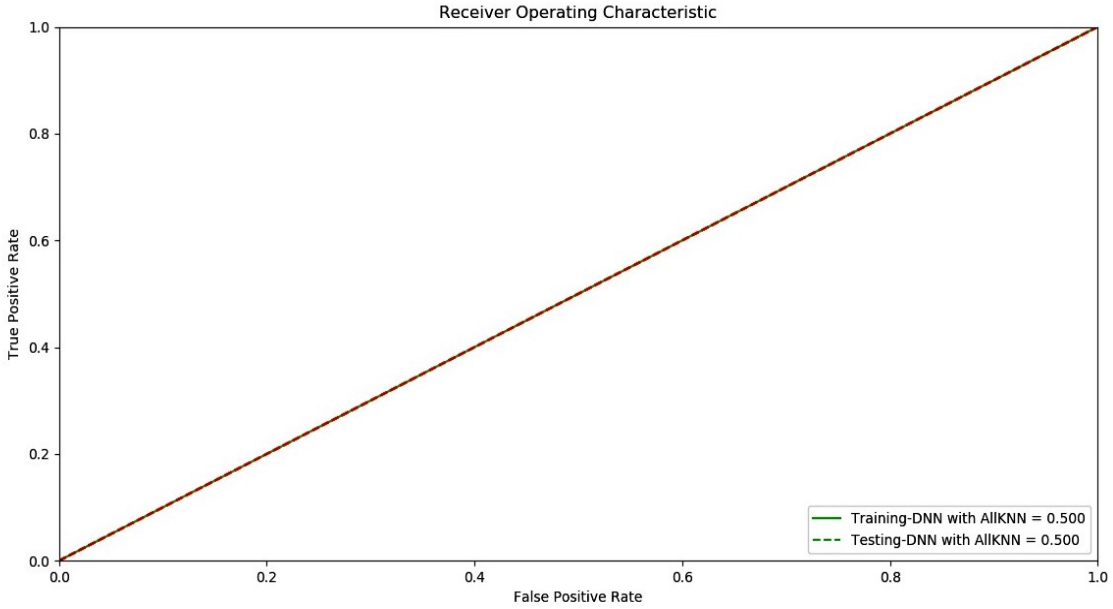


Figure 32.0 Receiver Operating Characteristics with 23 features - DNN with All-KNN

Table 32.0 outlays the performance metrics for the ANN Model with 23 features. As it can be observed, we have received consistent accuracy of 78.38% mainly due to true negatives, with sensitivity at 1.000 and balanced accuracy at 50%. ROC score for all the techniques has been flat 0.50 which indicates that with 24 features ANN Model is a useless classifier and cannot be used for further applications. As mentioned before both ANN and Deep Neural model has shown similar characteristics with respect to the introduction of features, indicating

that these models will require further study on their behaviour towards activation and loss functions. The major reason behind both ANN and DNN gives out similar results is that the error rate for both these models converges to the same values using 23 features.

Table 32.0 Performance Metrics with 23 features - ANN

Model = ANN					
	SMOTE	**SVMSMOTE**	**RUS**	**AllKNN**	**Average**
Accuracy	0.7838	0.7838	0.7838	0.7838	0.7838
Specificity	1.0000	1.0000	1.0000	1.0000	1.0000
Sensitivity	0.0000	0.0000	0.0000	0.0000	0.0000
Balanced Accuracy	0.5000	0.5000	0.5000	0.5000	0.5000
Geometric Mean	0.0000	0.0000	0.0000	0.0000	0.0000
Precision	0.0000	0.0000	0.0000	0.0000	0.0000
Recall	0.0000	0.0000	0.0000	0.0000	0.0000
F1	0.0000	0.0000	0.0000	0.0000	0.0000
Area Under the ROC Curve					
Training	0.500	0.500	0.500	0.500	0.500
Testing	0.500	0.500	0.500	0.500	0.500

Following figures show the ROC Curve for each of the techniques under the ANN Model with 23 features

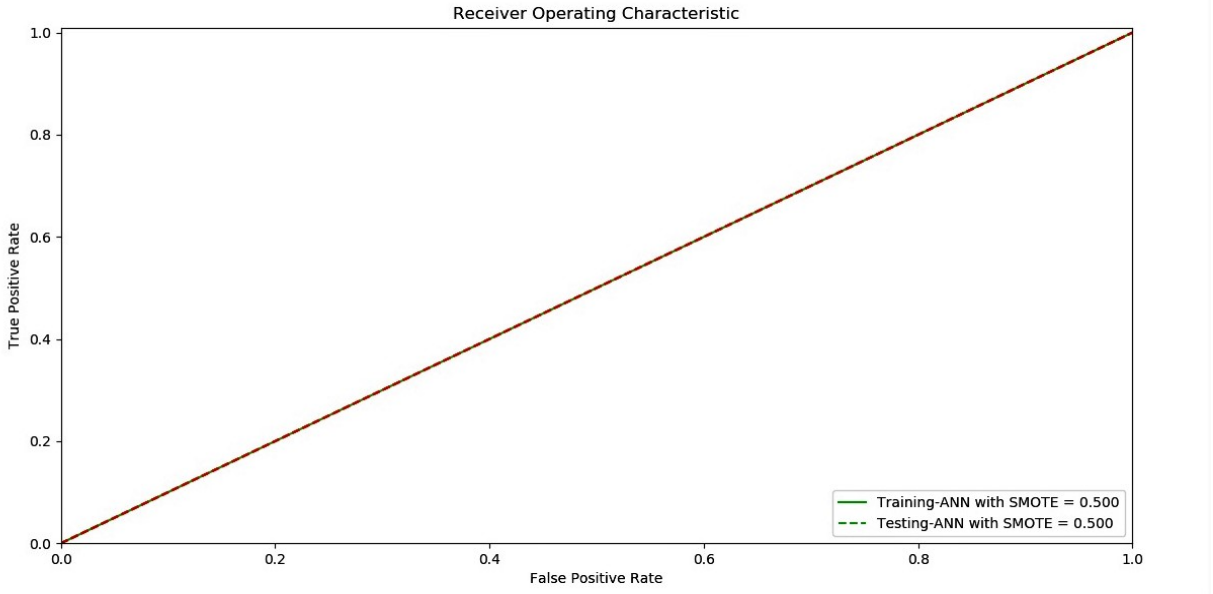


Figure 33.0 Receiver Operating Characteristics with 23 features - ANN with SMOTE

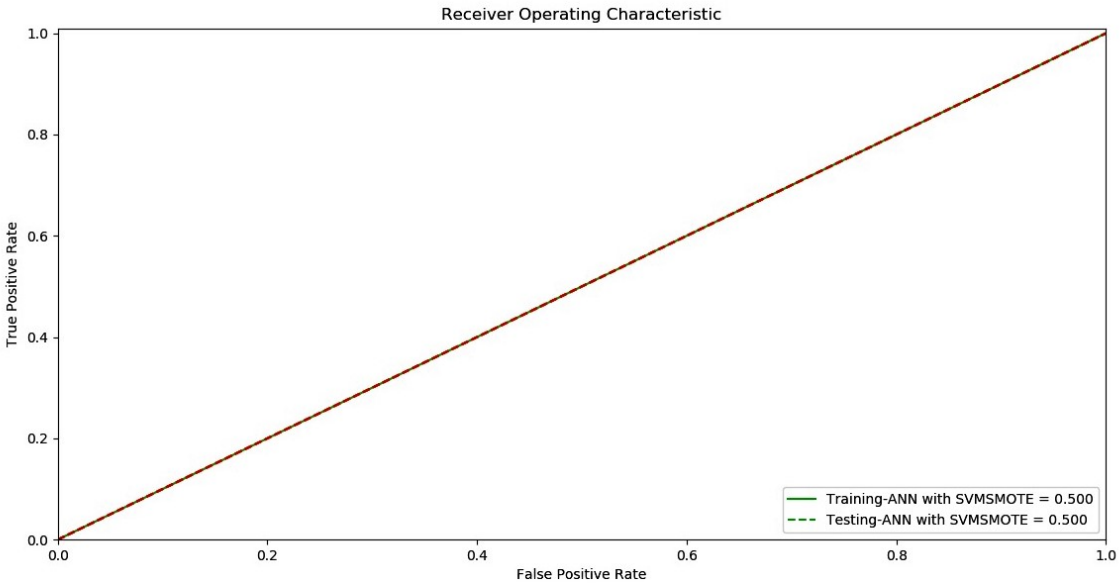


Figure 34.0 Receiver Operating Characteristics with 23 features - ANN with SVM SMOTE

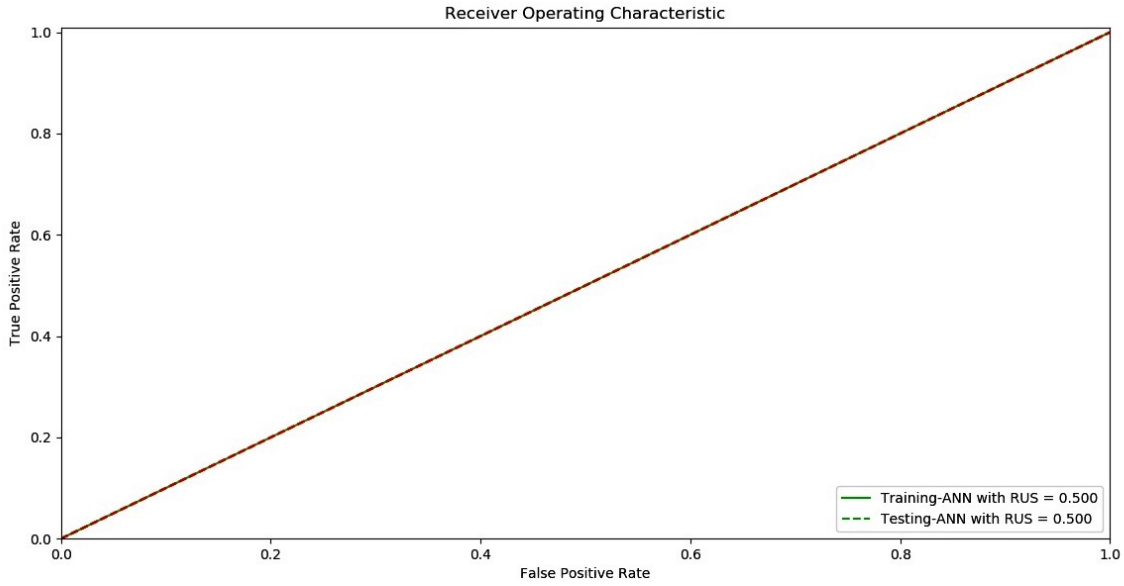


Figure 35.0 Receiver Operating Characteristics with 23 features - ANN with RUS

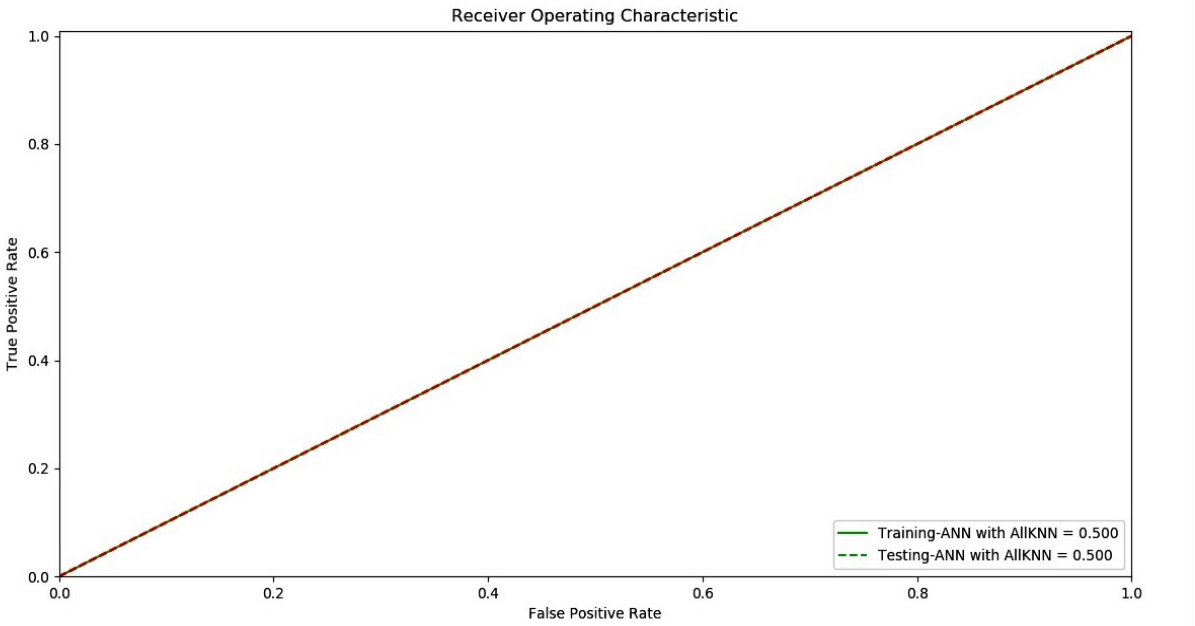


Figure 36.0 Receiver Operating Characteristics with 23 features - ANN with All-KNN

Table 33.0 outlays the performance metrics for the RBF Kernel-based Support Vector Machine with 23 features. As it can be observed, under this model All-KNN has achieved more than 78.42% accuracy whereas SMOTE and RUS have achieved closer to 78.1% accuracy. Based on Balanced Accuracy and G-Mean, RUS has performed much better than other techniques with this model. ROC score indicates that the performance of the classifier in terms of such an imbalanced dataset. A huge difference between the training and the testing ROC score indicates that the model is overfitting due to the use of sampling techniques. In this case, as we can observe the training ROC for most of the techniques except All-KNN are closing in at 0.99 and training score at 0.51, the models were overfitted. All-KNN technique is the only exception with a closer gap between the training and the testing ROC score but the score with this technique is closer to 0.50 as well indicating the model with the technique cannot be used for further application.

Table 33.0 Performance Metrics with 23 features - SVM - RBF Kernel

Model = SVM-RBF					
	SMOTE	**SVMSMOTE**	**RUS**	**AllKNN**	**Average**
Accuracy	0.7813	0.7820	0.7807	0.7842	0.7821
Specificity	0.9858	0.9875	0.9817	0.9983	0.9883
Sensitivity	0.0401	0.0370	0.0517	0.0077	0.0341
Balanced Accuracy	0.5130	0.5123	0.5167	0.5030	0.5112
Geometric Mean	0.1988	0.1911	0.2253	0.0877	0.1757
Precision	0.4370	0.4486	0.4379	0.5556	0.4698
Recall	0.0401	0.0370	0.0517	0.0077	0.0341
F1	0.0734	0.0684	0.0924	0.0152	0.0624
Area Under the ROC Curve					
Training	0.994	0.994	0.992	0.527	0.877
Testing	0.513	0.512	0.517	0.503	0.511

Following figures show the ROC Curve for each of the techniques under the SVM - RBF Kernel with 23 features

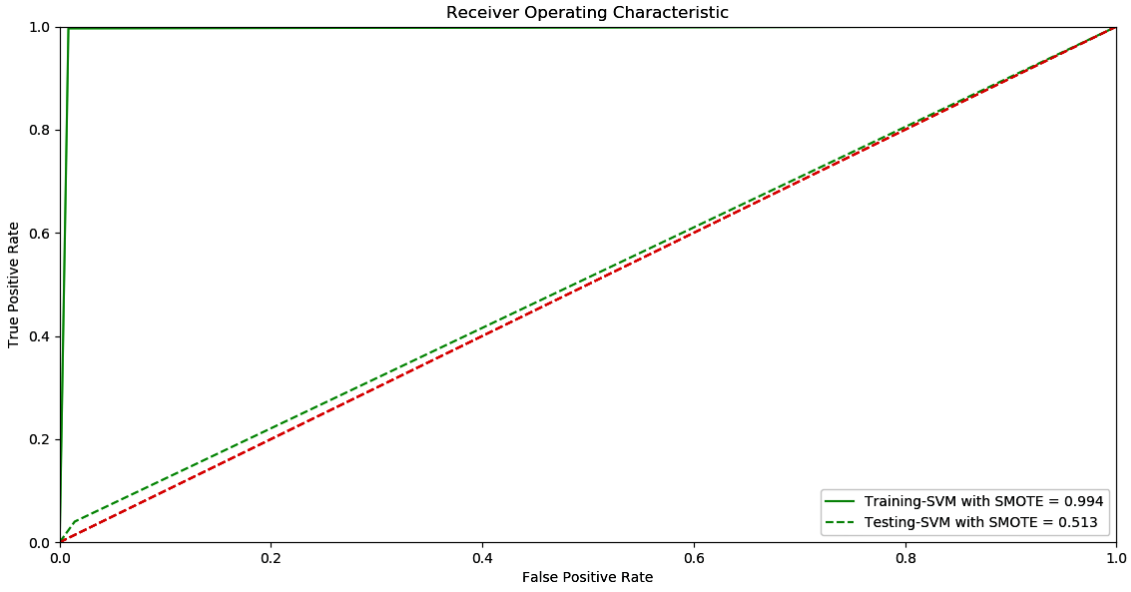


Figure 37.0 Receiver Operating Characteristics with 23 features - Support Vector Machine - RBF Kernel with SMOTE

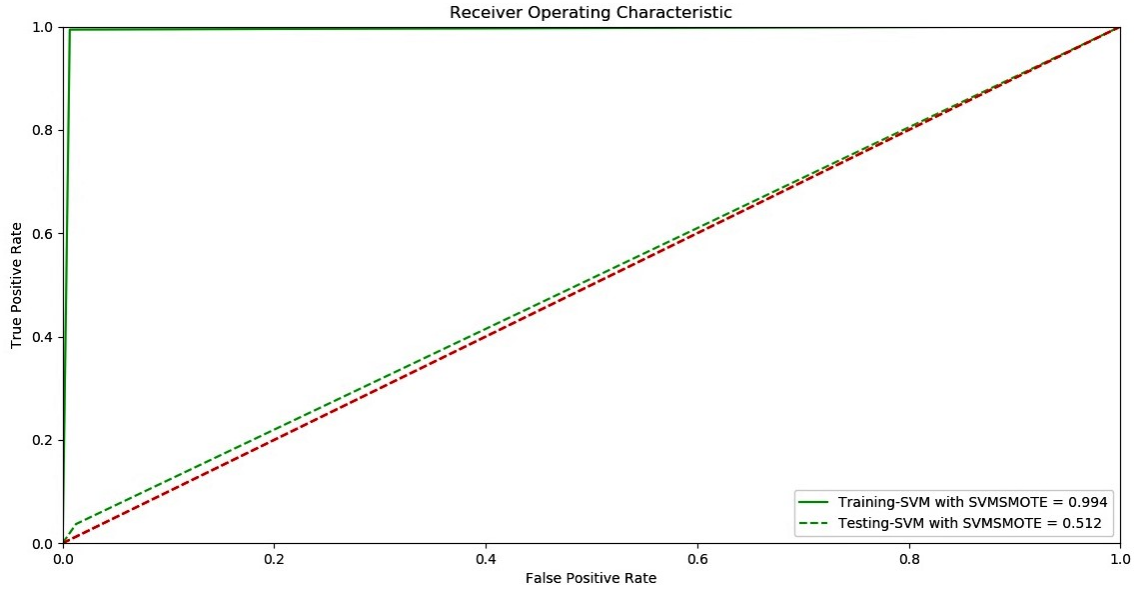


Figure 38.0 Receiver Operating Characteristics with 23 features - Support Vector Machine - RBF Kernel with SVM SMOTE

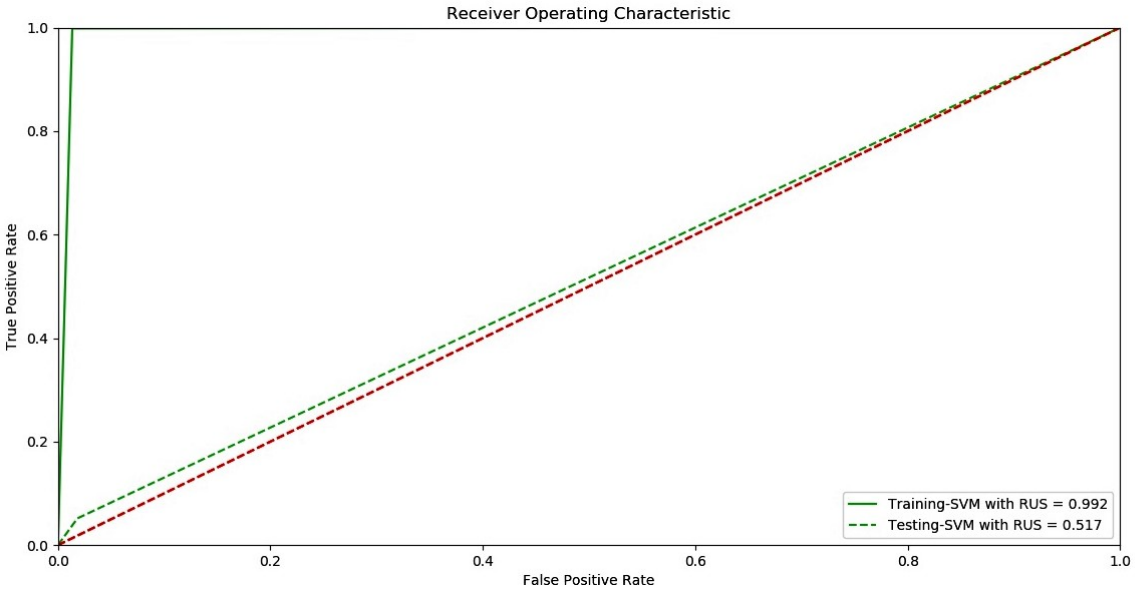


Figure 39.0 Receiver Operating Characteristics with 23 features - Support Vector Machine - RBF Kernel with RUS

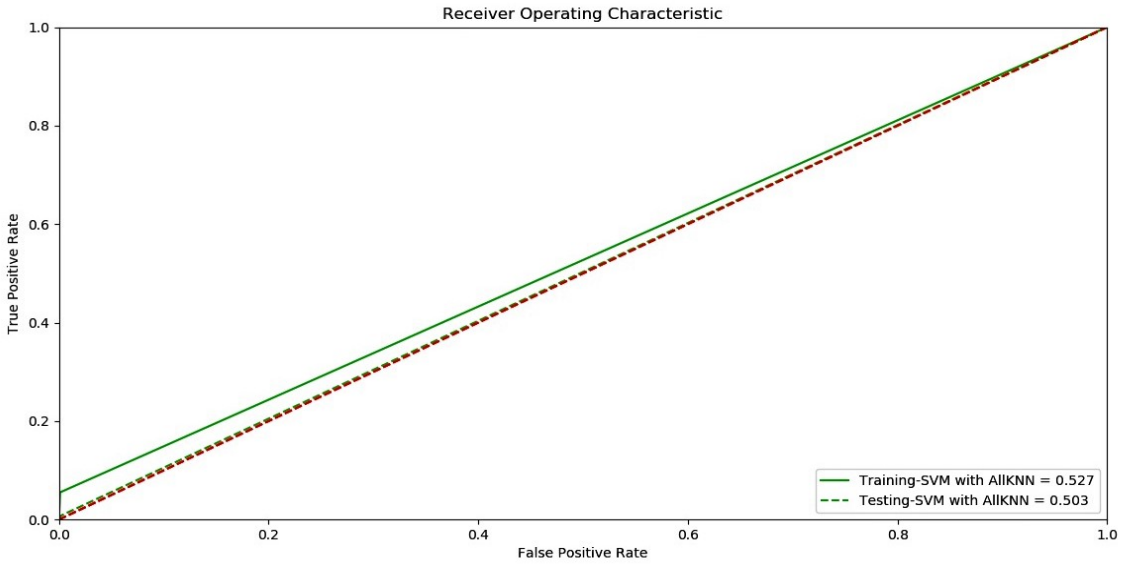


Figure 40.0 Receiver Operating Characteristics with 23 features - Support Vector Machine - RBF Kernel with All-KNN

Table 34.0 outlays the performance metrics for the KNN with 23 features. As it can be observed, under this model All-KNN has achieved more than 78.30% accuracy whereas SMOTE and RUS have achieved closer to 62% accuracy. Based on Balanced Accuracy and G-Mean, RUS has performed much better than other techniques with this model. ROC score for the oversampling techniques with this model shown a smaller gap between the training and the testing score as compared to the SVM. Under-sampling techniques have shown a much lesser gap and have been able to achieve a nearer score in both training and testing ROC. Along with accuracy, balanced accuracy and G-Mean, KNN along with RUS has shown to be useful classifier as compared to other techniques.

Table 34.0 Performance Metrics with 23 features - KNN

Model = KNN					
	SMOTE	**SVMSMOTE**	**RUS**	**AllKNN**	**Average**
Accuracy	0.6230	0.6680	0.6283	0.7830	0.6756
Specificity	0.6568	0.7353	0.6494	0.9789	0.7551
Sensitivity	0.5004	0.4241	0.5520	0.0725	0.3873
Balanced Accuracy	0.5786	0.5797	0.6007	0.5257	0.5712
Geometric Mean	0.5733	0.5584	0.5987	0.2664	0.4992
Precision	0.2868	0.3064	0.3027	0.4870	0.3457
Recall	0.5004	0.4241	0.5520	0.0725	0.3872
F1	0.3646	0.3558	0.3910	0.1262	0.3094
Area Under the ROC Curve					
Training	0.750	0.727	0.653	0.528	0.665
Testing	0.579	0.58	0.601	0.526	0.572

Following figures show the ROC Curve for each of the techniques under the KNN with 23 features

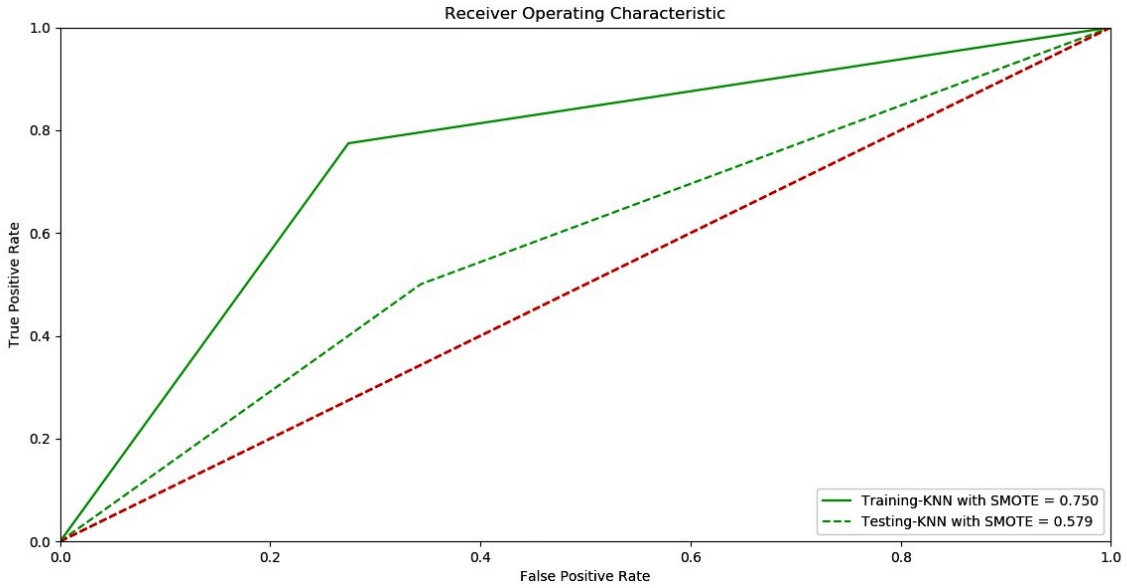


Figure 41.0 Receiver Operating Characteristics with 23 features - KNN with SMOTE

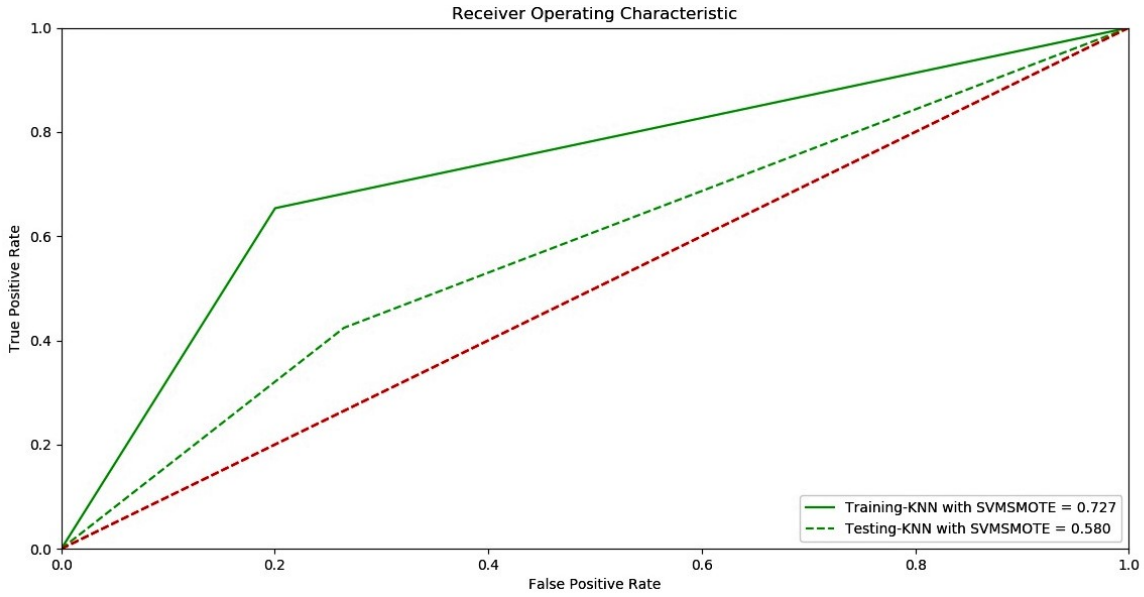


Figure 42.0 Receiver Operating Characteristics with 23 features - KNN with SVM SMOTE

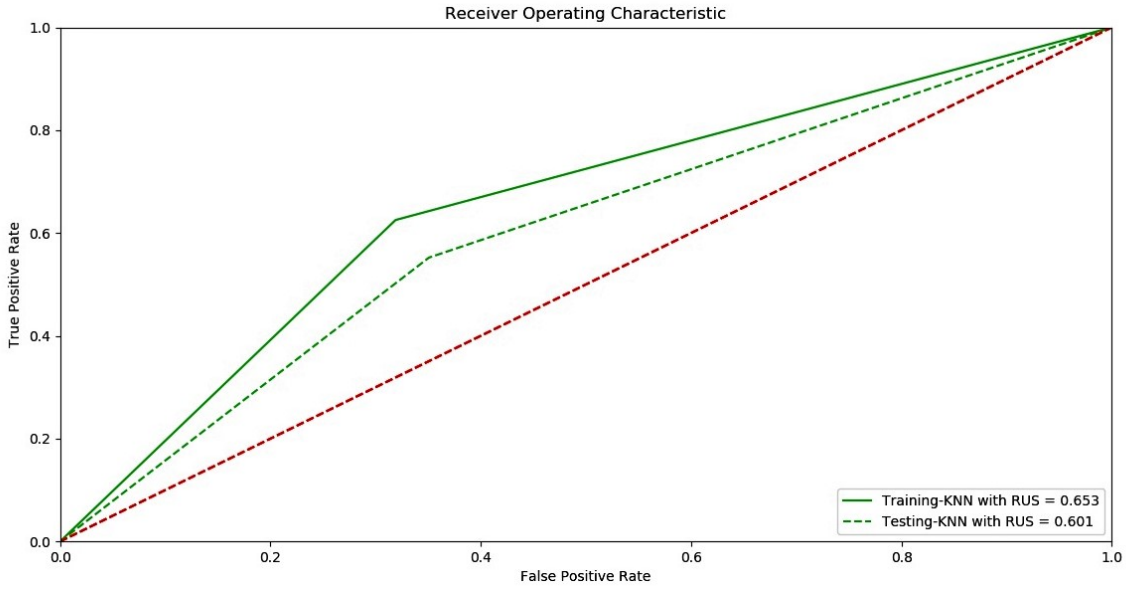


Figure 43.0 Receiver Operating Characteristics with 23 features - KNN with RUS

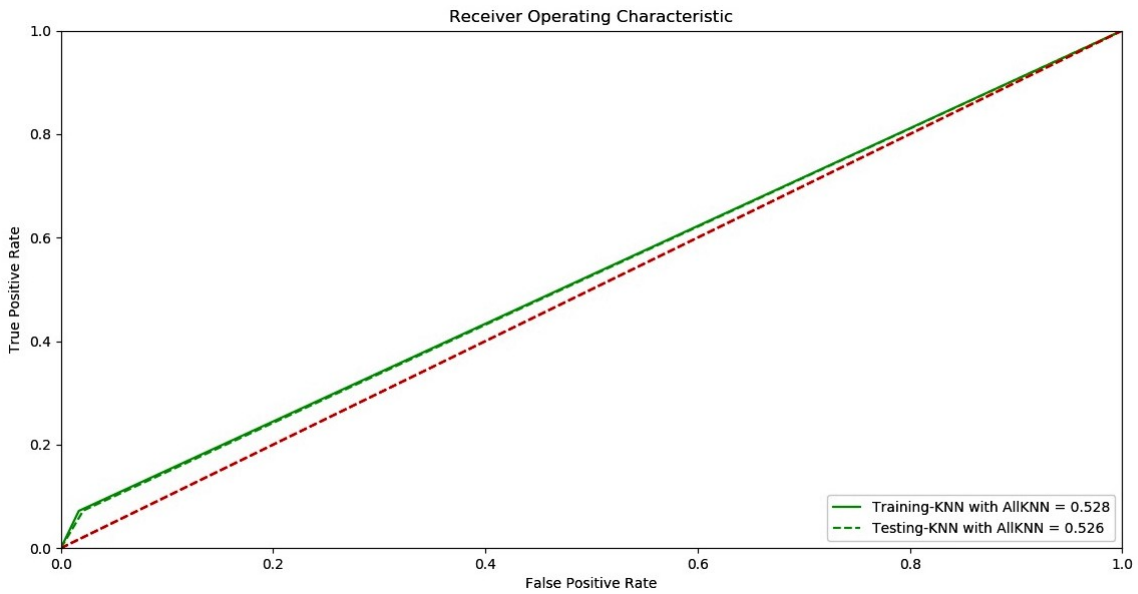


Figure 44.0 Receiver Operating Characteristics with 23 features - KNN with All-KNN

Table 35.0 Consolidated Accuracies of the Models - 23 features

Sampling	**SMOTE**	**SVMSMOTE**	**RUS**	**All KNN**	**Average**
DNN - Accuracy	0.7838	0.7838	0.7838	0.7838	0.7838
ANN - Accuracy	0.7838	0.7838	0.7838	0.7838	0.7838
SVM - Accuracy	0.7813	0.7820	0.7807	0.7842	0.7821
KNN - Accuracy	0.6230	0.6680	0.6283	0.7830	0.6756

Table 36.0 Consolidated Balanced Accuracies of the Models – 23 features

Sampling	**SMOTE**	**SVMSMOTE**	**RUS**	**All KNN**	**Average**
DNN - Accuracy	0.7838	0.7838	0.7838	0.7838	0.7838
ANN - Accuracy	0.7838	0.7838	0.7838	0.7838	0.7838
SVM - Accuracy	0.7813	0.7820	0.7807	0.7842	0.7821
KNN - Accuracy	0.6230	0.6680	0.6283	0.7830	0.6756

6.0 Implications and Conclusion

In this chapter, policies and implications for the use of machine learning and artificial intelligence in the banking sector and financial institutions have been discussed with a focus on the Canadian Banking sector. Section 6.1 discusses the policy implications and the development of a robust framework for unified implementation across the financial institutions in Canada. Section 6.3 outlines future work. Section 6.4 highlights the key contributions of the study. Section 6.5 concludes the study.

6.1 Policy Implications regarding the use of machine learning in Canada

Being one of the first national governments to establish a pan-national AI strategy, the Canadian government is at the forefront of bringing AI into applications than other national governments. The major aim of establishing the Pan-Canadian Artificial Intelligence Strategy was to increase the number of researches on AI and skilled graduates in the domain of AI and machine learning (CIFAR, 2019), to develop policies and thought leadership on economic, ethical and legal implications regarding the developments in the field of artificial intelligence (CIFAR, 2019). As a part of the efforts, recently CIFAR, the institution responsible for leading the strategy has increased the AI research Chair across Canada to 80 from 46 within the year 2019 itself.

The initiatives from the government have also been extended in terms of the establishment of the superclusters across Canada for implementing high-tech, AI-based applications for supporting different business functionalities. Out of these superclusters, the technology superclusters are located in the province of British Columbia for enhancing the applications of AI. Although the government has made headway in applying the knowledge of AI for a better business environment, but there have been limited responses from the private partners of the domain. It was not until late 2019, Canadian Banks has developed or implemented AI technologies in their systems in some or the other form, with Royal Bank of Canada (RBC) being the leader in the domain. RBC has also been a keen supporter of the government's CIFAR initiatives.

Bank of Canada's stance on machine learning and AI has been limited to research. Being the central bank, they could play a more developmental role in establishing a more robust framework for the application of artificial intelligence and machine learning in banking institutions. Through Partnerships in Innovation and technology program (PIVOT) Bank of Canada has been able to generate interest in developing innovative technologies for them but there have been limited applications of those technologies in actual business scenarios. As a central bank, it is of understanding they should consider that formulation of the future monetary policies may have a major impact due to the use of AI and machine learning in business.

Poloz (2019) in his discussion paper outlines how economies and monetary policies may drastically change in terms of implementation due to the fourth industrial revolution which calls for widespread application of machine learning and AI. Using Terms of Trade Economic Model, the author identified that real-time positive technology shock can lead to economic expansion and maintain downward pressure on the inflation targets. The technology in question has been the application of AI and machine learning in the economy. Poloz (2019) also discusses how the model has taken into account the major financial vulnerabilities faced by the central bank and the risk associated with macroeconomic factors.

The government of Canada and the Bank of Canada both are making headway towards the application of AI and machine learning in different parts of the economy. One would call for a more robust framework which can bring changes in the fundamental parts of the financial institutions like personal risk management for the credit instruments like the one we have used in this study. This would require collaborative actions from the banks operating in Canada along with the central bank being at the center stage of this framework implementation. To implement AI and machine learning models in such applications, the framework should also take into consideration the privacy and security of the datasets consisting of client information. The robust framework for creating such changes can be achieved through public-private partnerships and through a common understanding of the needs of the institutions participating in the framework.

6.2 Future Work

To identify the feature's importance Logistic Regression was used in the pre-processing stage and several of the features were discarded from further analysis. More robust feature

selection procedures can be implemented for the selection of features in conjunction with the DNN Model proposed in the study. It is of understanding that not all the discarded features may play an important role but feature selection can play a vital role in the output variable. The dataset used in the study had 30,000 different client information. To understand the complete working of the DNN Model proposed in the study, a larger dataset of the order of millions of records will help in further analyzing the model. A larger dataset can also help in understanding how fast the proposed model can help in getting the output as compared to the different models from the literature of credit risk assessment.

To realize the importance of DNN it is imperative that more similar studies will be required using different credit instruments like home mortgages, line of credits and vehicle loans. Comparative studies between two different datasets can also help in analyzing the model further.

6.3 Key Contributions

Some of the primary contributions of this study are as follows:

1. DNN Model proposed in the study has been able to achieve 81% accuracy with a ROC score of 0.70
2. Application of 4 different sampling techniques along with 4 different models for the study in credit risk assessment along with two sampling techniques to be used for the first time in credit risk assessment to the best of knowledge
3. Apart from SMOTE and RUS, All-KNN and SVM SMOTE are equally powerful sampling techniques under different models and scenarios as studied under this thesis

Some of the secondary contributions of this study are as follows:

1. Use of K-NN in the comparative study as through literature reviews it was identified that K-NN is the least studied model in credit risk assessment, although it is one of the base classifiers in the field of machine learning.
2. The proposition of a new framework for the widespread application and implementation of machine learning and artificial intelligence in the Canadian financial sector.

6.4 Practical Insights

Application of machine learning and DNNs can help the financial institutions in predicting the counterparty risk failure as we have seen in this study. Assuming the model is applied in the real-life scenario, the loss due to credit card delinquency can be reduced considerably. As per Mckinsey's Global Institute research on credit risk management (MGI, 2017), application of machine learning and advanced analytics can help financial institutions in three different ways. Firstly, by potential improvement in the revenue due to early detection of credit risk or counterparty risk. Secondly by saving potential money in cost reduction due to detection of potential fraud customers in the application of process of credit instruments such as credit cards. Thirdly by saving money which were previously employed in the risk mitigation strategies surrounding the credit risk management. At each of these stages' financial institutions, can save up to 10 to 15% of the potential value in revenue which in combination reduces the losses up to 30 to 35% by application of advanced analytical tools in credit risk management (Bahillo et al, 2016). Further application of advanced analytical models can help banks in improving their return on equity by approximately up to 4% (Harle, Havas & Samandari, 2015).

Canadian Bankers Association reported that over 600,000 credit cardholders were delinquents in 2018 (CBA,2018) with a net loss of approximately CAD $4.38 billion dollars as the net dollar value for credit cards transactions alone were at CAD $547.98 billion dollars. This dataset comprises for all the credit card issuing institutions in Canada. The delinquency rate for 2018 was at 0.8% (CBA,2018) which gives us the total loss value and total delinquent card holders. By the application of machine learning models, it can be brought down to CAD $2 to $3 billion dollars approximately if we apply the potential reduction percentages as stated by Bahillo et al (2016). This understanding and the application of the DNN based models can have profound impacts on the bottom line of the major financial institutions.

Considering a loss of CAD $4.38 billion dollars with 600,000 card holders, the average loss per card holder to the financial institutions can be approximated to be CAD $7300 dollars annually. Assuming the model applied in this study is applied to identity these 600,000 card holders in the earlier stage, at 82.18% accuracy 493,080 card holders will be classified as

delinquents. The savings would be approximately CAD $3.6 (493,080 x 7300) billion dollars to the financial institutions if these delinquent card holders are detected at the earlier stage.

Financial institutions like major banks and credit agencies can combine the application of models and computing powers to develop algorithms that can detect credit card delinquency with better accuracy. Being at the expense of the personal and more accurate information of the clients can also provide these institutions to accurately choose the required features to detect the default payments. Application of DNN models can only provide the required results if provided with the appropriate features to predict the dependent feature, in this study, it was the default payment for the next month. The choice of features creates a profound impact on the application of the DNN models as features with the least significant importance can result in noise and an increase in the error rates where as significant features can increase the accuracy rates as we have observed in this study.

6.5 Conclusion

One of the primary capabilities of a robust risk management system must be detecting the risks earlier, though many of the bank systems today lack this key capability which leads to further losses (MGI, 2017). This thesis was able to contribute to this gap by proposing a DNN model to be used along with sampling techniques for imbalanced datasets. The proposed model was able to achieve 82.18% accuracy with the use of the RUS sampling technique and a ROC score of 0.706. As a direct comparison with the models used by Hamori et al (2018) since they used the same dataset, our models and techniques have much better accuracy as they were only able to achieve 69.17% average accuracy in testing. Comparing with other models used in the literature, since many of them lacked the use of sampling techniques in one way or the other, this study could not place a direct comparison. Being said that at 82.18% accuracy and 0.706 ROC score, the DNN model proposed in this study can be concluded to be used as a real-life classifier in predicting credit risk assessment. Further, the application of such techniques and models will require the construction of a robust framework through a public-private partnership in the Canadian financial sector.

References

2019 Global payments trends report – Canada Country Insights. (2019). Retrieved from https://www.jpmorgan.com/merchant-services/insights/reports/Canada

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., … Zheng, X. (2016). TensorFlow: A system for large-scale machine learning. Retrieved from https://search.ebscohost.com/login.aspx?direct=true&db=edsarx&AN=edsarx.1605.08695&site=eds-live

Abdelmoula, A. K. (2015). Bank credit risk analysis with k-nearest neighbor classifier: Case of Tunisian banks. Accounting & Management Information Systems / Contabilitate Si Informatica de Gestiune, 14(1), 79–106. Retrieved from https://search-ebscohost-com.ezproxy.tru.ca/login.aspx?direct=true&db=bth&AN=102300233&site=eds-live

An Experiment with the Edited Nearest-Neighbor Rule. (1976). IEEE Transactions on Systems, Man, and Cybernetics, Systems, Man and Cybernetics, IEEE Transactions on, IEEE Trans. Syst., Man, Cybern, SMC-6(6), 448–452. https://doi-org.ezproxy.tru.ca/10.1109/TSMC.1976.4309523

Basel I: International Convergence of Capital Measurement and Capital Standards (1988, July 15). Retrieved from https://www.bis.org/publ/bcbs04a.htm

Bahillo, J. A., Ganguly, S., Kremer, A., & Kristensen, I. (2016). The value in digitally transforming credit risk management. Retrieved from https://www.mckinsey.com/business-functions/risk/our-insights/the-value-in-digitally-transforming-credit-risk-management

Basel I: International Convergence of Capital Measurement and Capital Standards (1988, July 15). Retrieved from https://www.bis.org/publ/bcbs04a.htm

Basel II: International Convergence of Capital Measurement and Capital Standards: a Revised Framework. (2004, June 10). Retrieved from https://www.bis.org/publ/bcbs107.htm

Basel III: A global regulatory framework for more resilient banks and banking systems - revised version June 2011. (2011, June 1). Retrieved from https://www.bis.org/publ/bcbs189.htm

Canadian demands for speed and convenience influencing payments innovation. (2018, December 12). Retrieved from https://www.payments.ca/industry-info/our-research/canadian-demands-speed-and-convenience-influencing-payments-innovation

Canadians rapidly adopting new payments channels. (2019, December 5). Retrieved from https://www.payments.ca/industry-info/our-research/canadians-rapidly-adopting-new-payments-channels

CBA - Credit card statistics. (2019, July 18). Retrieved from https://cba.ca/credit-card-statistics

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2011). SMOTE: Synthetic Minority Over-sampling Technique. https://doi-org.ezproxy.tru.ca/10.1613/jair.953

CIFAR - Pan-Canadian Artificial Intelligence Strategy. (2019). Retrieved from https://www.cifar.ca/ai/pan-canadian-artificial-intelligence-strategy

Cover, T., & Hart, P. (1967, January). Nearest neighbor pattern classification. IEEE Transactions on Information Theory, 13(1), 21–27. doi: 10.1109/TIT.1967.1053964

BAYRACI, S., & SUSUZ, O. (2019). A Deep Neural Network (DNN) based classification model in application to loan default prediction. Theoretical & Applied Economics, (4), 75–84. Retrieved from https://search-ebscohost-com.ezproxy.tru.ca/login.aspx?direct=true&db=bth&AN=140243898&site=eds-live

Cao, J., Lu, H., Wang, W., & Wang, J. (2013). A loan default discrimination model using cost-sensitive support vector machine improved by PSO. Information Technology & Management, 14(3), 193–204. https://doi-org.ezproxy.tru.ca/10.1007/s10799-013-0161-1

Chen, S., Härdle, W. K., & Moro, R. A. (2011). Modeling default risk with support vector machines. Quantitative Finance, 11(1), 135-154. doi: 10.1080/14697680903410015

Cimpoeru, S. S. (2011). Neural Networks and Their Application in Credit Risk Assessment. Evidence from the Romanian Market. Technological & Economic Development of Economy, 17(3), 519–534. https://doi-org.ezproxy.tru.ca/10.3846/20294913.2011.606339

Danenas, P., & Garsva, G. (2015). Selection of Support Vector Machines based classifiers for credit risk domain. Expert Systems With Applications, 42(6), 3194–3204. https://doi-org.ezproxy.tru.ca/10.1016/j.eswa.2014.12.001

Equifax history (2018). Retrieved from https://www.equifax.co.uk/resources/what_we_do/credit-experts-since-1899.html

Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R., & Lin, C.-J. (2008). LIBLINEAR: A Library for Large Linear Classification. JOURNAL OF MACHINE LEARNING RESEARCH, 9, 1871–1874. Retrieved from https://search-ebscohost-com.ezproxy.tru.ca/login.aspx?direct=true&db=edswsc&AN=000262636800009&site=eds-live

Fix. E., & Hodges, Jr., J. L. (1951, February). Discriminatory analysis, nonparametric discrimination. Retrieved from https://apps.dtic.mil/dtic/tr/fulltext/u2/a800276.pdf

Harris, T. (2015). Credit scoring using the clustered support vector machine. Expert Systems with Applications, 42(2), 741–750. https://doi-org.ezproxy.tru.ca/10.1016/j.eswa.2014.08.029

Härle, P., Havas, A., & Samandari, H. (2015). The future of bank risk management. Retrieved from https://www.mckinsey.com/business-functions/risk/our-insights/the-future-of-bank-risk-management

Haykin, S. S. (1998). Neural networks: a comprehensive foundation. Upper Saddle River, NJ: Prentice-Hall, c1998.

Hien M. Nguyen, Eric W. Cooper, and Katsuari Kamei. 2011. Borderline over-sampling for imbalanced data classification. Int. J. Knowl. Eng. Soft Data Paradigm. 3, 1 (April 2011), 4–21. DOI:https://doi.org/10.1504/IJKESDP.2011.039875

KARAA, A., & KRICHENE, A. (2012). Credit-Risk Assessment Using Support Vectors Machine and Multilayer Neural Network Models: A Comparative Study Case of a Tunisian Bank. Accounting & Management Information Systems / Contabilitate Si Informatica de Gestiune, 11(4), 587–620.

Kasabov, N. K. (1996). Foundations of Neural Networks, Fuzzy Systems, and Knowledge Engineering. Cambridge, Mass: MIT Press. Retrieved from https://search-ebscohost-com.ezproxy.tru.ca/login.aspx?direct=true&db=nlebk&AN=1810&site=eds-live

Khashman, A. (2010). Neural networks for credit risk evaluation: Investigation of different neural models and learning schemes. Expert Systems with Applications. https://doi.org/10.1016/j.eswa.2010.02.101

Khemakhem, S., & Boujelbènea, Y. (2015). Credit risk prediction: A comparative study between discriminant analysis and the neural network approach. Accounting & Management Information Systems / Contabilitate Si Informatica de Gestiune, 14(1), 60–78

Kim, A., Yang, Y., Lessmann, S., Ma, T., Sung, M.-C., & Johnson, J. E. V. (2020). Can Deep Learning Predict Risky Retail Investors? A Case Study in Financial Risk Behavior Forecasting. European Journal of Operational Research, 283(1), 217–234

Kvamme, H., Sellereite, N., Aas, K., & Sjursen, S. (2018). Predicting mortgage default using convolutional neural networks. Expert Systems With Applications, 102, 207–217. https://doi-org.ezproxy.tru.ca/10.1016/j.eswa.2018.02.029

Massaron, L., & Boschetti, A. (2016). Regression Analysis with Python. Packt Publishing

McFadden, D. (1974). Conditional logit analysis of qualitative choice behavior. In P. Zarembka (Ed.), Frontiers in econometrics (pp. 104-142). New York, NY: Academic Press

Murphy, K. P. (2012). Machine learning: A Probabilistic Perspective. Cambridge, MA: MIT Press

Oreski, S., Oreski, D., & Oreski, G. (2012). Hybrid system with genetic algorithm and artificial neural networks and its application to retail credit risk assessment. Expert Systems With Applications,39(16),12605–12617. https://doi-org.ezproxy.tru.ca/10.1016/j.eswa.2012.05.023

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., … Duchesnay, É. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12(10), 2825–2830. Retrieved from https://search-ebscohost-com.ezproxy.tru.ca/login.aspx?direct=true&db=bth&AN=70109929&site=eds-live

Poloz, S. (2019, November 14). Technological Progress and Monetary Policy: Managing the Fourth Industrial Revolution. Retrieved from https://www.bankofcanada.ca/2019/11/staff-discussion-paper-2019-11

Reed, R. D., & Marks, R. J. (1999). Neural Smithing: Supervised Learning in Feedforward Artificial Neural Networks. Cambridge, Mass: A Bradford Book. Retrieved from https://search-ebscohost-com.ezproxy.tru.ca/login.aspx?direct=true&db=nlebk&AN=9366&site=eds-live

Schmidhuber, J. (2014). Deep Learning in Neural Networks: An Overview. https://doi-org.ezproxy.tru.ca/10.1016/j.neunet.2014.09.003

Shigeyuki Hamori, Minami Kawai, Takahiro Kume, Yuji Murakami, & ChikaraWatanabe. (2018). Ensemble Learning or Deep Learning? Application to Default Risk Analysis. Journal of Risk & Financial Management, 11(1), 1. https://doi-org.ezproxy.tru.ca/10.3390/jrfm11010012

Sun, T., & Vasarhelyi, M. A. (2018). Predicting credit card delinquencies: An application of deep neural networks. Intelligent Systems in Accounting, Finance & Management, 25(4), 174–189. https://doi-org.ezproxy.tru.ca/10.1002/isaf.1437

Vapnik, V.N. (2000). The nature of statistical learning theory (2nd ed). New York: Springer

W. E. Henley, & D. J. Hand. (1996). A k-Nearest-Neighbour Classifier for Assessing Consumer Credit Risk. Journal of the Royal Statistical Society. Series D (The Statistician), 45(1), 77. https://doi-org.ezproxy.tru.ca/10.2307/2348414

Yannis Marinakis, Magdalene Marinaki, Michael Doumpos, Nikolaos Matsatsinis, & Constantin Zopounidis. (2008). Optimization of nearest neighbor classifiers via metaheuristic algorithms for credit risk assessment. Journal of Global Optimization, 42(2), 279-293. Retrieved from https://search-ebscohost-com.ezproxy.tru.ca/login.aspx?direct=true&db=edb&AN=34205028&site=eds-live

Zhu, B., Yang, W., Wang, H., & Yuan, Y. (2018). A hybrid deep learning model for consumer credit scoring. 2018 International Conference on Artificial Intelligence and Big Data (ICAIBD), Artificial Intelligence and Big Data (ICAIBD), 2018 International Conference On, 205–208. https://doi-org.ezproxy.tru.ca/10.1109/ICAIBD.2018.8396195