East West University

B.Sc. ENGINEERING THESIS

"Credit Card Fraud and Bankruptcy Detection Using Machine Learning"

Authors:

Habiba Islam Juma

2017-3-55-014

Rakibur Rahman Raja

2017-2-55-028

Md Rubayat Zaman

2018-1-55-010

Supervisor:

Rizwan Shaikh

Lecturer, Dept of ECE

East West University

This thesis has been submitted in fulfillment of the requirements for the degree of Bachelor of Science in Electronics and Communications Engineering

September 28, 2022

Letter of Acceptance

This thesis paper titled “Credit Card Fraud and Bankruptcy Detection Using Machine Learning" was submitted by Habiba Islam Juma (ID: 2017-3-55-014), Rakibur Rahman Raja (ID: 2017-2-55-028), and Md Rubayat Zaman (ID: 2018-1-55-010) of Electronics and Communications Engineering, East West University, Dhaka-1212, Bangladesh, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Electronics and Communications Engineering and approved as to its style and contents.

_____

Mohammad Arifuzzaman Ph.D.
Chairperson & Associate Professor
Department of Electronics and Communications Engineering

East West University

_____

Rizwan Shaikh

Lecturer
Department of Electronics and Communications Engineering

East West University

Declaration of Authorship

We hereby, certify that the work presented in our thesis, "Credit Card Fraud and Bankruptcy Detection Using Machine Learning," completed in partial fulfillment for the granting of a Bachelor of Science in Electronics and Communications Engineering is the result of our original research performed and authentically prepared by us under the direct supervision of Rizwan Shaikh, Lecturer, Department of Electronics and Communications Engineering, East West University. This thesis, or any substantial portion thereof, has not been previously submitted for publication anywhere. When we've used information or text from another paper, report, or book, we've made sure to properly attribute and credit the origin.

Signature,

_____

Date: September 2022

Habiba Islam Juma
ID: 207-3-55-014
Dept of ECE
East West University

Signature,

_____

Date: September 2022

Rakibur Rahman Raja
ID: 2017-2-55-028
Dept of ECE
East West University

Signature,

_____

Date: September 2022

Md Rubayat Zaman
ID: 208-1-55-010
Dept of ECE
East West University

Supervisor Signature,

_____

Date: September 2022

Rizwan Shaikh
Lecturer
Dept of ECE
East West University

Acknowledgment

First and first, we want to express our gratitude to Allah, the Almighty, the Lord and Sustainer of the universe, for His generosity, compassion, and blessings, all of which were necessary to the successful completion of this thesis.

Our profound appreciation goes out to Rizwan Shaikh, Lecturer in the Electronics and Communications Engineering Department, who gave us the chance to complete our thesis and who has been an excellent supervisor throughout. His enthusiasm, honesty, insight, and drive have been very motivating to us. He instructed us on the best way to go on and display the finished product. Getting to work and learn under his tutelage was a fantastic opportunity for us.

We'd also want to say thanks to everyone at East West University, including the ECE department's professors and the EWU Library.

The whole Electronics and Communications Engineering faculty and staff at East West University deserves our sincere appreciation for all of their help and support during our time there. In light of this, we would like to express our appreciation to everyone who helped us throughout the course of this study.

We would want to end by expressing our deep appreciation to our parents for everything they have done for us in terms of training and preparation for the future, including their dedication, prayers, care, and sacrifices.

Abstract

The study details the implementation of two separate machine learning projects referred to as "Credit Card Fraud Detection" and "Bankruptcy Detection," each of which makes use of four unique machine learning models. The research was carried out with the assistance of the K-Nearest Neighbor, Decision Tree, Support Vector Machine, and Logistic Regression models. When compiling the supervised datasets that we used, we used information from two reliable sources. We have under-sampled the data to build a more representative sample, and we have employed the feature selection approach to increase the accuracy with which these models anticipate our main goal. In the end, we assessed the results that each model produced against one another and chose the one that had the greatest overall performance. In addition to this, we have provided suggestions on how future models and data collecting may be improved.

Keywords: Machine Learning, Credit Card Fraud, Bankruptcy, K-Nearest Neighbor, Decision Tree, Support Vector Machine, Logistic Regression, Prediction, Accuracy.

TABLE OF CONTENTS

Contents

Title 1
Letter of Acceptance 2
Declaration of Authorship 3
Acknowledgement 4
Abstract 5
Table of contents 6
List of Chapters 7-8
List of Figures 9-10
List of Tables 10
List of Abbreviations 11

LIST OF CHAPTERS

1. INTRODUCTION

1. INTRODUCTION 7

1.1 Background and motivation 12

1.2 Related Works 15

1.3 Thesis Objectives 16

1.3.1 Thesis Outlines 16

2. MACHINE LEARNING TECHNIQUES AND WORKING PRINCIPLE

2.1 Machine Learning (ML) 17

2.2 Machine Learning Methods 19

2.3 Machine Learning Working Principle 23

2.4 ML Programming languages and tools 27

2.5 Building a Machine Learning Model 29

2.6 Machine Learning Models & Algorithms: Theoretical Discussion 33

3. MATERIALS AND METHODOLOGY

3.1 Credit Card Fraud Detection 39

3.1.1 Dataset 39

3.1.2 Methodology 41

3.2 Bankruptcy Detection 44

3.2.1 Dataset 45

3.2.2 Methodology 46

4. RESULT AND DISCUSSION

4.1 Results Discussion: Credit Card Fraud Detection 49

4.1.1 Models Evaluation: Credit Card Fraud Detection 52

4.1.2 Comparing the models: Credit Card Fraud Detection 55

4.2 Results Discussion: Bankruptcy Detection 58

4.2.1 Model Evaluation: Bankruptcy Detection 60

5. CONCLUSION AND FUTURE WORKS

5.1 Conclusion 62

5.2 Future Works 63

6. REFERENCE

References 64

LIST OF FIGURES

1.1: Machine Learning Work Process 12
2.1: Example of Supervised learning 19
2.2: Example of Unsupervised Learning 20
2.3: Example of Reinforcement Learning. 21
2.4: Example of Semi-Supervised Learning 22
2.5: Machine Learning techniques. 23
2.6: Supervised Learning technique 24
2.7: Unsupervised Learning technique. 25
2.8 Dataset Collection 29
2.9: Preparing dataset by cleaning and visualizing. 30
2.10: Choosing a model 30
2.11: Training the Model 31
2.12: Evaluating the model 32
2.13: Hamming distance formula. 33
2.14: Decision Tree Model structure 34
2.15: SVM working principle. 35
2.16: Linear SVM & Hyper-plane SVM. 36
2.17: Logistic Function 37
2.18: Logistic Regression Equation 37
3.1: Credit Card Dataset first 5 rows. 39
3.2: Shows a pie chart of the dataset's class distribution. 39
3.3: Class count of Normal (0) & Fraudulent (1) Transactions 40
3.4: The histogram of variables from the dataset. 42
3.11: First 5 Rows of the Dataset of Bankruptcy Detection. 44
3.12: Unbalanced dataset and data distribution plot (Bankruptcy). 44
3.13: National-Asset Flag. 45
3.14: Liability-Assets Flag and Bankruptcy. 46
3.15: Net Income Flag and Bankruptcy 47
3.16: Correlation between important categories to find out the most relevant features. 39

4.1: Prediction results of the K-Nearest Neighbor Model. 49
4.2: Prediction results of Decision Tree Model 50
4.3: Prediction result of Support Vector Machine (SVM) Model 50
4.4: Prediction results of Logistic Regression Model. 51
4.5: Confusion Matrix (Credit Card Fraud Detection). 53
4.6: The ROC and AUC curve value of all Models. 54
4.7: Comparing models 55
4.8: Classification Report of KNN. 57
4.9: Classification Report of DT. 58
4.10: Classification Report of LR. 58
4.11: Confusion Matrix (Bankruptcy Detection). 59
4.12: Comparison of models (Bankrupt Detection). 60

LIST OF TABLES

Table 2.1: Top ML programming Language (Source: GitHub) 27
Table 4.1: Comparison of Models for a different distribution of samples. 60

LIST OF ABBREVIATIONS

ML	Machine Learning
KNN	K-Nearest Neighbor
SVM	Support Vector Machine
DT	Decision Tree
LR	Logistic Regression
RF	Random Forest
SMOTE	Synthetic Minority Oversampling Technique
MLP	Multiplayer Perception
NB	Naïve Byes
GA	Genetic Algorithm
AI	Artificial Intelligence
MLOps	Machine Learning Operations
NLP	Natural Language Processing
ASM	Altered State Machine
MLLib	Machine Learning Library
AUC	Area Under Curve
ROC	Receiver Operating Characteristic
TP	True Positive
TN	True Negative
FP	False Positive
FN	False Negative
TPR	True Positive Rates
FPR	False Positive Rates

CHAPTER 1
INTRODUCTION

1.1 Background and motivation

Machine learning is a field of study that enables machines to learn without being given specific instructions. Machine learning is one of the most cutting-edge fields in IT today. As the name suggests, the ability to learn is crucial in bringing computers up to human standards of intelligence. Some of the current applications of machine learning may surprise you. With the fast expansion in storage capacity and processing power of computers, the discipline of machine learning, which may be roughly described as allowing computers to generate effective predictions from prior experiences, has shown spectacular improvement in recent years. Machine learning techniques have found widespread use in bioinformatics, amongst many other fields. Due to the complexity and expense of biological analysis, researchers have developed cutting-edge machine learning methods specifically for this field [1]. Retail is only one of its many potential uses; in the financial sector, for example, banks utilize historical data for purposes such as underwriting loans, preventing fraud, identifying insolvency, and predicting stock market performance. Maximizing some performance metrics with the use of example data or previous experience is the goal of machine learning. A model is specified up to a certain point in terms of parameters, and learning is the process of running a computer program to adjust those parameters in light of experience or training data. The model might be either descriptive, for learning from past examples, or predictive, for looking into the future. We have highlighted two machine learning projects in our thesis paper. First of all, we will experiment with a credit card dataset that is highly unbalanced. Therefore, we will apply different ML algorithms to detect fraudulent transactions. In our thesis work, we have shown two different machine learning applications. To begin, we will test several approaches using a credit card dataset that is extremely imbalanced. As a result, we'll use several ML algorithms to identify potentially fraudulent financial dealings. Our focus will be squarely on the study's findings and methods. Second, with the help of a bank asset dataset, we'll apply the

same ML models we used for credit card fraud detection to foresee the likelihood that a bank would fail soon. The next step is to evaluate each model's output, draw conclusions, and name the top models.

Machine learning is a group of AI algorithms used in fraud detection that are taught using past data to provide recommendations about future risks. If rules are established to prevent or permit certain user behaviors (such as suspect logins, identity theft, or fraudulent transactions), they may be put into place. To reduce the number of false positives and increase the accuracy of the risk criteria, it is essential to label past instances as either fraudulent or nonfraudulent before training the machine learning engine. The more time the algorithms are given to work, the more precise the suggested rules will be.

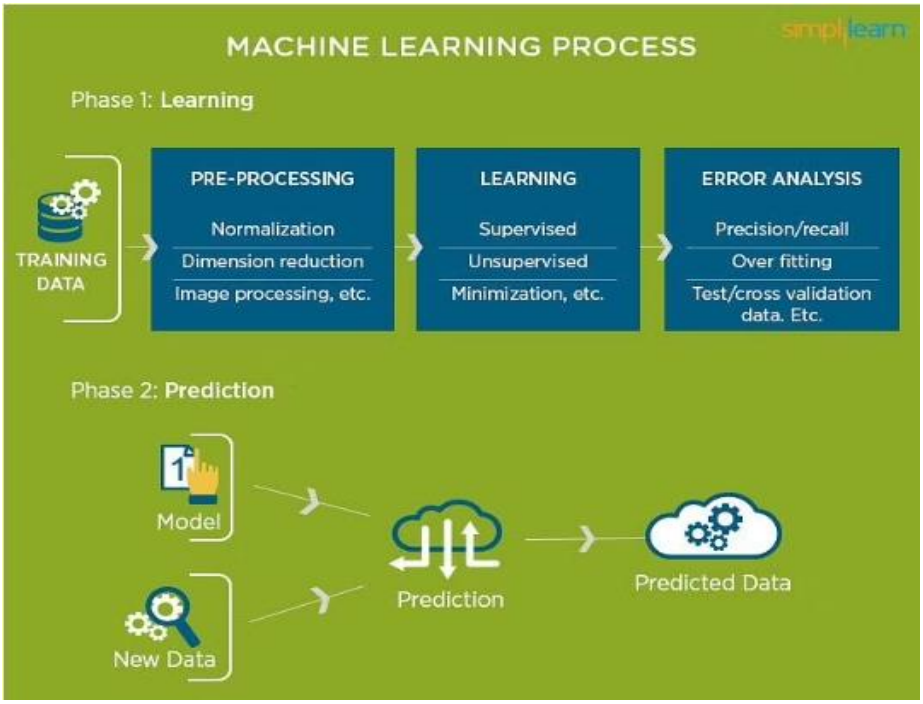


Figure 1.1: Machine Learning Work Process

The theft and fraud conducted using a payment card, such as a credit card or debit card, continues to be a major problem [2][3][4][5]. The widespread usage of fraud detection algorithms is one response to this problem. In addition to the actual card being stolen, fraudulent activity on a card may occur when the card is compromised in some way that makes the transaction seem genuine, such as by skimming, a hack, or an account takeover. The Global Payments Report 2015 found

that credit cards were the most popular payment method worldwide in 2014 [6]. This was in comparison to e-wallets and bank wire transfers. Credit card fraud is on the increase with the popularity of using plastic. The banking sector is feeling the effects of the increase in credit card theft. In 2015, the total amount of fraudulent charges on credit cards worldwide amounted to an astounding USD 21.84 billion.

Data mining and machine learning are widely used to examine patterns of typical and atypical behavior and individual transactions to detect and prevent fraudulent activity. As things stand, the most efficient use of resources is to use statistical algorithms to sift through accessible data in search of any proof of fraud. For each transaction, a fraud likelihood score is assigned based on the supervised model's examination of all labeled transactions and the model's mathematical determination of what a typical fraudulent transaction looks like. Various supervised methods, such as the neural network, K-Nearest Neighbor, Logistic Regression, Random Forest, and support vector machines (SVMs), as well as decision trees, are used to make predictions [4][9][11]-[21]. Nonetheless, a thorough comparison of all the prevalent algorithms, especially with actual data sets, has received little coverage in the literature for this study, we used a highly skewed dataset to distinguish between legitimate and fraudulent financial dealings. By under-sampling a portion of the data, we have created a more representative sample. We also used sample sizes and ratios to evaluate the performance of various models. As soon as we've compared every model's output, we'll announce which one is the most appropriate for this activity.

A firm is said to be in a state of bankruptcy when, as a result of its financial situation or its obligations, they are unable to conduct its business or continue its operations in the future. One of the most important issues in finance is learning how to predict who will go bankrupt. Many stakeholders, including investors, managers, and creditors, have a vested interest in knowing how likely it is that a company would go bankrupt. For this reason, forecasting insolvency has been the subject of a great deal of research. Beaver (1966) presented the first statistical rationale for the capacity of financial measures to account for default when he published a univariate analysis in the late 1960s. Then in 1968, Altman created the Z-score model, which uses five financial parameters to foretell the demise of U.S. businesses. Before making any financial commitments to a company, any sane business or investor would do well to learn its bankruptcy status. A uniform strategy may seem like a kind gesture when trying to anticipate any state, but it is very unlikely to

provide satisfactory results due to preexisting conditions. The study's overarching objective is to assess how well machine learning algorithms can anticipate bankruptcy rates and states of affairs and to analyze the results of such predictions. By using supervised learning techniques, we were able to pinpoint insolvency. We have not relied on just one machine learning technique but rather have used several different models, including logistic regression, k-nearest neighbors, support vector machines, and a decision tree classifier.

1.2 Related Works

N Khare and SY Sait used ML methods such as logistic regression, decision trees, support vector machines, and random forest to identify credit card fraud (RF). These classifiers were tested on a 2013 European credit card fraud dataset. This dataset is extremely uneven because the ratio of non-fraudulent to fraudulent transactions is skewed. The researcher evaluated each ML approach's categorization accuracy. LR, DT, SVM, and RF achieved 97.70%, 95.50%, 97.50%, and 98.60% accuracy, respectively. Authors argued that sophisticated pre-processing approaches might improve classifier performance [22]. Varmedja et al. [23] used ML to identify credit card fraud. Kaggle [24] provided a credit card fraud dataset. This dataset includes 2-day European credit card transactions. The researcher used SMOTE to cope with the dataset's class imbalance. RF, NB, and multilayer perceptron were used to evaluate the suggested technique (MLP). The RF algorithm detected fraud with 99.96% accuracy in experiments. NB and MLP techniques had 99.23% and 99.93% accuracy. The authors agree that further study is needed to increase the accuracy of ML algorithms via feature selection.

Machine learning automatically finds insights in data. Richard P. Hauser and David Booth [25] utilized machine learning to forecast bankruptcy. This research employs robust logistic regression, which determines the highest trimmed correlation between the remaining data and the predicted model [25]. This model has weaknesses. This strategy depends on including the right independent variables. If researchers don't identify all important independent variables, logistic regression is useless [26]. Its training set accuracy is 75.69% and its testing set accuracy is 69.44%. Using evolutionary algorithms, Myoung-Jong Kim and Ingo Han discovered experts' judgment principles using qualitative bankruptcy data in 2003. Inductive learning algorithms (decision trees), evolutionary algorithms, and neural networks are used without dropout. Since GA genomes have

fixed lengths, encoding an issue is difficult. GA doesn't guarantee global maxima. One-step-ahead node splitting without backtracking may provide a poor tree in inductive learning. Small data fluctuations might lead decision trees to be unstable [27]. Lack of dropout in neural network models encourages overfitting, which reduces accuracy. Accuracy is 89.7%, 94%, and 90.3%.

To better understand these two subjects, we have selected the K-nearest neighbor, Decision tree classifier, Logistic Regression, and support vector machine models. These models either use a recently discovered method, or we have sampled in such a way as to improve the accuracy of credit card fraud and bankruptcy detection.

1.3 Thesis Objectives

The thesis focused on two main areas: the identification of credit card fraud using machine learning, and the detection of bankruptcy using machine learning. We have gathered data from two supervised projects for this thesis and combined them into a single dataset. Since the datasets are significantly skewed in different directions, we've used the right methods to rebalance them, and we've developed several models and algorithms to anticipate the intended outcomes with more precision. The optimal models for these two datasets must be discovered via careful analysis and comparison.

1.3.1 Thesis Outlines

The remainder of the thesis is organized below:

- Chapter 2 introduces the reader to the fundamental concepts behind Machine Learning. Methods and ideas of Machine Learning are explored in great depth.
- Chapter 3 covers the tools and procedures that will be used on two separate assignments. In addition, the theoretical underpinnings of the used Machine Learning models are elaborately explained.
- Chapter 4 summarizes the experimental findings and engages in a thorough analysis of them. The similarity has also been shown with convincing justifications.
- Chapter 5 brings the thesis to the conclusion. In that chapter, there is also a discussion of future work and proposals for model change.

CHAPTER 2

MACHINE LEARNING TECHNIQUES AND WORKING PRINCIPLE

2.1 Machine Learning (ML)

Machine learning is writing code that instructs a machine to maximize some performance metric by studying and learning from examples. Learning is the process of running a computer program to optimize the model's parameters in light of training data or prior experience. The model is specified up to a certain set of parameters. A predictive model is used to anticipate the future, whereas a descriptive model is used to understand the past. In machine learning, researchers ask how to make computers learn from their own mistakes and become better over time. Machine learning, a subfield of AI and computer science, seeks to improve its accuracy by modeling human learning strategies using data and computational models. The discipline of data science is expanding rapidly, and machine learning plays an essential role in this development. In data mining projects, statistical approaches are used to train computers to classify data, generate predictions, and find hidden insights. These discoveries inform application and commercial decisions, which should eventually affect vital growth indicators. The need for skilled data scientists will rise as the prevalence of big data increases. Their assistance will be needed to determine which business issues are of the utmost importance, as well as which data is needed to answer those concerns. Frameworks that expedite solution creation are often used while developing machine learning algorithms. Examples of such frameworks are TensorFlow and PyTorch. Machine learning software can do tasks without being told how to accomplish them. Learning to do activities automatically by analyzing and interpreting data presented. Computers don't need to learn how to do basic jobs since we can instruct them step-by-step through the process of solving the issue. It might be difficult for a person to manually build the necessary algorithms

for increasingly complex jobs. In reality, assisting the computer in developing its algorithm may be more successful than having human programmers specifically explain each step [3].

These days, machine learning is used for forecasting future events as well as classifying data according to pre-existing models. To train a hypothetical data-classification system to distinguish between healthy and malignant moles, it may employ computer vision of moles in conjunction with supervised learning. The stock trader might be given a glimpse of the future thanks to a machine learning program.

In the 1990s, machine learning (ML) began to develop as a distinct subject. Artificial intelligence was once the ultimate objective of the profession, but now it's all about solving real-world issues. From the symbolic techniques it had acquired from AI, it moved its attention to statistical, fuzzy logic, and probability theory-informed methodologies and models [4]. Often, people don't realize that there's a distinction between machine learning and artificial intelligence. Machine learning (ML) is observation-based; artificial intelligence (AI) involves active participation from the agent to optimize its chances of success [5].

There are a lot of crossovers between machine learning and data mining, and the two fields share many of the same techniques. However, whereas machine learning is concerned with making predictions based on characteristics already established through training data, data mining is concerned with uncovering novel, previously undiscovered characteristics (this is the analysis step of knowledge discovery in databases). Different from the aims of supervised learning, the data mining approaches used in machine learning are used as "unsupervised learning" or as a preprocessing step to increase learner accuracy.

Many learning problems in machine learning are phrased as minimization of some loss function on a training set of instances, hence optimization is also a close companion of machine learning. When training a model, the loss function is used to quantify how far off the model's predictions are from the truth. Generalization is what sets optimization apart from machine learning; whereas optimization methods can reduce the loss on a training set, machine learning focuses on doing the same for unseen data. Especially for deep learning systems, characterizing its generalization is a hot subject in the field right now. Statistics provide population conclusions from a sample, whereas machine learning discovers predictive patterns. Michael I. Jordan says machine learning's methods

and tools have a long history in statistics [6]. He also recommended calling the discipline data science [7]. Leo Breiman identified two statistical modeling paradigms: data model and algorithmic model, which implies machine learning methods like Random forests [8]. Some statisticians use machine learning approaches, creating statistical learning [9].

2.2 Machine Learning Methods

The method by which an algorithm is taught to improve the precision of its predictions is often used as a classification scheme for classical machine learning. Learning may be accomplished in several ways, the most fundamental of which are supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning. The kind of data that scientists working with data seek to forecast determines the algorithm that they go with to make that prediction.

A. **Supervised Learning:** Supervised learning makes use of a mathematical model consisting of inputs and desired outputs for training purposes. "Supervised" learning refers to this kind of instruction. There is a supervisory signal, or output, for each input that matches. These outputs are paired with their corresponding inputs. Using the provided training matrix, the system can infer the relationship between the input and the output. After training, it will apply this correlation to new inputs to discover the corresponding output. If the output requirements are specific enough, supervised learning may be used in other areas, such as classification and regression analysis. Classification analysis is used when the range of possible outcomes is finite. However, a numerical range for the output values is established by regression analysis. Systems that use voice or face recognition to verify authenticity are two applications of supervised learning. In the field of machine learning, the job of learning a function that maps an input to an output based on example pairs of inputs and outputs is referred to as supervised learning. The data that is provided has been labeled. Both regression and classification issues fall within the category of supervised learning algorithms. Data scientists provide algorithms with labeled training data and identify the variables they want the algorithm to examine for correlations to participate in this form of machine learning. The inputs and outputs of the algorithm are both laid out in the description.

gender	age	label
M	48	sick
M	67	sick
F	53	healthy
M	49	sick
F	32	healthy
M	34	healthy
M	21	healthy

Figure 2.1: Example of Supervised learning

The dataset in Figure 2.1 above comprises new hospital admissions. Each patient is given a label indicating whether they are healthy or ill, and there are two columns labeled "gender" and "age" to reflect the gender and age of the patients. This is a supervised dataset because of its standardized nature of the data.

B. Unsupervised Learning: Unsupervised learning is a form of machine learning method that is used to derive conclusions from datasets that have input data but do not include labeled replies. In unsupervised learning techniques, the observations do not contain any kind of classification or categorization of the data being learned. Unsupervised learning, on the other hand, use training data that does not include the output in the same way that supervised learning does. The unsupervised learning method detects the input based on trends and similarities; the output, on the other hand, is decided based on the presence or absence of such patterns in the user input.

gender	age
M	48
M	67
F	53
M	49
F	34
M	21

Figure 2.2: Example of Unsupervised Learning

New hospital admissions are included in the dataset which can be seen above in Figure 2.2. Two columns indicate the gender and age of the patients, but there is no level of access. These columns are named "gender" and "age," respectively. Because the data are not standardized, this collection of information is known as an unsupervised dataset. It's a kind of learning that's analogous to how people determine that things or occurrences belong to the same category, such as by comparing their levels of resemblance. This style of learning is at the heart of several marketing automation suggestion systems.

C. **Reinforcement Learning:** The task of persuading an agent to behave in the environment in such a way as to maximize its rewards is the central focus of reinforcement learning. Instead of being instructed on what actions to perform, as is the case in the majority of kinds of machine learning, a learner is required to experiment with various activities to determine which ones result in the greatest amount of reward. Consider the scenario of teaching a new skill to a dog. Although we cannot tell the dog what to do, we may reward or penalize it depending on whether or not it follows our instructions correctly. When you

watch the movie, you'll see that the program starts awkward and inexperienced, but it becomes better and better as it goes through its training until it eventually becomes a champion. In teaching the system to choose a relevant context, this is employed to ascertain the best course of action given the current situation. Training gaming sites often use these to adjust gameplay based on player feedback. We explore a scenario where there is an agent, who is trying to go to a prize, and there are several obstacles along the way. To earn its reward, the agent must choose the most efficient means of doing so. The following problem provides a clearer illustration of the issue.

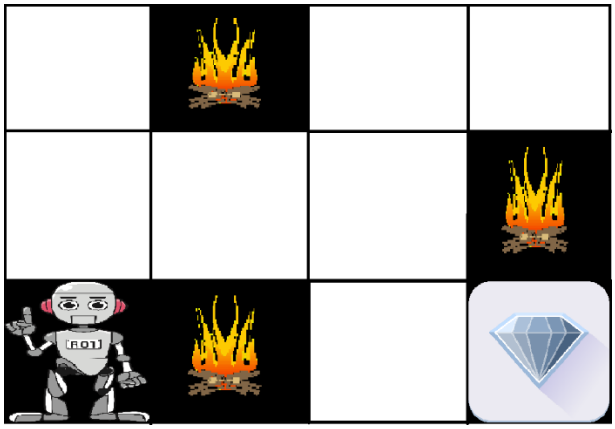


Figure 2.3: Example of Reinforcement Learning

The robot, diamond, and flames are shown up there. The robot's mission is to collect the diamond prize while avoiding the exploding obstacles. The robot learns by exploring all of its options and picking the one that leads to the greatest reward with the fewest obstacles. After doing each correct action, the robot will be rewarded, while incorrect actions will have the opposite effect. We'll add up all the rewards until we get to the diamond, at which point we'll do the math.

D. Semi-supervised learning: In situations when an incomplete training signal is provided, an example of this would be a training set that was missing some or many of the intended outputs. There is a particular application of this idea that is referred to as “Transduction.” In this scenario, the whole collection of issue instances is known at the time of learning; however, a portion of the targets is absent. During the training phase of machine learning,

the semi-supervised learning strategy mixes a relatively small quantity of data with a substantial amount of data that has not been labeled. Learning in a semi-supervised environment is a middle ground between learning in an uncontrolled environment and learning in a controlled environment.

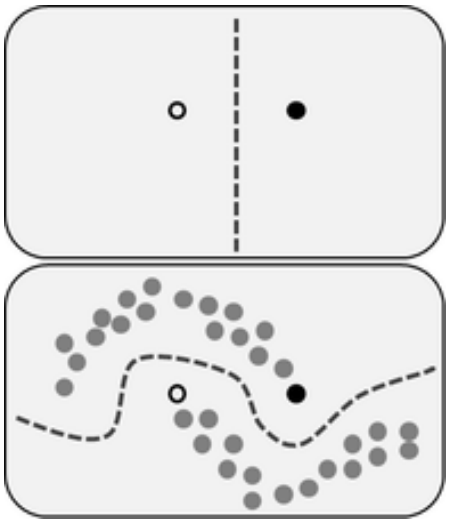


Figure 2.4: Example of Semi-Supervised Learning

Figure 2.4 shows how semi-supervised learning uses unlabeled data. The top panel depicts a decision boundary we may adopt after witnessing one good (white circle) and one negative (black circle) scenario. The bottom panel depicts a decision boundary if we had unlabeled data in addition to two labeled samples (gray circles). This might be interpreted as clustering and then labeling the clusters with labeled data, pushing the decision boundary away from high-density areas, or learning an underlying one-dimensional manifold where the data live.

2.3 Machine Learning Working Principle

It's safe to say that within the field of AI, Machine Learning is one of the most fascinating and rapidly developing areas. Learning from data is finished when the system is given clear instructions. Learning how Machine Learning works is crucial for making informed decisions about its future applications.

Training data is first input into the chosen algorithm in Machine Learning. The final Machine Learning method requires training data, which may be either known or unknown information.

To ensure the machine learning algorithm is functioning properly, new input data is fed into it. We then compare the forecast to the actual outcome.

When there is a discrepancy between what was predicted and what happened, the algorithm is retrained until the data scientist is satisfied. As a result, the machine learning algorithm may improve its accuracy over time by learning on its own and generating the best possible solution.

Machine learning is a subfield of AI that aims to program computers to learn and improve on their own, just as people do. Data is mined for hidden patterns with little to no human oversight.

Machine learning may be used to automate almost any operation if a pattern or set of rules can be identified for it in the data. Companies may now automate tasks that were formerly performed by people, such as taking and processing customer service calls, maintaining financial records, and screening applications. We have covered many of the models and methods that machine learning applies in the preceding subsection (2.2). However, machine learning primarily employs two methods: supervised and unsupervised (Figure 2.5).

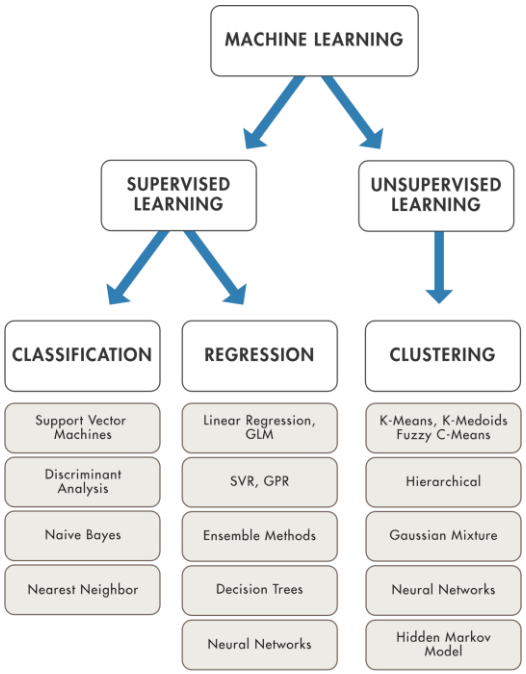


Figure 2.5: Machine Learning techniques
[24]

The data for a supervised learning task may be obtained from an existing ML deployment or created similarly. Excitingly, supervised learning mimics the manner that people do, making it a viable alternative to machine learning. The training set is a collection of data points that have been tagged and presented to the computer in supervised activities.

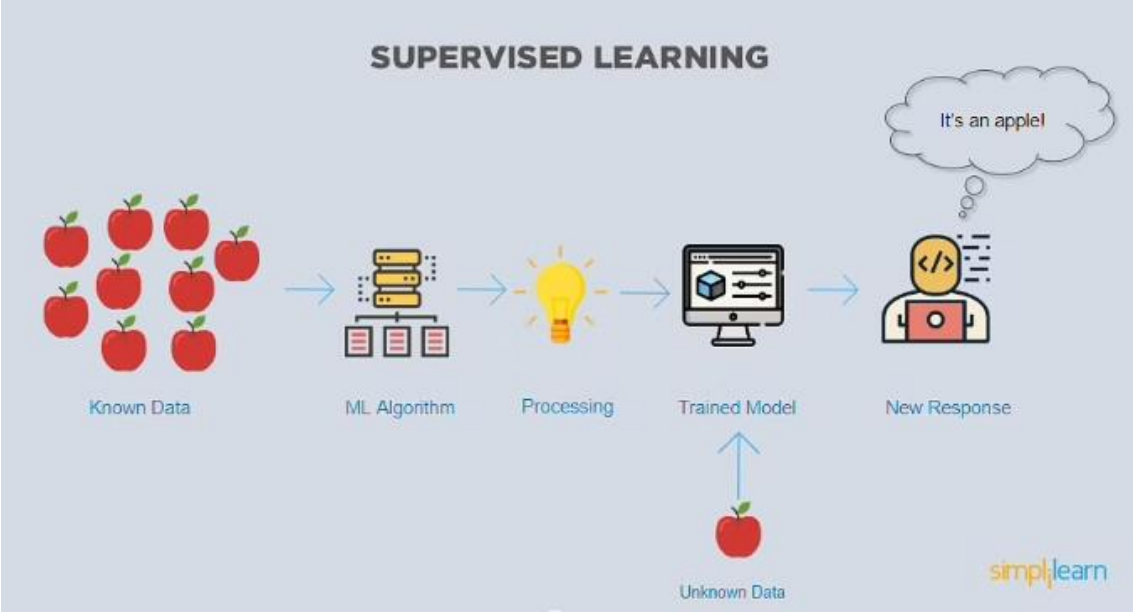


Figure 2.6: Supervised Learning technique

Here, the model (Figure 2.6) looks for clues as to whether the input is an apple or some other kind of fruit. In other words, after the model has been properly trained, it will correctly determine that the input data is an apple and respond accordingly.

Unsupervised machine learning is a method that uncovers patterns in data that were previously concealed by relying on a machine's inherent capacity to learn without the assistance of a human instructor. Unsupervised learning is a kind of machine learning in which the data are analyzed without the assistance of human instructors or labels to guide the process. Clustering and dimensionality reduction are two examples of common difficulties that arise in unsupervised learning.

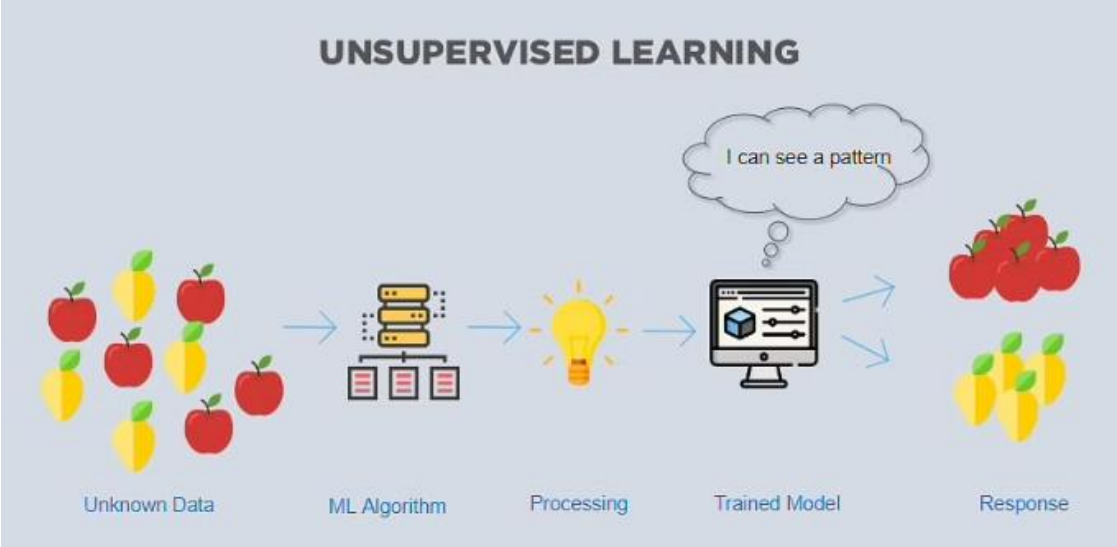


Figure 2.7: Unsupervised Learning technique

In this scenario (Figure 2.7), the unknown data consists of apples and pears, which are easily confused with one another due to their appearance. The trained model works to organize them all in such a way that you end up with the same items in groups that are comparable to one another.

Clustering is a method for organizing large amounts of data by looking for groups of comparable items that are distinct from one another. Market segmentation is only one application where clustering can be in handy. To better analyze a dataset, dimensional reduction models aggregate variables with similar or correlating characteristics (and more effective model training).

With the current boom in interest in machine learning, academics have created a dizzying array of algorithms for use in the discipline. There may be many different models to choose from, but ultimately, they can all be broken down into three categories. The three main parts of every machine learning model are the representation, the evaluation, and the optimization. Even while these three are most often discussed in the context of supervised learning, they may also be used in unsupervised learning.

- **Representation-** This is the method through which you want to examine the information. Data is often best analyzed when seen as people (as in k-nearest neighbors) or nodes and edges (as in a graph) (like in Bayesian networks).

- **Evaluation-** In supervised learning, the goal is to have the learner become better over time based on a score you give it based on how well it did. To do this analysis, an evaluation function is used (also known as an objective function or scoring function). Accuracy and mean squared error are two such examples.
- **Optimization-** The goal is to use your preferred optimization method on the aforementioned evaluation function to determine which student has the highest score. Among them are the greedy search and the gradient descent.

Machine learning's strength stems from not needing to hard code or explicitly specify data parameters. Machine learning generalizes a learner's discoveries. To examine a learner's generalizability, utilize a distinct, non-training data set. This may be built by dividing your training data set or gathering new data. Using test data would prejudice the learner to perform better than in reality. Cross-validation is one way to predict a learner's performance on a test data set. This randomly separates training data into a certain number of subgroups (for example, ten) and leaves one out while the learner learns on the rest. After training, the learner is tested with the left-out data. This training leaves out one subset and tests as you rotate subsets.

Overfitting the model is another crucial issue in machine learning. A good hypothesis is not all that is indicated whether a learning algorithm works effectively with a certain dataset. When the hypothesis function $J(\Theta)$ has a large variance and low error on the training set but a high-test error on any other data. This is called overfitting. Overfitting occurs when the error of the hypothesis on the data set used to train the parameters is less than the error on any other data set [12].

2.4 ML Programming languages and tools

The majority of data scientists have some familiarity with the usage of the R and Python programming languages for machine learning, but there are many more options available as well. Software libraries, toolkits, and suites are common types of machine learning and AI tools that facilitate the completion of tasks. Python, however, is the most popular language for machine learning due to its extensive support and plenty of libraries.

According to GitHub, Python is the best language for machine learning. Python facilitates the development of several machine learning models and methods, making it a popular choice for these tasks.

classification, regression, clustering, and dimensionality reduction are all methods that can be run in Python. Python is the most widely used language for machine learning, although it is far from the only option. Machine learning operations (MLOps) are useful because some ML applications use models written in languages other than their own. One must be fluent in computer-readable programming languages to put Machine Learning into practice. Listed below are some of the most popular languages used for Machine Learning programming. The most popular languages for Machine Learning in 2021, according to the GitHub 2021report, are listed below:

RANK	Programming Language
1	JavaScript
2	Python
3	Java
4	Go
5	TypeScript
6	C++
7	Ruby
8	PHP
9	C#
10	C

Table 2.1: Top ML programming Language (Source: GitHub).

Open-source tools for machine learning are nothing more than libraries that may be used in programming languages such as Python, R, C++, Java, Scala, and Javascript, amongst others, to get the most out of machine learning algorithms.

- **Keras**: Keras is a Python library for neural networks that are freely available to the public. It may be implemented as a TensorFlow plug-in.
- **PyTorch**: PyTorch is a Torch-based Python Machine Learning toolkit that may be used in NLP and other similar applications.
- **TensorFlow**: TensorFlow is an open-source toolkit for numerical computing and massive-scale Machine Learning developed by the Google Brain team.
- **Scikit-learn**: Python's Scikit-learn package (also known as Sklearn) has been widely used to address issues in the hard sciences, mathematics, and statistics because of its accessibility and versatility in the realm of machine learning.
- **Shogun**: Shogun is compatible with popular programming languages including Java, Python, R, Ruby, and MATLAB. It provides an extensive collection of well-integrated Machine Learning techniques.
- **Spark MLlib**: Spark MLlib is the Apache Spark and Apache Hadoop Machine Learning library. While Java is the recommended language for MLlib work, the NumPy package allows Python users to use MLlib as well.

2.5 Building a Machine Learning Model

The process of generating a model for machine learning is quite similar to the process of producing a product. To mention just a few phases, there is the ideation phase, the validation phase, and the testing phase. In most cases, the construction of a model for machine learning may be split down into the following processes.

1. **Collecting Dataset:** The importance of high-quality training data in the field of machine learning cannot be overstated. As was previously discussed, the training dataset consists of a set of data points. These facts instruct the model on how to approach the issue it was built to solve. The training dataset often consists of some combination of visual, textual, auditory, and other sensory inputs. The training data set is analogous to a textbook in a math course. The more concrete instances provided, the better. If we want a reliable model, it's not enough to just have a large dataset. The actual world circumstances under which the model will be deployed should be reflected in the training dataset.

Figure 2.8: Dataset Collection

2. **Preparing the Dataset:** The degree of labeling in the training dataset might range from complete to none to partial. This aspect of the dataset is, as was noted before, method-specific when it comes to machine learning.

   In any case, there can't be any repetitions in the data used for training. A high-quality dataset will have been cleaned via many passes and will have all the necessary characteristics for the model to learn. Preparing a dataset involves the following procedures:

   - Combining all with a dash of randomness. This ensures that data is spread out uniformly and that the order of the input has no effect on the efficiency of the learning process.
   - Removal of duplicates, invalid data types, missing values, rows, and columns, and other forms of data cleaning. It may need to rearrange the data, making new rows, columns, or perhaps a whole new index.
   - Make sense of the data's structure and the interconnections between its many variables and categories by seeing it graphically.
   - Separating the cleansed data into a training and testing set. The data used to teach your model is called the training set. After training the model, its performance may be evaluated using a testing set.

Figure 2.9: Preparing dataset by cleaning and visualizing

3. **Choosing an algorithm or model:** The results of applying a machine learning algorithm to a dataset are predetermined by a machine learning model. Selecting a model that is not appropriate for the work at hand is a common mistake. Scientists and engineers have spent years perfecting a wide range of models, each of which is ideally suited to a particular use. In addition, it needs to determine whether or not the model works best with numerical or categorical data. The issue we're trying to answer, the nature of the data (labeled or unlabeled), and the quantity of data at our disposal all play a role in deciding which method to use.

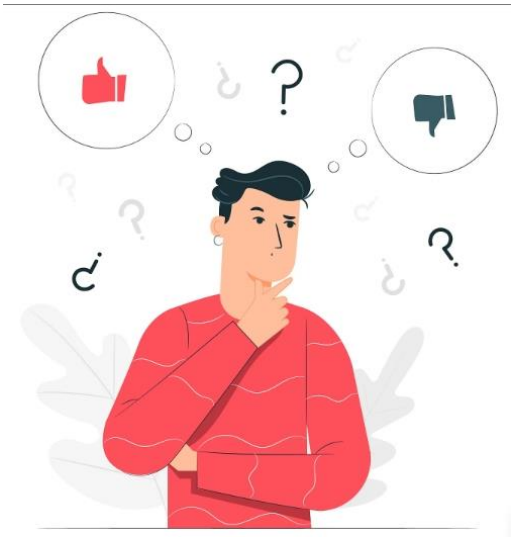


Figure 2.10: Choosing a model

Some potential algorithms for labeled data sets are as follows:

- Decision Tree
- Linear Regression
- Logistic Regression
- Support Vector Machine
- K-Nearest Neighbor

Some potential algorithms for unlabeled data sets are as follows:

- K-means Clustering algorithm
- Apriori algorithm
- Singular Value Decomposition
- Neural Networks

4. **Training the algorithm or model:** The primary process in machine learning is training. In this stage, the algorithm goes through several iterations. Adjustments to the algorithm's weights and biases are made after each iteration by comparing the actual results to the ideal ones. When this happens, we have a machine learning model, which is the result of the algorithm's improved accuracy. It leads to the model acquiring the knowledge it needs to complete the goal-oriented job. The model's predictive abilities improve with repeated training.

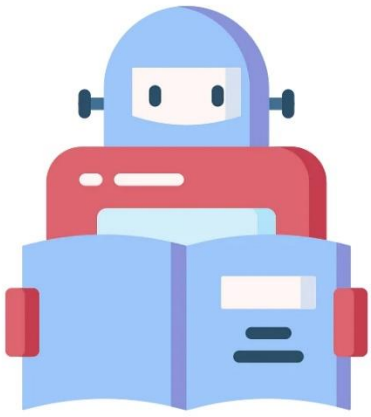


Figure 2.11: Training the Model

5. **Evaluating the Model:** After the model has been trained, we must evaluate its results. To achieve this, we put the model through its paces with some data we have never seen before. The data utilized in the test is from the testing set we created previously. Because the model is already familiar with the data and looks for the same patterns it did during training, a reliable result would be impossible to get if testing was conducted using the same data that was used during training. Thanks to this, our precision will be remarkably increased.
   By applying it to test data, we can obtain a good idea of how well and quickly our model will function in the real world.

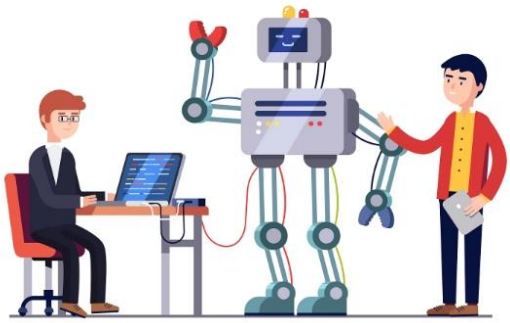


Figure 2.12: Evaluating the model

2.6 Machine Learning Models & Algorithms: Theoretical Discussion

Here, we'll go through the machine learning models we stated before and explain the algorithms that run underneath them. We employed four algorithms in our thesis, and we'll go through them here.

- **K-Nearest Neighbor:** The machine learning technique known as K-nearest neighbors (KNN) is a kind of supervised learning that may be used for both regression and classification problems. It is necessary to provide a supervised machine learning algorithm with labeled training data for the algorithm to be able to use the information it has gained to deliver reliable results when unlabeled data is provided as input. KNN is used to generate predictions on the test data set using features (labels) from the training data. By comparing the test and training sets and using the resulting distance, it is possible to make educated guesses about the future.
  KNN is an effective classification algorithm. Data Scientists and Machine Learning engineers categorize data regularly. It solves several problems. KNN is a reliable method

for categorizing and regressing patterns. Because it's assumption-free, KNN outperforms other classification algorithms. It's easy to apply and comprehends. 'K' in KNN stands for the number of closest neighbors, and this value provides a context in which data points may learn to recognize and appreciate their shared characteristics with those in close physical vicinity. To classify unlabeled data more accurately, we calculate the K-value, which is the distance between the test data points and the training labeled points. It is recommended that the K-value be odd, and the K-value itself be a tiny positive integer. Overfitting occurs when the K-value is low because the model is too restrictive when the error rate is low and the bias is large but the variance is low. We utilized hamming distance to determine how far apart our data points were for our thesis. Two binary data strings are supplied to the Hamming distance algorithm, which then calculates the number of distinct characters at each place in both strings.

- Hamming Distance: This method is often used with Boolean or string vectors to pinpoint the discrepancies between them. This has led to the term's alternative name, "overlap metric," being used. An appropriate formula for this is as follows:

$$\text{Hamming Distance} = D_H = \left(\sum_{i=1}^{k} |x_i - y_i|\right)$$

$$x=y \quad D=0$$
$$x\neq y \quad D\neq 1$$

Figure 2.13: Hamming distance formula

- **Decision Tree:** A kind of supervised learning method, decision trees are not uncommon. The decision tree approach, unlike other supervised learning algorithms, may be used for both regression and classification issues. Using basic decision rules derived from existing data, a Decision Tree is a training model used to predict the class or value of the target variable (training data). When using Decision Trees to determine the most likely category for a given record, we begin at the top of the tree. We check the attribute values of the root

element against those of the record. When comparing two values, we take the branch that leads to the next node based on the difference.

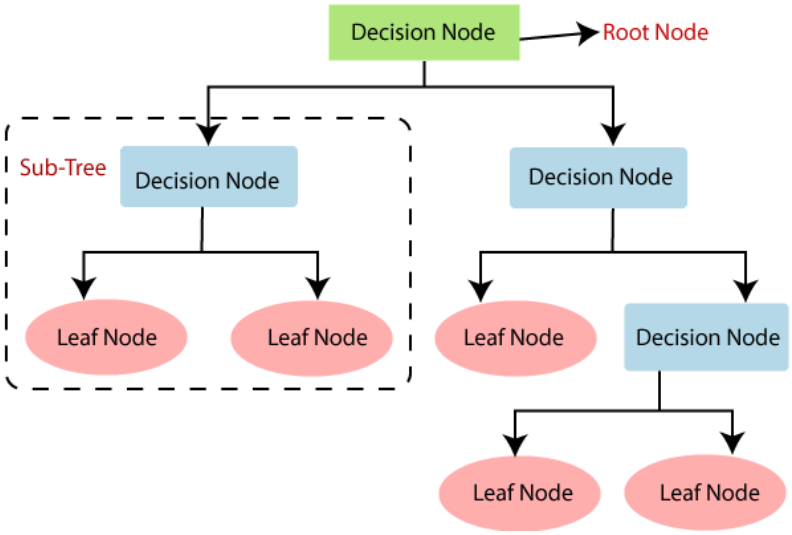


Figure 2.14: Decision Tree Model structure

Each decision tree has a starting point, known as the root node. It is a representation of the complete dataset that is then split into two or more similar groups. After a leaf node is obtained as an output, further division of the tree is impossible. The term "splitting" refers to the process of subdividing the decision node/root node into other nodes based on the input criteria. When a tree is cut in half, it creates a new branch. When a tree is pruned, dead or diseased branches are cut off. All other nodes in the tree are considered children of the root node.

The algorithm in a decision tree starts at the leaf node and works its way up to predict the category of the input dataset. In this case, the algorithm compares the values of the root property with those of the record (actual dataset) attribute, and if they're different, it continues down the branch and skips to the next node. As before, the next node's attribute value is compared to that of the preceding sub-nodes. It does this repeatedly until it reaches the leaf node of the tree. To further understand the process, please refer to the following algorithm:

- **Step-1:** S advises starting the tree from the root node, which has the whole dataset.
- **Step-2:** Utilize the Attribute Selection Measure to identify the dataset's top attribute (ASM)
- **Step-3:** Subsets of the S that include potential values for the best qualities should be created.
- **Step-4:** Create the best attribute-containing decision tree node.

- **Step-5:** Using the subsets of the dataset generated in step 3, repeatedly design new decision trees. Continue down this path until you reach a point when you can no longer categorize the nodes and you refer to the last node as a leaf node.

- **Support Vector Machine:** Classification and Regression issues are two common applications of the Supervised Learning method known as Support Vector Machine (SVM). Though its primary use is in Machine Learning Classification issues. To classify fresh data points efficiently in the future, the SVM algorithm seeks to find the optimal line or decision boundary that divides the space into n distinct classes. A hyperplane defines the optimal boundary for making a choice. To create the hyperplane, SVM picks the most extreme points and vectors possible. Such outlier examples are referred to as support vectors, and the resulting method is known as a Support Vector Machine. Take a look at the picture below, which uses a decision boundary (or hyperplane) to classify items into two groups:

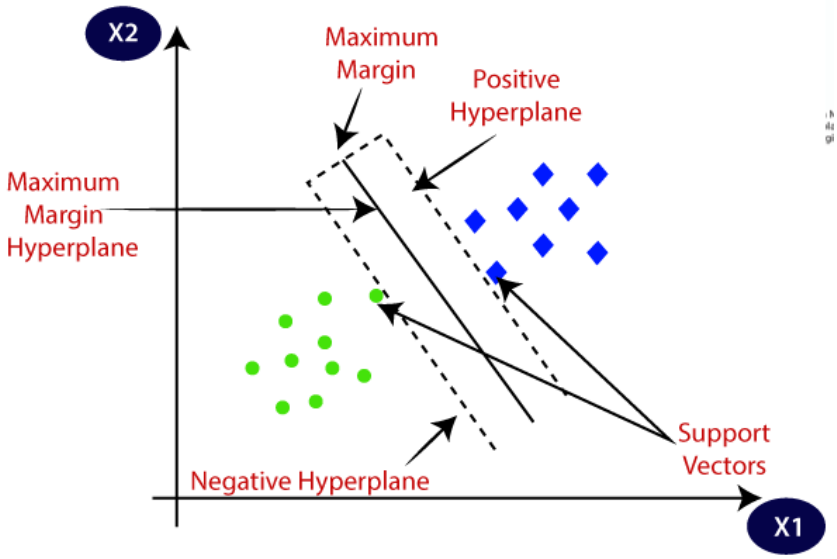


Figure 2.15: SVM working principle

Making a straight line between two classes is how a basic linear SVM classifier works. In other words, all of the data points to the left of the line will be assigned to one category, while all of the data points to the right of the line will be assigned to another. This suggests that the potential number of lines is essentially limitless. The linear support vector machine (SVM) method outperforms other algorithms such as the k-nearest neighbors' classifier because it selects the most informative line to use in classifying your data. The line that

best divides the data and is furthest from the nearest data points is selected. We have, essentially, a grid with some data points on it. We're attempting to classify these pieces of information, but we don't want any of them to end up in the incorrect bucket. In other words, we want to locate the line that connects the two nearest points while maintaining the separation of the remaining data points.

Thus, the two nearest data points provide the support vectors we need to locate that line. The decision boundary is that dividing line.

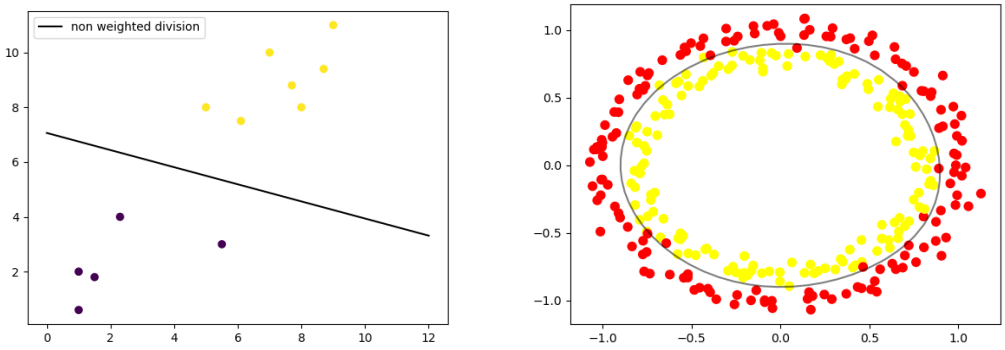


Figure 2.16: Linear SVM & Hyper-plane SVM

The border of the choice need not be a straight line. Since the decision boundary may be located using more than two characteristics, it is often referred to as a hyperplane.

- **Logistic Regression:** One of the most well-known Machine Learning algorithms, logistic regression belongs to the category of Supervised Learning. To predict a categorical dependent variable from a collection of independent factors is utilized. Using a categorical dependent variable, the outcome may be predicted using logistic regression. This means that the result must take the form of a categorical or discrete number. To avoid supplying the precise values of Yes and No, 0 and 1, true and False, etc., it instead provides probabilistic values that range from 0 to 1. Logistic Regression is comparable to Linear Regression in how it's employed. Linear regression is used for regression issues, whereas logistic regression is used for classification. In logistic regression, we fit an "S"-shaped logistic function that predicts two maximum values (0 or 1). The logistic function curve represents the possibility of things like malignant cells, obesity in an animal, etc. Logistic Regression is a key machine learning approach because it can categorize continuous and

discrete data. Logistic Regression can categorize observations using multiple forms of data and discover the best variables for classification. Below is the logistic function:

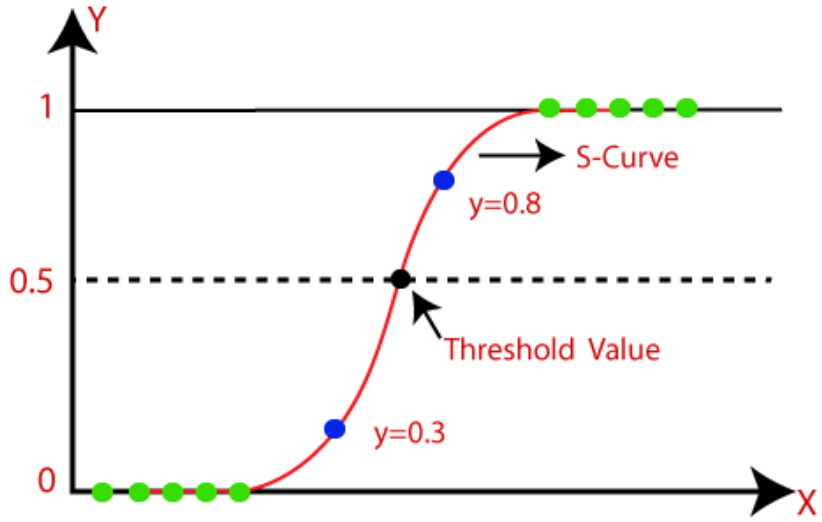


Figure 2.17: Logistic Function

Logistic regression may be divided into two distinct categories:

- Binary Logistic Regression Model- In binary or binomial logistic regression, the dependent or target variable may take on just two potential values. just two possibilities: 1 and 0.
- Multinomial Logistic Regression- In multinomial logistic regression, the dependent or target variable might contain three or more unordered kinds, which are types with no quantitative significance.

As an easy-to-understand S-curve, the logistic function transforms data into a number between 0 and 1.

$$h\Theta(x) = 1/1 + e - (\beta o + \beta 1X)$$

*'**hΘ(x)**' is output of logistic function , where 0 ≤ hΘ(x) ≥ 1*

*'**β1**' is the slope*
*'**βo**' is the y-intercept*
*'**X**' is the independent variable*

*(βo + β1*x) - derived from equation of a line Y(predicted) = (βo + β1*x) + Error value*

Figure 2.18: Logistic Regression Equation

CHAPTER 3
MATERIALS AND METHODOLOGY

In this section, we'll go a little further into our two thesis projects. In this section, we'll go through everything that's been done and how it's been done, including the methods and tools that were used and the sources of data, and major models that were utilized. All of the implemented algorithms will be documented, along with a thorough analysis of their rationale. In this section, we will examine two machine learning applications: "Credit Card Fraud Detection" and "Bankruptcy Detection".

3.1 Credit Card Fraud Detection

Victims of credit card fraud are often taken advantage of because they are easy prey. There is a higher chance of online fraud due to the proliferation of online payment options made possible by e-commerce and other online sites. Following an uptick in online fraud, scientists have turned to a variety of machine learning techniques to help them identify and analyze instances of crime. In this study, we used four different machine learning techniques (decision tree, support vector machine, k-nearest neighbors, and logistic regression) to highly asymmetric data and compare their performance. The information we acquired and the methods we used will be detailed in the following sections.

3.1.1 Dataset

We gathered the data for this research from the website "Kaggle". The dataset includes credit card purchases done across Europe in September 2013. For a total of 284,807 transactions, 492 were fraudulent throughout these two days. There is a significant imbalance in the dataset, with 0.172% of all transactions belonging to the positive class (frauds).

It takes just numeric variables produced by principal component analysis as input. Sadly, original characteristics and further context about the data were concealed owing to confidentiality

concerns. Components V1, V2, V3… Only the 'Time' and 'Amount' features were not PCA-transformed, therefore V28 represents the major components derived using PCA. Time is a feature that tracks how much time has passed from the very first transaction in the dataset. An example-dependent kind of cost-sensitive learning that may make advantage of the 'Amount' feature is proposed. Feature The 'Class' response field is true if fraud has occurred and false otherwise. Thus, this is a supervised dataset with a binary class.

	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...
0	0.0	-1.359807	-0.072781	2.536347	1.378155	-0.338321	0.462388	0.239599	0.098698	0.363787	...
1	0.0	1.191857	0.266151	0.166480	0.448154	0.060018	-0.082361	-0.078803	0.085102	-0.255425	...
2	1.0	-1.358354	-1.340163	1.773209	0.379780	-0.503198	1.800499	0.791461	0.247676	-1.514654	...
3	1.0	-0.966272	-0.185226	1.792993	-0.863291	-0.010309	1.247203	0.237609	0.377436	-1.387024	...
4	2.0	-1.158233	0.877737	1.548718	0.403034	-0.407193	0.095921	0.592941	-0.270533	0.817739	...

5 rows × 31 columns

V21	V22	V23	V24	V25	V26	V27	V28	Amount	Class
-0.018307	0.277838	-0.110474	0.066928	0.128539	-0.189115	0.133558	-0.021053	149.62	0
-0.225775	-0.638672	0.101288	-0.339846	0.167170	0.125895	-0.008983	0.014724	2.69	0
0.247998	0.771679	0.909412	-0.689281	-0.327642	-0.139097	-0.055353	-0.059752	378.66	0
-0.108300	0.005274	-0.190321	-1.175575	0.647376	-0.221929	0.062723	0.061458	123.50	0
-0.009431	0.798278	-0.137458	0.141267	-0.206010	0.502292	0.219422	0.215153	69.99	0

Figure 3.1: Credit Card Dataset first 5 rows.

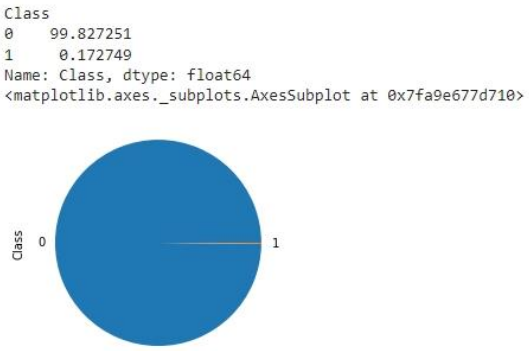


Figure 3.2: Shows a pie chart of the dataset's class distribution

The dataset's class distributions are shown in Figure 3.2. There are just two possible values in this dataset: 1 and 0. Because of the extreme inequalities in the dataset, the proportion of class 1 observations is much less than those of class 0.

3.1.2 Methodology

In this section, we will detail the procedures, standards, and algorithms that we used in the thesis work. Early on, we spoke about where and how we found all these datasets. In this step, we'll start working with the dataset, picking out the right machine learning methods, and training and fitting the models. An extensive theoretical explanation will be given of the models and algorithms employed in this research, as well as their underlying functioning principles.

There is a significant lack of equality in our credit card fraud detection dataset for normal and fraudulent transactions. As it contains only binary class columns, the dataset must first be balanced before it can be used in machine learning methods. Before then, however, a distinction was made between "fraudulent" and "regular" financial dealings. The default was set to 0 for legitimate purchases and 1 for fraudulent ones. Next, we made sure the class was ready to go by balancing the data. These categories will be predicted from the financial dealings using various machine learning methods.

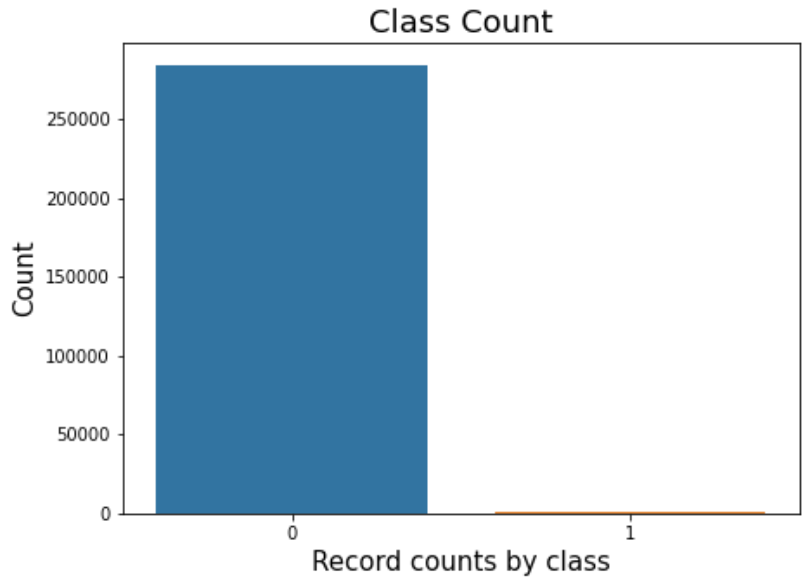
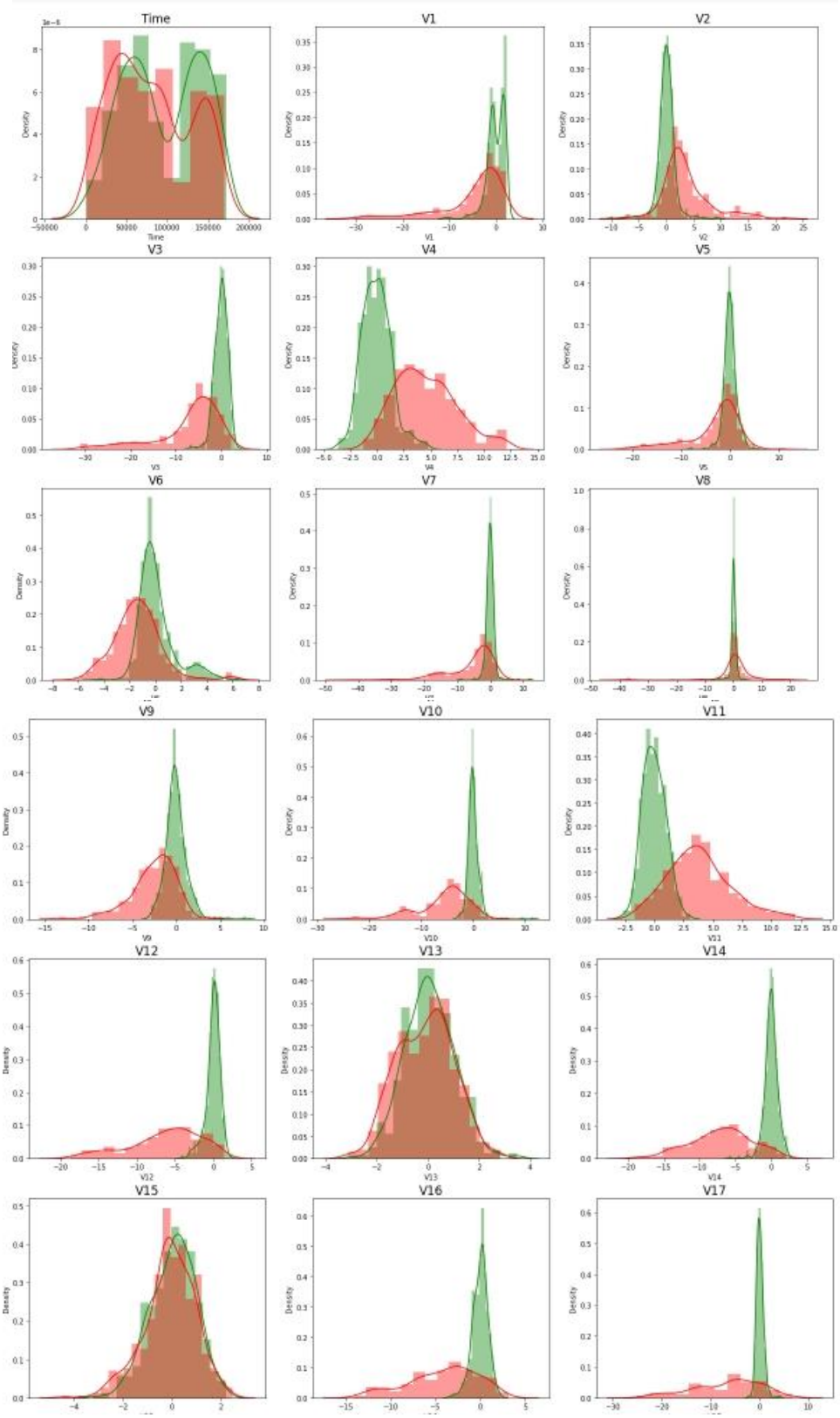


Figure 3.3: Class count of Normal (0) & Fraudulent (1) Transactions


Time
V1
V2
V3
V4
V5
V6
V7
V8
V9
V10
V11
V12
V13
V14
V15
V16
V17
Density

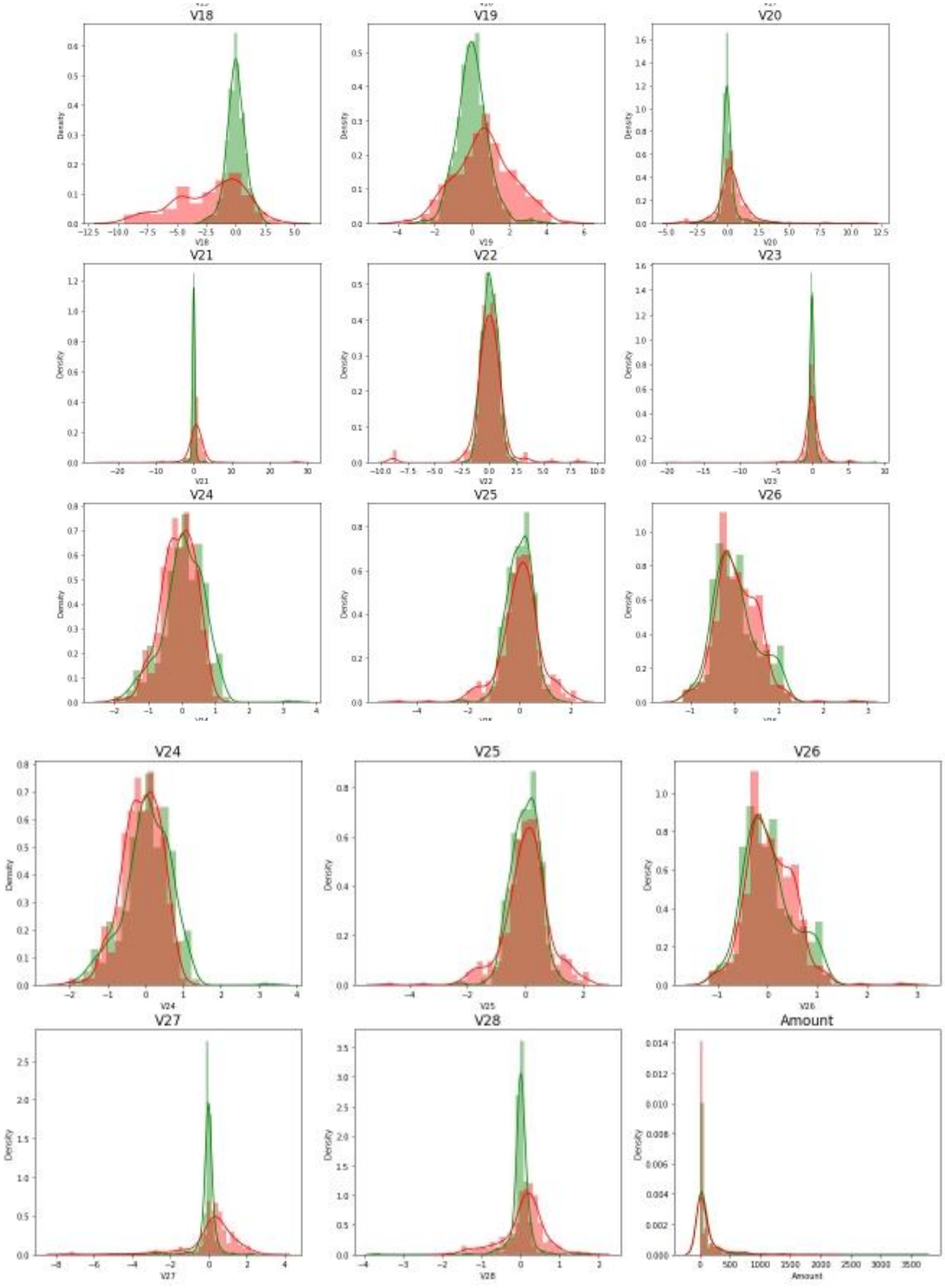


Figure 3.4: The histogram of variables from the dataset

This skewness may be seen in the histogram of variables in the dataset, as shown in figure 3.4. In this case, the values 0 and 1 correspond to honest and dishonest dealings. All transactions from all features are given, along with their skewness between 0 and 1. In graphical form, it shows us how legitimate and fraudulent transactions look across the board in the dataset.

Up-sampling and under-sampling are two methods that may be used to equalize an uneven dataset. We have employed under-sampling in our theses. With this strategy, the number of fraudulent transactions will remain the same while the number of legitimate transactions will be randomly drawn from class 0 but have the same total as class 1. In our situation, there have been 492 fraudulent transactions and 2,84,315 legitimate ones. As a result, the machine learning models compare legitimate transactions with fraudulent ones, using a sample of 492.

Now that the dataset is evenly distributed, we may divide it into test and train sets. As a rule of thumb, we used 20% of the data for testing and 80% for training. To create machine learning models, data must first be partitioned. K-Nearest Neighbor, Decision Tree, Support Vector Machine, and Logistic Regression are the four machine learning models employed here (The theoretical explanation of these models will be provided in the next section). Each model was developed by fitting actual train data onto preexisting frameworks. After the models were fitted, their ability to predict test data was used to rank them. Some measures, like Precision, Recall, F1-Score, ROC-AUC curve, and confusion matrix, are used to evaluate the accuracy of the predictions. The classification report and model score were both created using the score function from the sklearn package.

3.2 Bankruptcy Detection

In this section, we will discuss our second thesis project which is about bankruptcy. We will explain our working process step by step from dataset collection to the methodology we used. In the world of finance, spotting a potential bankruptcy is a hot issue. Many stakeholders, including shareholders, managers, and creditors, have a vested interest in knowing how likely it is that a company would go bankrupt. The machine learning techniques described in Section 3.1 were also put to use in this project.

3.2.1 Dataset

The dataset for this project has also been collected from the Kaggle website. When a firm cannot pay its obligations, bankruptcy is a real possibility. The Taiwan Economic Journal released a list of bankrupt enterprises from 1999 to 2009. The Taiwan Stock Exchange began trading on February 9, 1962. Taipei-based bank and 900+ companies. Most of the information is numerical, and it may be used to calculate the likelihood of bankruptcy. By using this dataset, we can properly address the issue of unbalanced data. Through the process of feature selection, we will identify the most useful characteristics. We may experiment with how to programmatically choose the best features and examine the impact on our model since the dataset has over 50 properties.

	Bankrupt?	Roa(C) Before Interest And Depreciation Before Interest	Roa(A) Before Interest And % After Tax	Roa(B) Before Interest And Depreciation After Tax	Operating Gross Margin	Realized Sales Gross Margin	Operating Profit Rate	Pre-Tax Net Interest Rate	After-Tax Net Interest Rate	Non-Industry Income And Expenditure/Revenue	...
0	1	0.370594	0.424389	0.405750	0.601457	0.601457	0.998969	0.796887	0.808809	0.302646	...
1	1	0.464291	0.538214	0.516730	0.610235	0.610235	0.998946	0.797380	0.809301	0.303556	...
2	1	0.426071	0.499019	0.472295	0.601450	0.601364	0.998857	0.796403	0.808388	0.302035	...
3	1	0.399844	0.451265	0.457733	0.583541	0.583541	0.998700	0.796967	0.808966	0.303350	...
4	1	0.465022	0.538432	0.522298	0.598783	0.598783	0.998973	0.797366	0.809304	0.303475	...

Net Income To Total Assets	Total Assets To Gnp Price	No-Credit Interval	Gross Profit To Sales	Net Income To Stockholder'S Equity	Liability To Equity	Degree Of Financial Leverage (Dfl)	Interest Coverage Ratio (Interest Expense To Ebit)	Net Income Flag	Equity To Liability
0.716845	0.009219	0.622879	0.601453	0.827890	0.290202	0.026601	0.564050	1	0.016469
0.795297	0.008323	0.623652	0.610237	0.839969	0.283846	0.264577	0.570175	1	0.020794
0.774670	0.040003	0.623841	0.601449	0.836774	0.290189	0.026555	0.563706	1	0.016474
0.739555	0.003252	0.622929	0.583538	0.834697	0.281721	0.026697	0.564663	1	0.023982
0.795016	0.003878	0.623521	0.598782	0.839973	0.278514	0.024752	0.575617	1	0.035490

Figure 3.5: First 5 Rows of the Dataset of Bankruptcy Detection

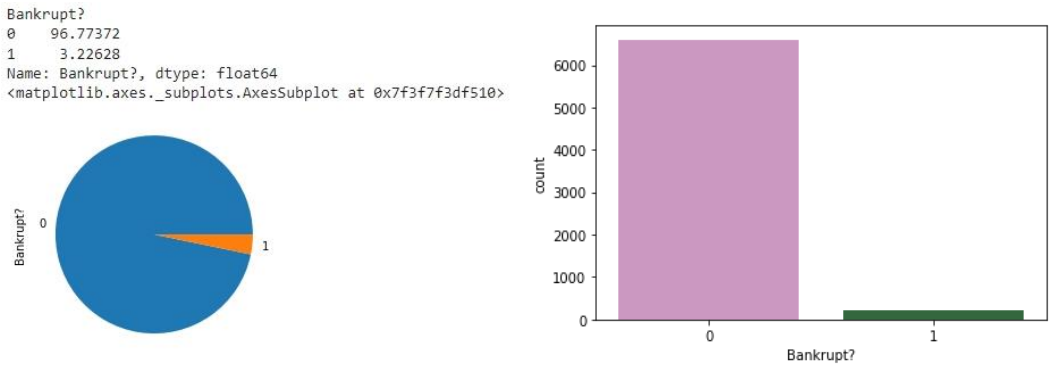


Figure 3.6: Unbalanced dataset and data distribution plot (Bankruptcy)

The dataset's distribution is seen in Figure 3.13. The primary column for making forecasts is labeled "bankrupt?" and it might be one of two values: 0 or 1. Class 0 has a larger numerical distribution than Class 1, indicating that the dataset is significantly imbalanced.

3.2.2 Methodology

The project's methodology and techniques are discussed here. In this part, we will go through the specifics of how to prepare a dataset for use with machine learning models, as well as how to train data in the model to make predictions. There are 96 columns in this dataset, however not all of them include information that is useful for our purpose of predicting insolvency. Accordingly, we first looked for the categorized rows. Additionally, we found three categorized columns: 'Bankrupt?', 'Liability-Asset Flag,' and 'Net Income Flag. Then we looked into those aisles. If a company's debts are higher than its assets, the "Liability-Asset Flag" will show a 1, and vice versa.

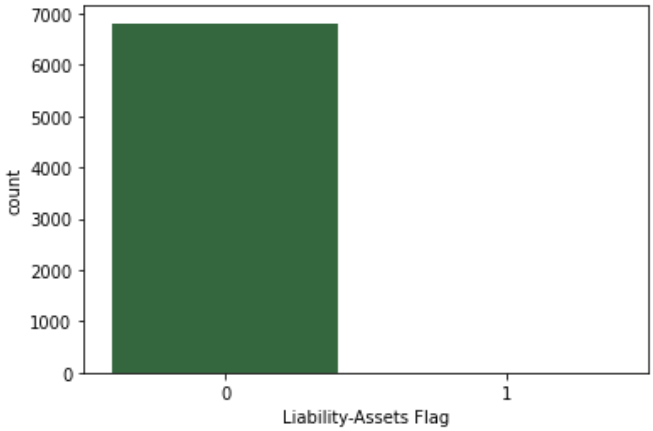


Figure 3.7: National-Asset Flag

W Figure 3.14 shows that, in most cases, an organization's assets exceed its liabilities. Bankrupt? was then evaluated for both scenarios.

```
Liability-Assets Flag  Bankrupt?
0                      0            6597
                       1             214
1                      1               6
                       0               2
dtype: int64
<matplotlib.axes._subplots.AxesSubplot at 0x7f3f7ee33050>
```

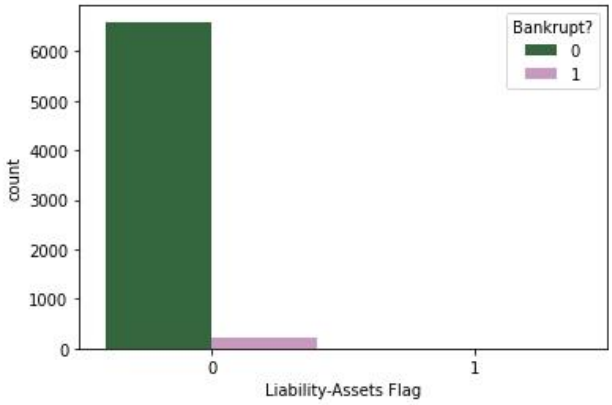
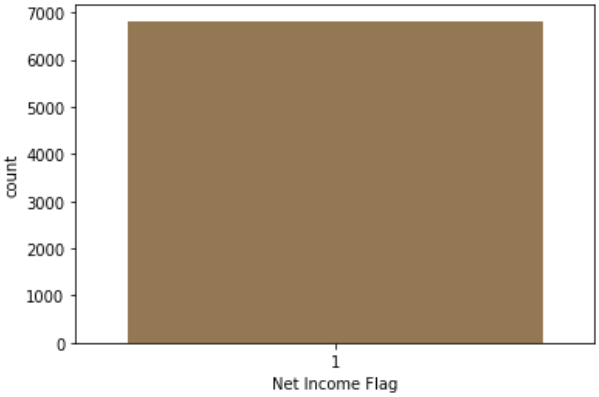


Figure 3.8: Liability-Assets Flag and Bankruptcy

Based on what we've seen, some businesses nonetheless go bankrupt despite having more assets than debts. Following this, we used the same procedure with the Net Income flag and analyzed the outcomes. All of the organizations had negative net income and lost records during the last two years, as shown by the findings.




```
Net Income Flag  Bankrupt?
1                0            6599
                 1             220
dtype: int64
<matplotlib.axes._subplots.AxesSubplot at 0x7f3f7ed0f850>
```

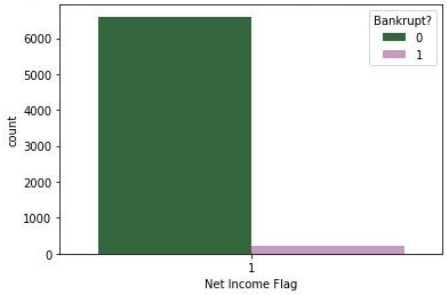


Figure 3.9: Net Income Flag and Bankruptcy

Then, we analyzed the relationships between each of the key factors and isolated the attributes most predictive of insolvency in the dataset.

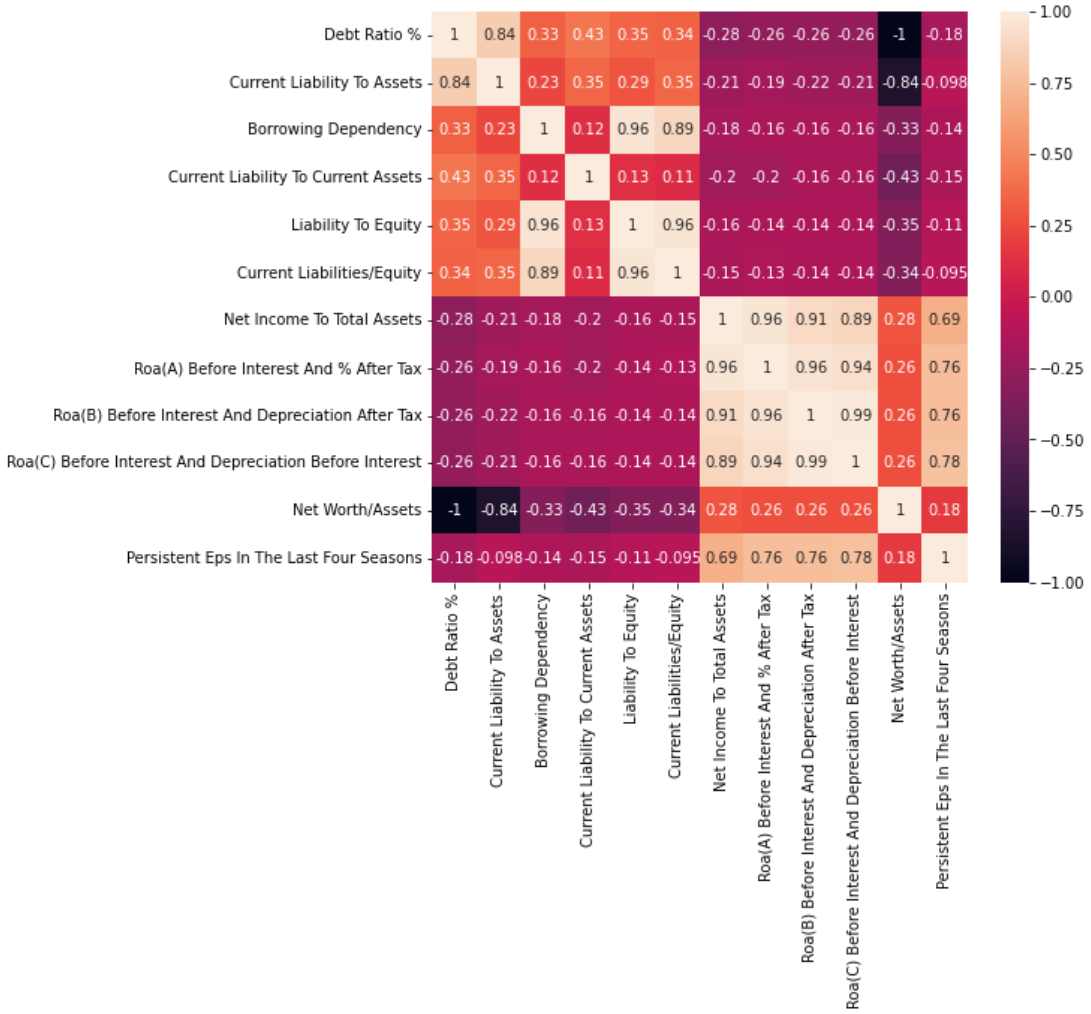


Figure 3.10: Correlation between important categories to find out the most relevant features

We addressed the extremely imbalanced data before applying the machine learning models to this dataset. We used an 80%:20% split for training and testing data. Only 20% of the data was used for testing purposes, while the remaining 80% was used for training the model. For the distribution of the remaining 20% of test data, we used a stratified K Fold Cross-Validation. Since we had to deal with more than 50 characteristics, we turned to "Randomized Search Cross-Validation," which improved our results for a wide range of them. After cleaning the data, we used three different machine learning models (K-Nearest Neighbor, Decision Tree, and Logistic Regression) to make a bankruptcy prediction based on the raw data.

CHAPTER 4
RESULTS AND DISCUSSIONS

The outcomes of the predictions are described in this section. The primary goal of comparing these models' prediction outputs is to identify the most effective model for identifying fraudulent credit card transactions in a given dataset. Machine learning involves making predictions about data sets by using a variety of models and algorithms. Parameters in the models and algorithms are used to assess whether or not they are suitable for a given dataset. Accuracy, precision, recall, the F1 score, etc., are all examples of such measures. Our dataset to make predictions from is very imbalanced and case-sensitive. Only "True-Positive" data will be used in our comparison of our model. For Credit Card Fraud Detection and Bankruptcy Detection, we need a model that properly predicts only fraudulent transactions. This is why we'll evaluate each model by contrasting its Recall value. Because of this, the model with the highest Recall values will be the best choice for our data. All the models we've used in our projects thus far have been described in detail in the preceding chapter, which focused on their respective methodologies. This section will solely focus on the findings and make comparisons between the models based on their prediction abilities.

4.1 Results Discussion: Credit Card Fraud Detection

Here we'll detail the findings from many anti-fraud models we trained. We have used a total of four models in our quest to identify credit card fraud, as indicated in an earlier chapter. Models like K-Nearest Neighbor, Decision Tree, Support Vector Machine, and Logistic Regression are used. The findings contain several values for the model's parameters, which are used to evaluate how well the model can locate a prediction. In this particular instance, we will be concentrating on the recall values since it is our goal to locate the specific fraudulent transactions with a high degree of precision. If any models have greater accuracy in predicting accurate fraudulent transactions,

then we will declare that model to be suitable for our project when we find one such model. As, for the outcomes of the various models, they are as follows:

classification Report

	precision	recall	f1-score	support
0	0.58	0.64	0.61	99
1	0.59	0.53	0.56	98
accuracy			0.58	197
macro avg	0.58	0.58	0.58	197
weighted avg	0.58	0.58	0.58	197

Figure 4.1: Prediction results of the K-Nearest Neighbor Model

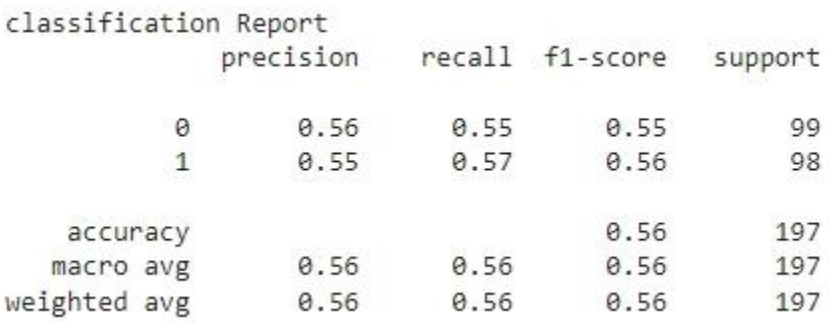
Parameter values for this instance of the K-nearest-neighbor model are shown in Figure 4.1. The metrics used here are Precision, Recall, and F1-Score. Prediction values regarding whether a transaction is fraudulent or not are available. However, as we covered before, we'll just focus on fraudulent transactions and give them the highest priority within the Recall criteria (53%). Precision is set at 59% and the F1-Score is set at 56%. As the Recall value is 53%, it can predict fraudulent transactions by 59% which is a not good prediction, we have to look for a better model.

classification Report

	precision	recall	f1-score	support
0	0.90	0.95	0.92	99
1	0.95	0.89	0.92	98
accuracy			0.92	197
macro avg	0.92	0.92	0.92	197
weighted avg	0.92	0.92	0.92	197

Figure 4.2: Prediction results of Decision Tree Model

Figure 4.2 depicts the parameters used in this specific implementation of the Decision Tree model. For this purpose, we make use of the Precision, Recall, and F1-Score measures. Precision (95), Recall (89), and F1-Score (92%) are their respective values. This model has a high Recall of 89%, indicating that it accurately predicts the occurrence of fraudulent transactions 89% of the time.

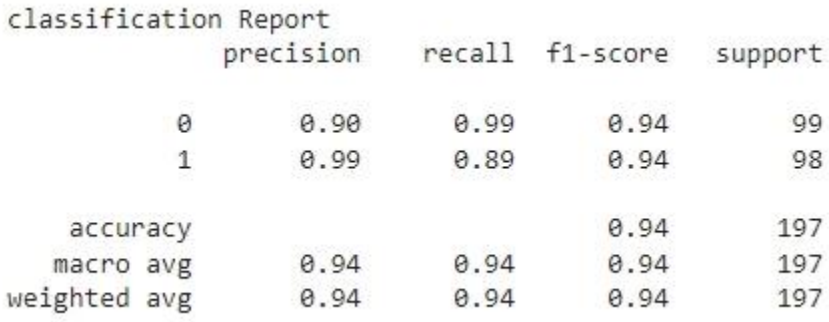


classification Report

	precision	recall	f1-score	support
0	0.56	0.55	0.55	99
1	0.55	0.57	0.56	98
accuracy			0.56	197
macro avg	0.56	0.56	0.56	197
weighted avg	0.56	0.56	0.56	197

Figure 4.3: Prediction result of Support Vector Machine (SVM) Model

The parameters of the support vector machine model are shown in the preceding figure 4.3. For the fraudulent transactions, the corresponding Precision, Recall, and F1-Score values are 55%, 57%, and 56%. Prediction values show that this specific model is only able to identify 57% of transactions as fraudulent, therefore it does not perform well enough to meet our goals.



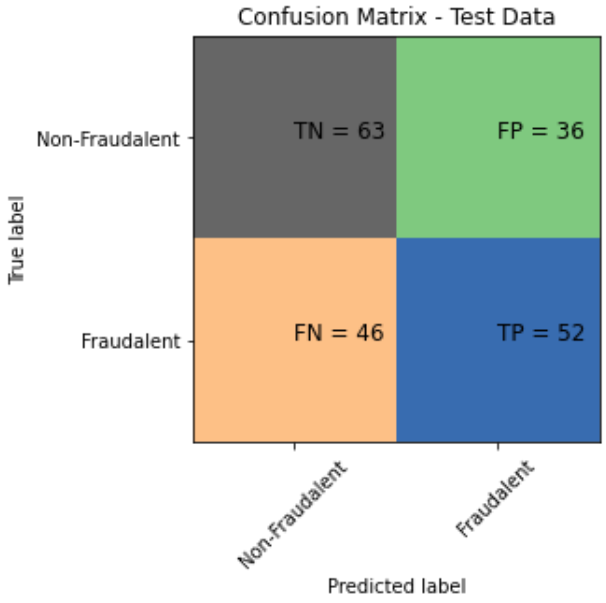
classification Report

	precision	recall	f1-score	support
0	0.90	0.99	0.94	99
1	0.99	0.89	0.94	98
accuracy			0.94	197
macro avg	0.94	0.94	0.94	197
weighted avg	0.94	0.94	0.94	197

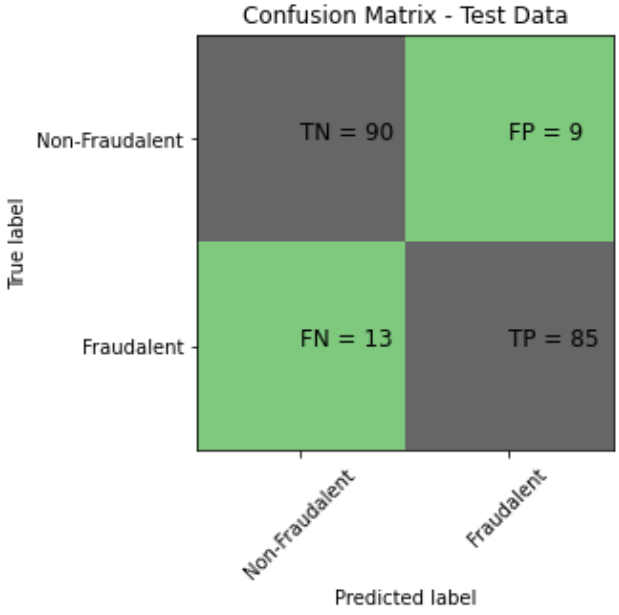
Figure 4.4: Prediction results of Logistic Regression Model

The values of LR model parameters are shown in Figure 4.4. That was the last model we used in our credit card fraud detection system. Our models have an 89% (Recall) rate of accuracy in predicting fraudulent transactions, as well as a 99% rate of Precision and a 94% F1-Score. The model has high predictive accuracy, as seen by its 89% recall rate for fraudulent transactions in our dataset. In the following sections, we will evaluate the performance of the top models in terms of prediction and choose the most appropriate model for our needs.

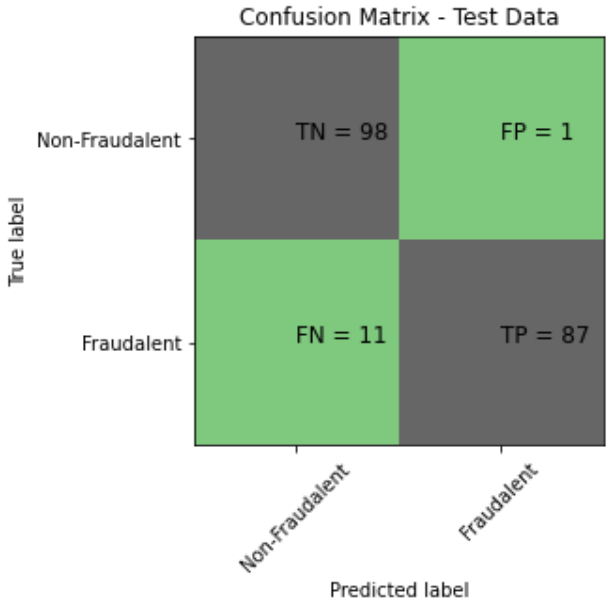
4.1.1 Models Evaluation: Credit Card Fraud Detection



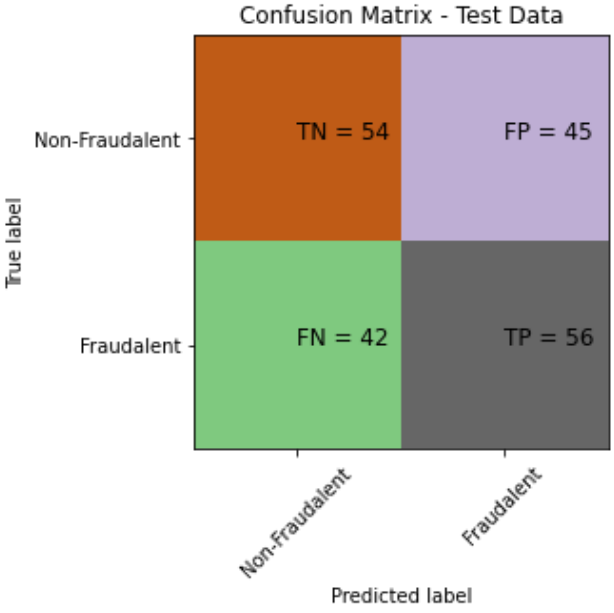
1) K-Nearest Neighbor



2) Decision Tree



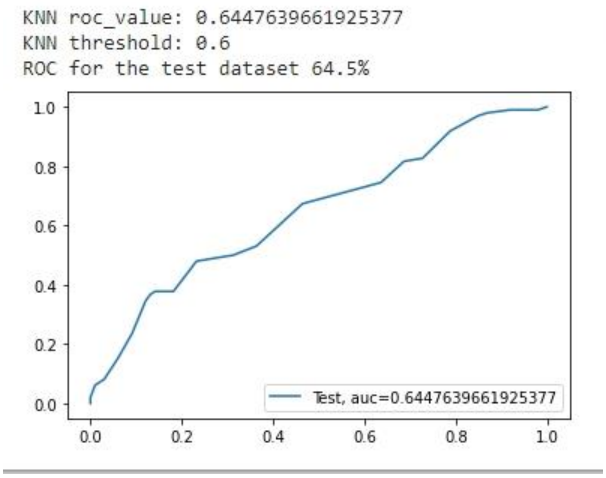
3) Logistic Regression



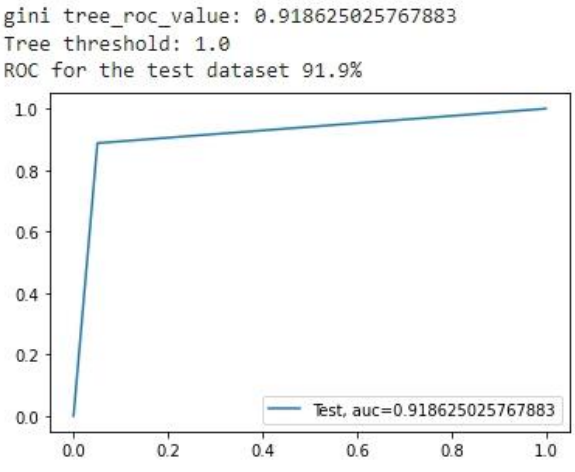
4) Support Vector Machine

Figure 4.5: Confusion Matrix (Credit Card Fraud Detection)

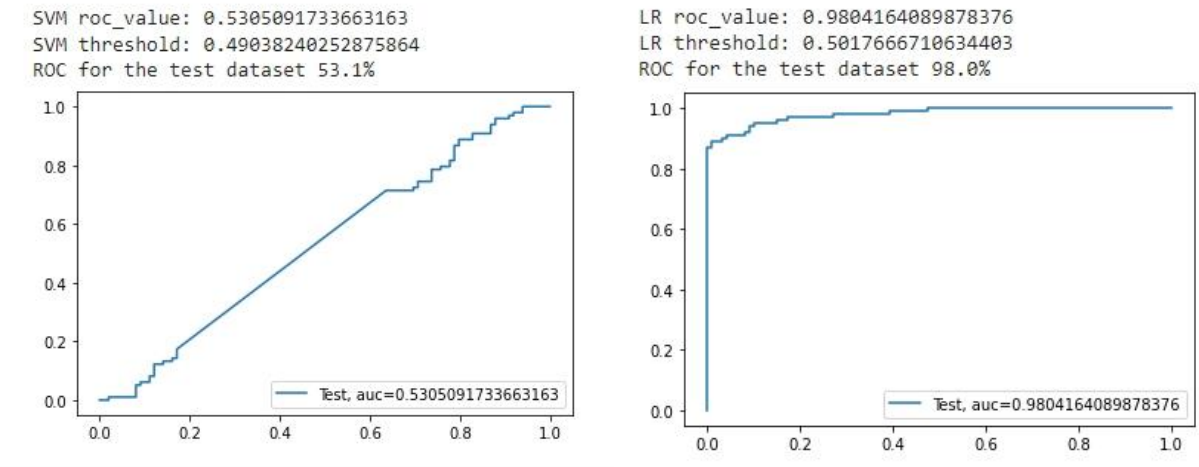
Figure 4.5 displays the confusion matrices of all the models we have tested for detecting credit card fraud. Understanding how well a model performed may be deduced from its confusion matrix. A confusion matrix is a useful tool for illustrating the detection outcome of any given model or algorithm. To understand the confusion matrix, it is necessary to know the True positive, False positive, True negative, and False negative values. This includes four models and their respective confusion matrices., 1) represents the confusion matrix of the K-Nearest Neighbor Model. From the matrix, we see the values of TP, FP, TN, and FN. For K-Nearest Neighbor, TP= 52, TN= 63, FN= 40, and FP= 25. This explains, among all the transactions each time the model can detect 58 transactions exactly to be fraudulent and 74 to be exactly not fraudulent. Rest two values explain, for FN=40, 40 transactions are not fraudulent but there can be fraudulent transactions and FP= 25 means 25 transactions to be fraudulent but could be not fraudulent. 2) explains the confusion matrix of the Decision Tree model, here TP= 85, TN= 54, FP= 9, and FN=13. We will discuss only True Positive and True Negative values as it matters the most. Here the exactly fraudulent and exactly not fraudulent transaction predictions are 85 and 54 respectively. For Logistic Regression Model which is describes in 3), has the confusion matrix values of TP= 87, TN= 98, FN= 11, and FP= 1. Since it has a high accuracy rate for distinguishing between fraudulent and legitimate transactions, and a low error rate, this model outperforms its competitors. The confusion matrix 4) shows the results of the Support Vector Machine Model. It has the value of TP= 56, TN= 54, FN= 42, and FP= 45. The model's performance is subpar since it makes about the same number of accurate and incorrect predictions about whether a given transaction is fraudulent or not.



1) ROC of KNN



2) ROC of Decision Tree



3) ROC of SVM 4) ROC of LR

Figure 4.6: The ROC and AUC curve value of all Models

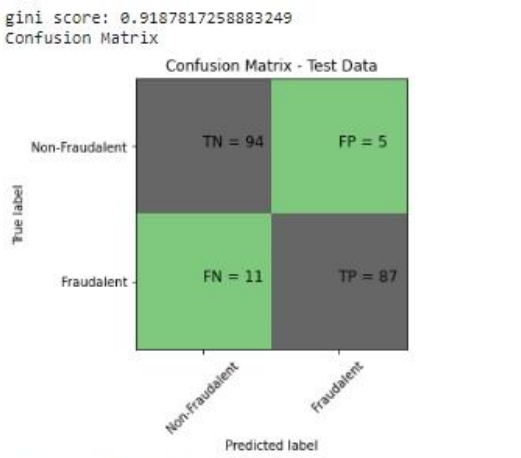
The AUC-ROC curve is an alternative method for validating the quality of a model. Both Area Under Curve and Receiver Characteristics Operator (AUC and ROC) are measures of performance (ROC). The ROC curve is a measure of performance for issues of binary classification. It's a curve that separates the signal from the noise based on the threshold percentage of correct responses (TPR) vs false positive rates (FPR) at different levels. The Area Under the Curve (AUC) summarizes the Receiver Operating Characteristic (ROC) curve and measures the classifier's ability to discriminate across classes. The better the model can tell the difference between the positive and negative classes, the higher the area under the curve (AUC). In the ideal case, when AUC=1, a classifier can accurately separate all instances into the Positive and Negative classes. In contrast, if the AUC is zero, the classifier would incorrectly label all negative examples as positive and all positive examples as negative. If the AUC is greater than 0.5, then the classifier is likely to correctly separate the positive and negative values. That's because the classifier has a higher detection rate for True Positives and True Negatives than it does for False Negatives and False Positives. If the area under the receiver operating characteristic curve (AUC) is 0.5, then the classifier cannot discriminate between Positive and Negative class points. This might indicate that the classifier is incorrectly estimating the data's category, or that it is consistently wrong.

Therefore, a classifier's capacity to separate positive and negative classes increases as its AUC value rises.

Figure 4.6 shows the ROC-AUC value that will be used to assess the quality of our models. Figure 1 displays the ROC-AUC value for K-Nearest Neighbor, which at 64% indicates it does an excellent job of differentiating between positive and negative examples. The ROC-AUC score of 91% for the Decision Tree model indicates that it can reliably separate positive and negative class points. Finally, we find (3) Support Vector Machine, which has a ROC-AUC value of just 53% but a good likelihood of distinguishing between the positive and negative class points. If you look at the area under the receiver operating characteristic curve, however, Logistic Regression stands out as the best option.

4.1.2 Comparing the models: Credit Card Fraud Detection

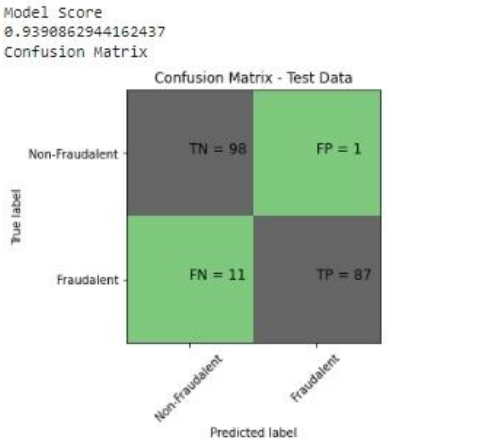
Here, we'll evaluate the two best models we considered for our credit card fraud detection project and choose a winner. All models will be compared using metrics like ROC-AUC and model scores to determine which one is best. Decision Tree and Logistic Regression fared the best in our evaluations, which we completed in the preceding part. Our next step is to contrast the two models.



1) Confusion Matrix & Classification Report of DT

classification Report

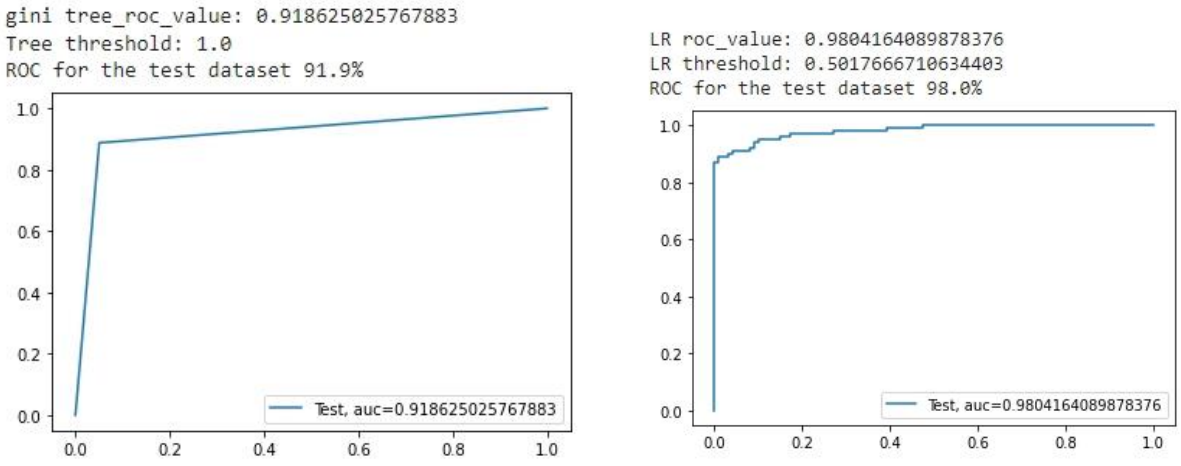
	precision	recall	f1-score	support
0	0.90	0.95	0.92	99
1	0.95	0.89	0.92	98
accuracy			0.92	197
macro avg	0.92	0.92	0.92	197
weighted avg	0.92	0.92	0.92	197



3) Confusion Matrix & Classification Report of LR

classification Report

	precision	recall	f1-score	support
0	0.90	0.99	0.94	99
1	0.99	0.89	0.94	98
accuracy			0.94	197
macro avg	0.94	0.94	0.94	197
weighted avg	0.94	0.94	0.94	197



2) ROC-AUC of DT

4) ROC-AUC of LR

Figure 4.7: Comparing models

All the models were compared in the previous section in terms of precision, recall, and ROC-AUC. We compared all of the models and settled on two that performed well. We will now evaluate each mode against one another to see which is most suitable for the data we have. Figure 4.7 displays the Decision Tree model's statistics in panels 1) and 2), whereas the logistic regression model is shown in panels 3 and 4. Initially, both models had the same Recall (the prediction of recognizing the precise fraudulent values) at 89%. However, Logistic Regression has a higher model score (93% vs. 91%) than Decision Tree. The Logistic Regression model offers a higher ROC-AUC value (98%) for differentiating between true and negative class points than the Decision Tree model (91%). This leads us to the inevitable conclusion that Logistic Regression is the superior model.

Now, to make a conclusive comparison, we have crafted a table in which we altered the sample population distribution. Up until this point, the ratio of fraudulent to legitimate transactions was even, with 50% of all transactions being fraudulent and the remaining 50% being legitimate. However, to conduct further evaluation of the models, we have recently altered the distribution of the samples. Because of the severely uneven dataset, the number of fraudulent transactions is a very significant amount lower than the number of legitimate transactions. We did not change the fraudulent transaction that had been going on, but we did increase the sample size of the legitimate

transaction so that we could compare it to the fake data. It resulted in a change to the distribution of the samples and made it possible for us to analyze the findings of the model that we had used.

Samples Distribution	Model Name	Transaction 0 = Not Fraud 1 = Fraud	Precision	Recall	F1-Score	Support	Model Score
For equal distribution of normal (50%) and fraudulent (50%) transactions	K-Nearest Neighbor	0	58%	64%	61%	99	63%
		1	59%	53%	56%	98	
	Decision Tree	0	90%	95%	92%	99	91%
		1	95%	89%	92%	98	
	Support Vector Machine	0	56%	55%	55%	99	55%
		1	55%	57%	56%	98	
	Logistic Regression	0	90%	99%	94%	99	93%
		1	99%	89%	94%	98	
For distribution of normal (60%) and fraudulent (40%) transactions	K-Nearest Neighbor	0	68%	83%	75%	148	66%
		1	62%	41%	49%	98	
	Decision Tree	0	91%	91%	91%	114	90%
		1	90%	90%	90%	99	
	Support Vector Machine	0	51%	54%	53%	114	47%
		1	43%	39%	41%	99	
	Logistic Regression	0	92%	96%	94%	114	92%
		1	95%	90%	92%	99	
For distribution of normal (70%) and fraudulent (30%) transactions	K-Nearest Neighbor	0	75%	90%	82%	230	72%
		1	56%	32%	41%	98	
	Decision Tree	0	97%	94%	95%	230	93%
		1	87%	93%	90%	98	
	Support Vector Machine	0	69%	67%	68%	230	55%
		1	28%	31%	29%	98	
	Logistic Regression	0	96%	98%	97%	230	96%
		1	96%	91%	93%	98	

Table 4.1: Comparison of Models for a different distribution of samples.

Table 4.1, compares all the models for different distributions of samples. The Logistic Regression model achieves the best Recall values as well as the model score of any of the other models tested. Hence, the Logistic model is performing better in our case.

4.2 Results Discussion: Bankruptcy Detection

Here, we provide the findings and address our second main issue, the identification of bankruptcies. K-Nearest Neighbor, Decision Tree, and Logistic Regression are the three models we've used here. Similar to the last project, we will track the values of the various parameters, with an eye on the Recall numbers as they are what ultimately decide whether or not a bank would fail as an institution. According to the Recall, we will utilize the same methods as before to provide an accurate projection of how many businesses will go bankrupt. Then, we'll evaluate each candidate model and choose the one that works best for this task. Finally, we'll draw some conclusions about the relative merits of the two projects' models at the end of this chapter. The results are as follows:

Evaluation Of Models

Random Model Evaluation

	precision	recall	f1-score	support
0	1.00	0.98	0.99	1313
1	0.67	0.94	0.78	51
accuracy			0.98	1364
macro avg	0.83	0.96	0.89	1364
weighted avg	0.99	0.98	0.98	1364

Figure 4.8: Classification Report of KNN

Figure 4.8 depicts the K-Nearest Neighbor prediction report. The categorization report displays the model-predicted values of each parameter for both insolvent and solvent institutions, respectively. Precision = 67%%, Recall = 94%, and F1-Score = 78% all provide excellent

explanations for the prediction values of bank failure. As bank failure prediction is our primary focus, we are simply taking the predictive values of bank failure into account. This model will predict bankruptcy by 94%. It is a very good prediction. When this occurs, we will only assess our models based on their Recall values. We will follow the same procedure as previously, observing and comparing the outcomes of each model before selecting the one that works best for this particular project.

Evaluation Of Models

Random Model Evaluation

	precision	recall	f1-score	support
0	1.00	0.84	0.91	1313
1	0.18	0.90	0.31	51
accuracy			0.85	1364
macro avg	0.59	0.87	0.61	1364
weighted avg	0.97	0.85	0.89	1364

Figure 4.9: Classification Report of DT

Prediction values and classification reports for bank failure are shown in Figure 4.9, with a Precision of 18%, Recall of 90%, and F1-Score of 31%. According to the Recall value, this Decision Tree model is 90% accurate in predicting which banks would fail in the future. It seems like our initiative has a decent chance of succeeding.

Evaluation Of Models

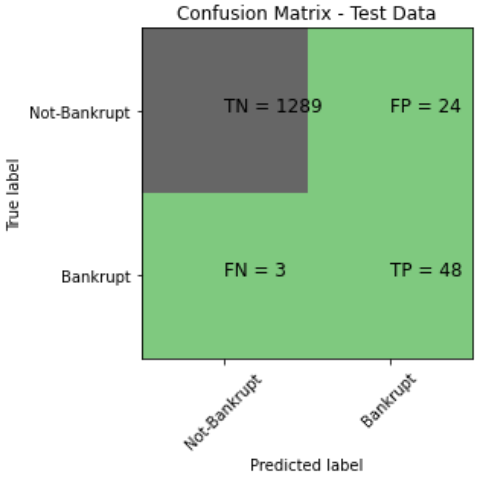
Random Model Evaluation

	precision	recall	f1-score	support
0	0.99	0.89	0.94	1313
1	0.23	0.82	0.36	51
accuracy			0.89	1364
macro avg	0.61	0.86	0.65	1364
weighted avg	0.96	0.89	0.92	1364

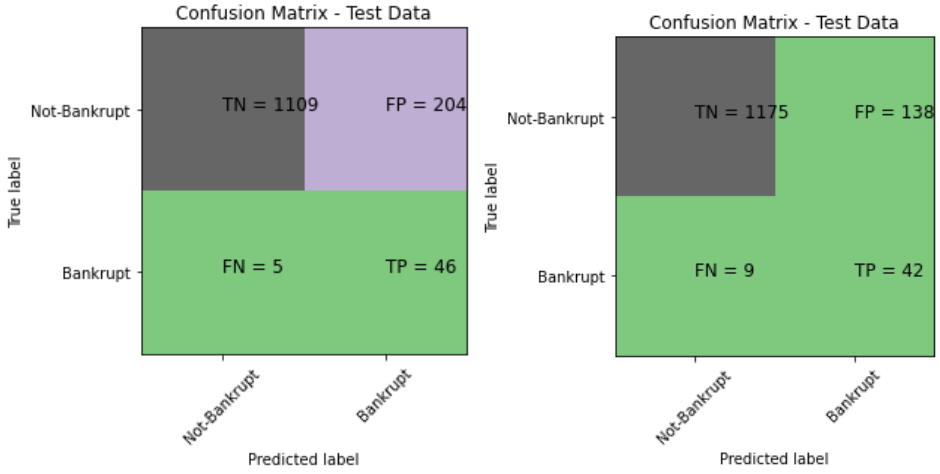
Figure 4.10: Classification Report of LR

Logistic Regression served as the project's final model. All of the expected values for insolvency from a categorization report are shown in Figure 4.10. Similar to before, it displays various parameter values. F1-Score is 36%, Recall is 82%, and Precision is 23%. Recall claims that this model fares about as well as the median of the other two it compares itself to. The success rate of this model in predicting insolvency is 82%. The efficiency of both preceding models exceeds that of this one.

4.2.1 Model Evaluation: Bankruptcy Detection



1) KNN model



2) DT model 3) LR model

Figure 4.11: Confusion Matrix (Bankruptcy Detection)

Here, we'll compare the accuracy of all the models that were used to make a decision. As such, we have taken into account the confusion matrix, which effectively characterizes the performance of machine learning models, every time. Above, in Figure 4.11, you can see how well each model predicts and how accurate its predictions are. With these numbers, we can determine every possible outcome of each parameter. The True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) components make up the K-Nearest Neighbor model's confusion matrix, which we may first examine (FN). The accurately anticipated negative numbers are the true negatives. True positive values, on the other hand, accurately predicted good outcomes. The accuracy with which TN and TP values can be predicted is a measure of the model's usefulness. As can be seen above, the TP= 48 and TN= 1289 for the K-nearest neighbor (1) model. With this information at hand, the model has a 48 percent chance of properly predicting a bank's failure and a 129% chance of correctly predicting that it would succeed. Now we know that TP = 46 and TN = 1109 correspond to the Decision Tree model (2). Therefore, the model is right 46 times out of 1,109 times when predicting insolvency. Logistic Regression (3), the final model, predicts TP = 42 and TN = 1175. Thus, this model successfully predicts both positive (42 times) and negative (1175 times) values. Now that we've compared these three models to others, we can see that the K-Nearest Neighbor (1) model has the highest predictive accuracy.

	Algorithm	Model Score	Precision	Recall	F1 score	ROC-AUC score
0	K Nearest Neighbour	98.02%	0.67	0.94	0.78	0.96
1	Logistic Regression	89.37%	0.24	0.82	0.37	0.86
2	DecisionTree Classifier	84.68%	0.18	0.90	0.31	0.87

Figure 4.12: Comparison of models (Bankrupt Detection)

Figure 4.12 displays the ROC-AUC score, another method of comparison described in the introduction to credit card fraud detection. The ROC-AUC value indicates that the K-Nearest Neighbor method has the highest performance since it can more accurately differentiate between class values of zero and one. The stronger the model, the higher the ROC-AUC score should be. K-Nearest Neighbor is the optimal solution to this specific issue.

CHAPTER 5

CONCLUSION AND FUTURE WORKS

This section of the thesis will summarize our work to this point and highlight our accomplishments. In this section, we'll talk about the challenges we encountered while developing this project, as well as our plans for the future. Having used machine learning for two projects—"Credit Card Fraud Detection" and "Bankruptcy Detection"—we have gleaned a great deal of information. These are written, and the results are as follows:

5.1 Conclusion

- Our thesis makes use of machine learning, a subfield of artificial intelligence that focuses on developing programs with the ability to learn on their own, both on an ongoing basis and in response to specific stimuli, seemingly without being programmed. As part of our thesis, we have implemented the aforementioned two machine learning projects. The initial step in machine learning is always to gather a cleaned and organized dataset. We now know of numerous reputable places to look for such datasets, and we have done so for a variety of educational and scientific objectives. Benefits await us in the academic sector of the future. The next critical stage was finalizing our project's selection of algorithms. We collected data and then extensively explored several algorithms. Moreover, we gained knowledge of the various algorithms' capabilities on various datasets. When it comes to our work, we have compiled two such very asymmetrical data sets. The dataset was also binary class supervised. Based on this need, we have selected four algorithms capable of processing supervised data. K-Nearest Neighbor, Logistic Regression, Support Vector Machine, and Decision Tree are only a few of the techniques that have been developed.
- Because our dataset was so skewed, we had to resort to sampling to manage it. Up-sampling is one way to collect data, whereas under-sampling is the other. The under-sampling technique was applied. Here, we took on the challenge of rectifying an

imbalanced dataset and succeeded. Our algorithm for detecting credit card fraud has been further evaluated using data from a variety of sample sizes. The outcomes were reviewed and compared after careful observation. We also utilized filter-based selection, correlating the variables in the Bankruptcy Detection dataset to eliminate superfluous ones and extract the most important feature. We waited at the edge of a threshold and recorded just the results right at the edge.

- We were able to determine the model that was the best match for each of our projects by using the Recall parameters.

5.2 Future Works

Several aspects may be improved upon and included in our thesis. We will be able to get better outcomes if we do this. Because datasets may be sampled in a variety of ways, some of which are more effective than others, further research must be conducted for each dataset. There are a few other ways of selection that may be utilized, and some of them could provide superior outcomes. Additionally, some additional models have to be put into action so that the outcomes may be evaluated and contrasted. Cross-validation is a crucial component that prevents the model from being over-adjusted in any way. In addition to that, it calculates the total errors. In addition, the management of big data, also known as an enormous quantity of information, may be accomplished via the use of machine learning. The whole process of generating data, storing data, accessing data, and evaluating data may be automated via the use of machine learning. As a result, our thesis might serve as food for thought when considering the larger picture of the future of big data.

REFERENCE

References

[1] Baştanlar, Yalin, and Mustafa Özuysal. “Introduction to Machine Learning | SpringerLink.” *Introduction to Machine Learning | SpringerLink*, 11 Nov. 2013, link.springer.com/protocol/10.1007/978-1-62703-748-8_7.

[2] Hastie, T., Tibshirani, R. and Friedman, J. (2009) Boosting and Additive Trees. In: The Elements of Statistical Learning, Springer, New York, 337-387.

[3] Jensen, D. (1997) Prospective Assessment of AI Technologies for Fraud Detection: A Case Study. In: The AAAI Workshop on AI Approaches to Fraud Detection and Risk Management, AAAI Press, Palo Alto, CA, 34-38.

[4] Randhawa, K., Loo, C.K., Seera, M., Lim, C.P. and Nandi, A.K. (2018) Credit Card Fraud Detection Using AdaBoost and Majority Voting. IEEE Access, 6, 14277-14284. https://doi.org/10.1109/ACCESS.2018.2806420

[5] Ryan, J., Lin, M.-J. and Miikkulainen, R. (1998) Intrusion Detection with Neural Networks. In: Advances in Neural Information Processing Systems, MIT Press, Cambridge, MA, 943-949.

[6] Worldpay (2015) Global Payments Report 2015. http://offers.worldpayglobal.com/rs/850-JOA-856/images/GlobalPaymentsReportNov2015

[7] HSN Consultants (2016) The Nilson Report. https://www.nilsonreport.com/upload/content_promo/The_Nilson_Report_10-17-2016.pdf

[8] Stolfo, S., Fan, D.W., Lee, W., Prodromidis, A. and Chan, P. (1997) Credit Card Fraud Detection Using Meta-Learning: Issues and Initial Results. AAAI-97 Workshop on Fraud Detection and Risk Management, Providence, RI, 27-28 July 1997, 83-90.

[9] Wang, S. (2010) A Comprehensive Survey of Data Mining-Based Accounting-Fraud Detection Research. 2010 International Conference on Intelligent Computation Technology and

Automation, Changsha, 11-12 May 2010, 50-53. https://doi.org/10.1109/ICICTA.2010.831

[10] Baştanlar, Yalin, and Mustafa Özuysal. “Introduction to Machine Learning | SpringerLink.” *Introduction to Machine Learning | SpringerLink*, 11 Nov. 2013, link.springer.com/protocol/10.1007/978-1-62703-748-8_7.

[11] “ML | What Is Machine Learning ? - GeeksforGeeks.” *GeeksforGeeks*, 1 May 2018, www.geeksforgeeks.org/ml-machine-learning.

[12] *"Introduction to AI Part 1". Edison. 2020-12-08. Retrieved 2020-12-09.*

[13] *Langley, Pat (2011). "The changing science of machine learning". Machine Learning. **82** (3): 275–279. doi:10.1007/s10994-011-5242-y.*

[14] Alpaydin, Ethem (2010). Introduction to Machine Learning. MIT Press. p. 9. ISBN 978-0-262-01243-0.

[15] Bzdok, Danilo; Altman, Naomi; Krzywinski, Martin (2018). "Statistics versus Machine Learning". *Nature Methods*. **15** (4): 233–234. doi:10.1038/nmeth.4642. PMC 6082636. PMID 30100822.

[16] *Michael I. Jordan (2014-09-10). "statistics and machine learning". reddit. Retrieved 2014-10-01.*

[17] Cornell University Library (August 2001). "Breiman: Statistical Modeling: The Two Cultures (with comments and a rejoinder by the author)". *Statistical Science*. **16** (3). doi:10.1214/ss/1009213726. S2CID 62729017. Retrieved 8 August 2015.

[18] *Gareth James; Daniela Witten; Trevor Hastie; Robert Tibshirani (2013). An Introduction to Statistical Learning. Springer. p. vii.*

[19] “An Introduction to Machine Learning - GeeksforGeeks.” *GeeksforGeeks*, 24 Aug. 2017, www.geeksforgeeks.org/introduction-machine-learning.

[20] “What Is Machine Learning and Why Is It Important?” *SearchEnterpriseAI*, 1 Mar. 2021, www.techtarget.com/searchenterpriseai/definition/machine-learning-ML.

[21] “Reinforcement Learning - GeeksforGeeks.” *GeeksforGeeks*, 25 Apr. 2018, www.geeksforgeeks.org/what-is-reinforcement-learning.

[22] Alpaydin, Ethem. “Introduction to Machine Learning - Ethem Alpaydin.” *Google Books*, books.google.com.bd, books.google.com.bd/books?id=NP5bBAAAQBAJ&printsec=frontcover&dq=introduction+to+machine+learning&hl=en&sa=X&ved=2ahUKEwjL5NCk4Kb6AhUrRWwGHVw9AzQQ6AF6BAgLEAI.

[23] “How Machine Learning Works - DataRobot AI Cloud.” *DataRobot AI Cloud*, 4 July 2022, www.datarobot.com/blog/how-machine-learning-works.

[24] Arya, Nisha. “K-Nearest Neighbors in Scikit-Learn - KDnuggets.” *KDnuggets*, www.kdnuggets.com/2022/07/knearest-neighbors-scikitlearn.html. Accessed 26 Sept. 2022.

[25] “What Is the K-Nearest Neighbors Algorithm? | IBM.” *What Is the K-Nearest Neighbors Algorithm? | IBM*, www.ibm.com/topics/knn#:~:text=The%20k%2Dnearest%20neighbors%20algorithm%2C%20also%20known%20as%20KNN%20or,of%20an%20individual%20data%20point. Accessed 26 Sept. 2022.

[26] “The Complete Guide to Machine Learning Steps.” *Simplilearn.Com*, www.simplilearn.com/tutorials/machine-learning-tutorial/machine-learning-steps. Accessed 26 Sept. 2022.

[27] “What Is Machine Learning? Definition and How Its Works.” *Intellipaat Blog*, 4 Jan. 2017, intellipaat.com/blog/what-is-machine-learning.

[28] Education, IBM Cloud. “What Is Machine Learning?” *What Is Machine Learning? | IBM*, 15 July 2020, www.ibm.com/cloud/learn/machine-learning.

[29] “Credit Card Fraud Detection.” *Credit Card Fraud Detection | Kaggle*, www.kaggle.com/datasets/mlg-ulb/creditcardfraud. Accessed 26 Sept. 2022.

[30] Chauhan, Nagesh Singh. “Decision Tree Algorithm, Explained - KDnuggets.” *KDnuggets*, www.kdnuggets.com/2020/01/decision-tree-algorithm-explained.html. Accessed 26 Sept. 2022.

[31] “Machine Learning Decision Tree Classification Algorithm - Javatpoint.” *Www.Javatpoint.Com*, www.javatpoint.com/machine-learning-decision-tree-classification-algorithm. Accessed 26 Sept. 2022.

[32] “Support Vector Machine (SVM) Algorithm - Javatpoint.” *Www.Javatpoint.Com*, www.javatpoint.com/machine-learning-support-vector-machine-algorithm. Accessed 26 Sept. 2022.

[33] “SVM Machine Learning Tutorial – What Is the Support Vector Machine Algorithm, Explained with Code Examples.” *freeCodeCamp.Org*, 1 July 2020, www.freecodecamp.org/news/svm-machine-learning-tutorial-what-is-the-support-vector-machine-algorithm-explained-with-code-examples.

[34] “Logistic Regression in Machine Learning - Javatpoint.” *Www.Javatpoint.Com*, www.javatpoint.com/logistic-regression-in-machine-learning. Accessed 26 Sept. 2022.

[35] “Machine Learning - Logistic Regression.” *Machine Learning - Logistic Regression*, www.tutorialspoint.com/machine_learning_with_python/machine_learning_with_python_classification_algorithms_logistic_regression.htm. Accessed 26 Sept. 2022.

[36] “Logistic Regression for Machine Learning | Capital One.” *Capital One*, www.capitalone.com/tech/machine-learning/what-is-logistic-regression. Accessed 26 Sept. 2022.