Expertise
and insight
for the future

Tikaram Acharya

Supervised Machine Learning

Accuracy Comparison of Different Algorithms on Sample Data

Metropolia University of Applied Sciences

Bachelor of Engineering

Information Technology

Bachelor’s Thesis

11 January 2021

Author Title Number of Pages Date	Tikaram Acharya Supervised Machine Learning: Accuracy Comparison of Different Algorithms on Sample Data 39 pages 11 January 2021
Degree	Bachelor of Engineering
Degree Programme	Information Technology
Professional Major	Software Engineering
Instructors	Janne Salonen, Head of School

The purpose of the final year project was to discuss the theoretical concepts of the analytical features of supervised machine learning. The concepts were illustrated with sample data selected from the open-source platform. The primary objectives of the thesis include explaining the concept of machine learning, presenting supportive libraries, and demonstrating their application or utilization on real data. Finally, the aim was to use machine learning to determine whether a person would pay back their loan or not. The problem was taken as a classification problem.

To accomplish the goals, data was collected from an open data source community called Kaggle. The data was downloaded and processed with Python. The complete practical part or machine learning workflow was done in Python and Python's libraries such as NumPy, Pandas, Matplotlib, Seaborn, and SciPy.

For the given problem on sample data, five predictive models were constructed and tested with machine learning with different techniques and combinations. The results were highly accurately matched with the classification problem after evaluating their f1-score, confusion matrix, and Jaccard similarity score. Indeed, it is clear that the project achieved the main defined objectives, and the necessary architecture has been set up to apply various analytical abilities into the project.

Keywords	Python, NumPy, Scikit-Learn, Matplotlib, Predictive Analysis

Contents

List of Abbreviations

FDIC	Federal Deposit Insurance
KNN	K Nearest Neighbors
IPO	Initial Public Offering
CSV	Comma-separated Value
PC	Personal Computer
UCI	Unique Client Identifier
SVM	Support Vector Machine

1 Introduction

LendingClub is an American club, the headquarters being located in San Francisco, California. LendingClub connects people who needed money (borrower) with people who want to give money as a loan (investor). LendingClub provides services to borrowers to create personal unsecured loans between $1,000 and $40,000. Loan lenders can find loan listings on lendinclub.com and set loan plans to invest based on supplied information about the loan taker, the purpose of the loan, credit score of borrowers, the income of the loan taker (monthly or yearly), and the duration of the loan (number of years). During the transaction or loan completion period, the lending club collects money by charging borrowers an organization fee and investors a service fee. On the other hand, LendingClub also decides the customer's loans for automobile transactions through WebBank, an FDIC-insured and state-chartered industrial bank. The loans are not granted by investors but are assigned to other financial institutions. [1.]

LendingClub is the most promising company in the United States and the club became the largest technology IPO of 2014 in the US [1.]. In some cases, it may be a risk to lend money to people without reviewing the history of security during the transaction completion and clearance period. A sample dataset of LendingClub’s transaction period from 2007-2010 was taken from an open-source platform called Kaggle. The practical part of the final project completely focuses on the potentiality of people and whether they pay back their loan fully or not. Investors want to give money to people who will most likely pay back their loans.

The final year project was carried out to show how machine learning algorithms and methods can be applied in the case of LendingClub and a real-life scenario. The Python language was used in the project. The goal of the project was to build predictive machine learning models, which are described in section 3.4. Section 2 details the concept and discusses the importance of machine learning. The section also explains the business approach linked to machine learning. Section 3 is very important as it thoroughly presents machine learning from a theoretical viewpoint. The section also illustrates the types of machine learning that exist and the types of models that are used to describe the accuracy of the predictive models made. In contrast, section 4 illustrates the practical part of a machine learning workflow. Moreover, chapter 4 also explains the data and data source, data implementation, data handling, data preparation, model formation, and model evaluation with a proper theoretical and practical explanation. Finally, in the last two sections, the results are discussed in detail, and conclusions are drawn in sequential order.

2 Introduction Machine Learning

Machine learning is a subset of Artificial Intelligence related to computational statistics. Machine learning does the same task as humans do. For example, earlier some emails had to be marked as spam but nowadays some email is directly marked as spam. The reason behind the case is that the application of machine learning has already classified some emails as spam based on their content structure. Machine learning consists of algorithms and methods to teach computers to understand the features of objects. Then computers build predictive models, prediction, and degree of performance. In another way, machine learning is one of the most popular fields of computer science which supports computer programs to analyze large amounts of data using different machine learning models to generate a prediction on given problems. Machine learning has also been implemented in large companies such as Netflix. Machine learning is powerful and is easily available to many users to use. [2.]

In any datasets, machine learning plays a vital role in analyzing and interpreting the patterns and structures to enable user learning, reasoning, and decision making without conducting any human interaction. Machine learning allows the user to feed algorithms on a large amount of data to start computer analysis and make decisions or recommendations based on input. The algorithm incorporates the information to improve its results in case of a correction is identified. [2.]

The thesis will showcase three simple methods on implementing when it comes to utilizing machine learning. This section describes the process and how more businesses besides technological companies could gain potential benefits, supporting the growth of a business, the concept of models, and importance.

2.1 Concepts and Models

Machine learning has a big impact on today’s world. As it is known that, machine learning is a field of artificial intelligence. It deals with large numbers of methods that discover relationships between data by identifying and classifying them into different sets of categories. For example, it is possible to find the suitable price of a certain house depending on the price of other houses with various values affecting features such as number of rooms, area of rooms, number of floors, and location by applying machine learning concepts. Machine learning concepts mainly focus on the prediction of certain possible events by using appropriate algorithms. Once data is fed on algorithms, then it is useful for prediction as a mathematical model. Certain parts of data known as training data are used to make mathematical models and the remaining parts of data known as testing data are used for the predictions. Meanwhile, the possible results of the specific problem can be found. [3.]

2.2 Business Understanding

Machine learning algorithms find the meaningful hidden insight or pattern from the given data to solve complex business problems. Machine learning algorithms take data as input and make it possible to find various kinds of patterns or ideas without using a programming language. Machine learning is evolving rapidly and being run by new computing tools and technologies.

Machine learning has given so many advantages to the business field. For example, machine learning helps to increase business scalability and business operation across the globe. Artificial intelligence and machine learning have huge popularity in the field of the business analytics community. Business sectors may have different factors affecting their business such as growing volumes of a certain product, easy accessibility of data, cheaper and faster computational processing, and affordable data storage. With the full understanding of the business use of machine learning, any organization can implement it in its process. [4.]

2.3 Importance

Machine learning has various functionalities where its models can be used to predict the future instance. Machine learning enables users to create algorithms able to do various tasks such as recognize numbers in images, recognize speech or sound, and detect diseases and frauds. The most interesting aspects of machine learning are that it makes easy things easy and hard things possible. Implementing machine learning algorithms has become easier and more convenient. There is no need to have a strong mathematical or programming background to build these programs. For instance, when a model is fed by data, it is executed, and adjustments are made until the desired output is gotten. Furthermore, large amounts of data can be used to make effective models and processes for higher computational decisions. In chapter 4, the thesis shows how simple the process of using machine learning with Python is.

3 Theory of Machine Learning

Machine Learning is the part of data science that acts as a primary tool in computational operations and algorithms, meeting statistical thinking in terms of a specific problem. The result is made from a collection of approaches to inference and data exploration.

The study of the machine learning concept arose from the researched context of applied data science methods. The methods are more helpful and easier when machine learning is thought of as a means of building models of data. Machine learning models are incredibly powerful, and they must be used under certain conditions. It is important to know which method is suitable for the specific problem. Machine learning also includes clear concepts of statistical terms such as bias, variance, overfitting, and underfitting.

Building different mathematical models are an important part of the machine learning process. They are used to analyze and understand data. A model is first set up with the necessary parameter imputation to adapt the observed data to start learning continuously. Once the model is set, it is useful to predict and understand aspects of newly observed data. [3.]

The following sub-sections to the current section have different parts such as machine learning history, modern machine learning, types of machine learning, and types of machine learning models. The sub-sections also include the five different algorithms belonging to supervised learning and some other basic terms of machine learning.

3.1 History of Machine Learning

The first Machine learning program to play checkers was written by Arthur Samuel in 1959 during his work period at IBM. The program was based on a minimax strategy of checkers or the program made moves to optimize the value of function assuming the opponent was doing the same things to optimize the same function. Arthur Samuel was well known as one of the top developers and researchers in the sector of computer gaming and AI in the United States. Samuel had impacted the early development and definition of machine learning during his career. [5.]

According to Arthur Samuel, machine learning is a field of study that gives computers the ability to learn without being explicitly programmed. On the checkers game, the program used to look at the status of boards, the position of the player, and the opponent while the game was played by multiple users. The game result was drawn at the board’s state after the game was played. Similarly, the checker's program checked the hypothetical position of the board pieces mapped from the current positions. At the end of the game, the furthest path from the hypothetical positions that led to the higher scores and it deserves a higher chance of winning. Finally, the program became a champion and showed the great promise in the field of Machine learning that can be achieved. [5.]

The development in the field of technology and huge practical research on the startup business machine learning has come to the modern state. According to Stanford University, machine learning was defined as “The science of getting computers to act without being explicitly programmed” [5.]. Nowadays, machine learning is also responsible for self-driving vehicles. The new set of concepts and technologies along with the different contents such as supervised, unsupervised learning, and various algorithms for robots, analytical tools are promoted due to machine learning. The major roles of machine learning in the current business of the world are listed below.

- Machine learning analyzes sales data.
- Machine learning helps in Real-Time mobile personalization to promote the experience.
- Natural language processing.
- Product recommendation for example displaying the same YouTube video next time on the home screen that was just played before.
- Dynamic or flexible pricing of product on demand of customer or need. That means customers see the product and price as they want or similar.

Machine learning tools enable organizations to identify profitable opportunities and potential risks more quickly by analyzing large and complex data faster and more accurately. Machine learning algorithms come with new computing technologies and business analytics and can resolve complex business problems. The field of information and technologies is rapidly developing so the application of machine learning has almost limitless possibilities. [5.]

3.2 Supervised and Unsupervised Learning

Supervised learning models show the relationship between features of data and some labels of data. When the model is ready, the model is useful for predictive analysis. Normally, to supervise means to observe and to direct the execution of tasks. In contrast, the final year project was carried out in the practice of supervised learning. The model was motivated by experience, so the model was trained or taught by a part of data chosen from the main data. In supervised learning, there is a further division between algorithms. Classification and regression algorithms will be discussed in subsequent sections. [6.]

In contrast to supervised machine learning, unsupervised learning models deal with unlabeled data to extract features and patterns on it. Supervised learning is one of the main categories of machine learning models. Unsupervised learning takes input probability densities, so the learning is also known as self-organization. Principal components and cluster analysis are the main methods used in supervised learning. Since the main topic of the thesis is based on supervised learning so it only focuses on related methods and terms to subjective learning. [6.]

3.3 Classification and Regression

Understanding the difference between regression and classification leads to the proper utilization of machine learning accurately. Some of the problems may need both classification and regression concepts but in-depth knowledge is crucial while selecting the appropriate model with a sufficient reason. The thesis has explained in detail the selection of the right models by solving real problems and evaluating the performance. [2.]

In classification methods, labels are discreetly categorized. Mainly, in the classification task, a set of labels data points are given and they need to be classified. The following Figure 1 shows a more descriptive feature of classification visually.

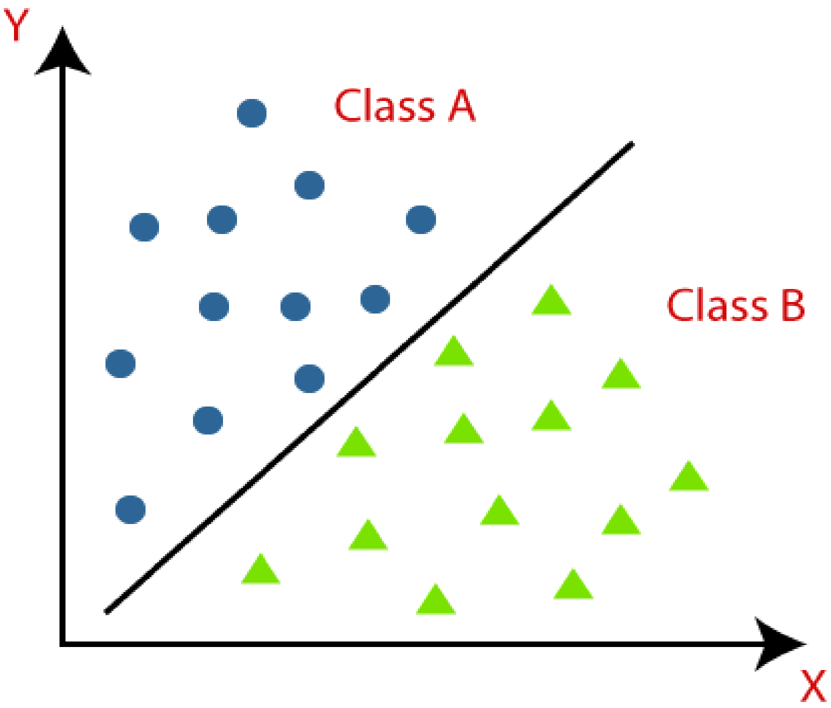


Figure 1. Diagram of classification. Screenshot [7.]

In figure 1, two types of input data, Class B as the first feature and Class A as the second feature, are seen. The line is drawn between ‘blue’ and ‘green’ classes so that the line divides both groups into their zones. The model is now ready after drawing the straight line also known as the trained model. Later, trained models, as shown in figure 1, can be generalized to a new or unlabeled dataset. In simple terms, the above model can label whether the unknown label is ‘blue’ or ‘green’.

Similarly, classification can also be used regardless of whether the given mail is ‘spam’ or ‘not spam’. In this case, a model should be trained by teaching the features of mail such as mail contents that may cover the length of the mail or number of words or phrases to categorize the mail as ‘spam’ or ‘not spam’ by inspection. The classification has various algorithms some of them are described in the later section of this chapter.

Regression methods are very useful to predict a continuous variable using some other variables. Two types of variables, such as dependent and independent, are presented in the regression problem. The dependent variable is always one but the independent variable can be one or more in a regression task. For example, a dataset has players' information about a certain club. Regression can be applied to predict the height of a player based on weight, gender, sport name, and diet. The height is a continuous quantity, and it has an infinite possibility for a person, so regression needs to be applied.

3.4 Types of Models

Performance in machine learning is evaluated based on accuracy and generalization. Generalization describes how properly a model works on seen and unseen data and accuracy checks how well the model predicts the right target value. When a machine learning model is formed by training data (seen data) and performance is calculated by testing data (unseen data). Based on the performance of the model, sometimes models may be overfitting or underfit or of the right fit. [8.]

When very simple algorithms are used to build models and are not able to learn complex patterns from the data to make the right predictions then underfitting occurs. The accuracy of underfit models is very lower either on testing or training data. Underfitting is also known as high bias. The underfit model is very simple because the model is made of the assumption that the data gives minimal focus to the training data or making the training data (taking fewer features) oversimple. For example, if a linear algorithm is applied to build a prediction model, the model would not be able to learn non-linear relationships between features and target values, and accuracy would decrease. High bias leads to a high error on training as well as on testing data sets [8.]. In the figure below the curve shows the underfitting character.

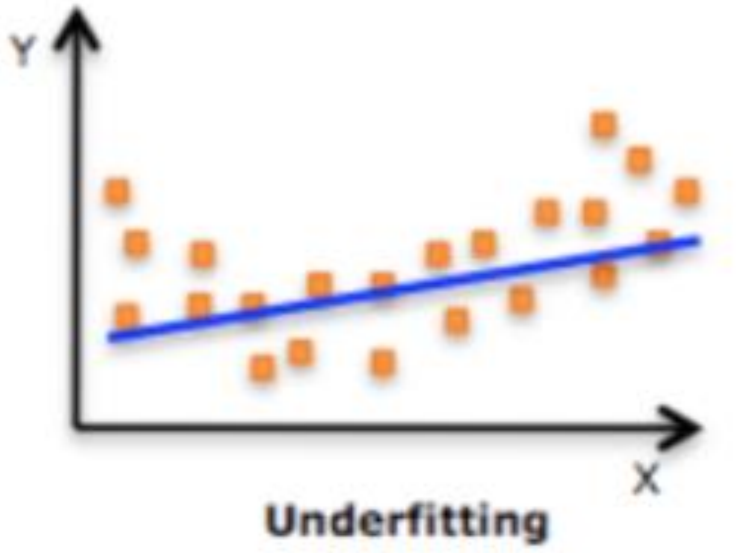


Figure 2. Diagram showing the underfitting model. Screenshot [7.]

When built models are very complex and based on training data, resulting in high accuracy of training data but the low accuracy of testing data, then overfitting occurs. In simple words, underfitting means a model has high accuracy on seen data and lower accuracy on unseen data so the case is also referred to as high variance. Overfitting happens in supervised learning when algorithms are strongly influenced by features of training data and cannot learn patterns properly which are noisy and are not generalized and fit only for training data sets. Figure 3 below clearly describes the overfitting character of the machine learning model.

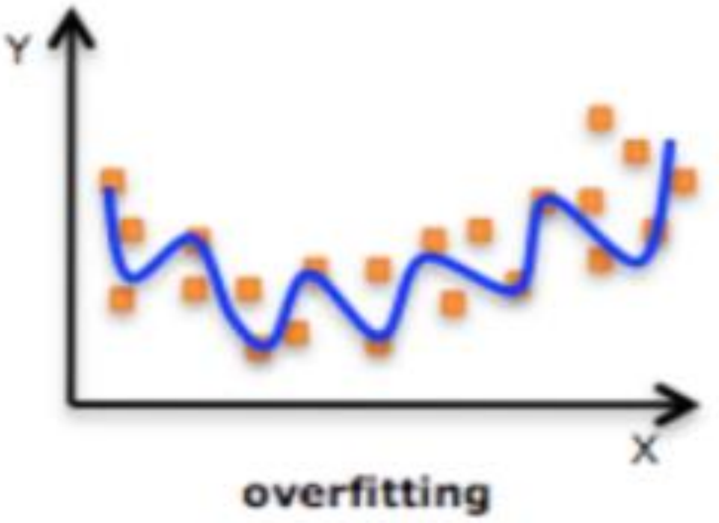


Figure 3. Diagram showing overfitting model. Screenshot [7.]

The model is said to be the right fitted model if the model is well generalized and behaves more or less accurately on both training and testing data. In other words, the model should have almost the same accuracy on training and testing data. The right model is seen when the model is neither underfitting and nor overfit. In practice, a trade-off between overfitting and underfit should be taken to get the right model [8.]. The figure below shows the right model.

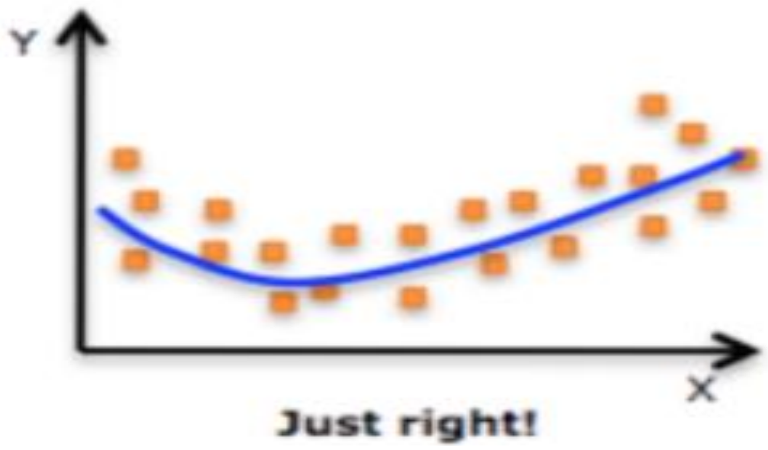


Figure 4. Diagram showing right fitted model. Screenshot [7.]

3.5 Machine Learning Algorithms

This section deals with the core and clear of five different machine learning algorithms used for the project. It also explains the reason behind them apply to a different situation.

3.5.1 Support Vector Machine

Support vector machine (SVM) is one of the powerful and flexible methods of supervised learning algorithms for both classification and regression. SVM is effectively useful in high dimensional space. SVM also uses a technique to transform the data. Then it finds the optimal boundary between the possible outputs where the number of dimensions is greater than the number of samples. Training points are used in SVM in a decision function to become memory efficient. SVM produces significant accuracy with less computational power so SVM is also a preferred algorithm used in the classification objectives. [2.]

3.5.2 K-nearest Neighbor

K-nearest neighbor (KNN) is a supervised machine learning algorithm known for instance-based learning algorithms. The k-nearest neighbor (KNN) algorithm aims to learn such functions like f(x)=y, where x is input and y is output. The main features of k-nearest neighbor (KNN) are that it is a lazy learning algorithm and a non-parametric method. Lazy learning is defined as an algorithm that takes zero time to learn data because k-nearest neighbor (KNN) stores only training data. Then k-nearest neighbor (KNN) will be used for the evaluation of a new query of data. K-nearest neighbor (KNN) does not need a hyperparameter as it does not assume any distribution so it is known as a non-parametric method. K-nearest neighbor (KNN) can produce a different approximation for each instance so its accuracy may be different from that of the target function. When a class attribute or a dependent variable can take a certain range of numbers, then k-nearest neighbor (KNN) is the best solution regardless of it [3.]. For example, whether a person has paid all loans displayed as ‘1’ or has not paid as ‘0’.

K-nearest neighbor (KNN) is very useful for recommendation systems mainly because it is an excellent baseline approach for the system. Similarly, k-nearest neighbor (KNN) is useful to search for semantically similar documents that contain identical topics if they are close to each other. Likewise, k-nearest neighbor (KNN) is useful to find credit card fraud detection. The use case of k-nearest neighbor (KNN)   will be discussed in chapter 4.

3.5.3 Random Forest Classifier

Random forest or ensemble learning is made for both classification and regression machine learning tasks. The random forest model builds many trees during the learning process and gathers all collected results. In terms of classification problems, the model is considered as the result. The random forest model has a significant advantage as compared to a single Decision Tree as a tree constructed independently based on random sample data. The random forest model has a faster learning rate and it avoids overfitting and underfitting and imputes missing values.

The random forest model also works on high dimensional data. The model can be applied to small as well as large portions of featured data. The random forest model can handle binary features, categorical features, and numerical features, so the Random forest is useful in both classification and regression problems. The data does not need to be rescaled or transformed as very little pre-processing is enough to procedure the process. [2.]

3.5.4 Decision Tree

The decision tree is a predicting modeling effort widely used in data science, machine learning, and statistics. Decision trees use predictive models to go from an observation about an item to a conclusion about the target values. The discrete set of values taken by target variables in decision tree models are called classification trees where leaves represent labels and branches represent labels. Decision trees have their intelligibility and simplicity, so decision trees are one of the most popular machine learning algorithms. Decision trees are used for both classification and regression problems. Decision trees break down the data set into smaller subsets with the development of decision trees. The resultant tree is with decision nodes and leaf nodes. Similarly, categorical and numerical data are also handled by decision trees. [9.]

3.5.5 Logistic Regression

Logistic regression is one of the most used supervised machine learning algorithms used for predicting categorical variables using the set of independent variables. The method can take multiple predictors that explain the response value and calculate the probability of categorical response either Yes or No, 0 or 1, true or false. Logistic regression is similar to linear regression, but it is also used for classification problems. Logistic regression can provide probabilities and classify new data using a continuous function and discrete datasets. The application of logistic regression is very high, as it can be used to detect whether the cells are cancerous or not or whether a cat is obsessed or not based on its weights. Similarly, logistic regression can be used to classify the observed data to find the effective features. [2.]

4 Methods and Materials

This is the most important section of the thesis. It describes the complete workflow of the project in detail and provides the necessary theoretical background.

4.1 Tools and Technologies

In this section, the tools and techniques applied in the project are described. The section mainly focuses on Python programming and the major libraries were used in the project.

4.1.1 Python

Python is an object-oriented high-level programming language created by Guido Van Rossum in 1991. The programming language was found to be used for general-purpose programming. It includes various interactive features that help the write clear and logical code for small as well as large projects. Because Python supports multiple programming paradigms and automatic memory management and dynamic type systems including object-oriented, imperative, functional programming, and procedure style, Python is one of the tops and most popular programming languages. Python 1.0 was released in January 1991, but Python code was already developed in February 1991 which had core functionality such as list, string, and some other data types. Besides, Python was known for object-oriented programming, and also the program came with a different module system. In 2000, Python 2.0 was released which supports Unicode and other notable values. The current version of Python is 3.8.0 and it was released in October 2019. [10.]

that the thesis demonstrates how simply the Python language can be used to create an effective and reliable machine learning program. Python's bunch of libraries and inbuilt modules are very useful to complete an array task along with machine learning, data visualization, and data analysis. In other words, Python provides large numbers of packages for making API calls and handling large datasets, so Python was chosen for the project to extract data from the chosen platforms. Some of the important Python libraries will be discussed in the next section.

4.1.2 Python Libraries

Python is a preferred language among data scientists because of its powerful functionality such as the Python standard library. The Python standard library is defined as the collection of scripted modules accessible to programs to simplify the program process which can be used by calling or importing them at the beginning of a script. In simple words, libraries are a collection of functions and methods that help to perform many tasks without writing code. Each of the Python libraries contains a huge number of modules that we can import into our daily programming task. Although there are so many libraries available in Python, only the main libraries used in the thesis project are discussed below.

Pandas

Pandas is an open-source library and a popular data science tool for data analysis and structure analysis. Machine learning supports the Pandas data structure, which is responsible for cleaning, transforming, manipulating, and analyzing data. Pandas is a fast, powerful, flexible, and open-source tool that allows importing data from various file formats such as CSV, JSON, SQL, and Microsoft Excel. Likewise, it operates different operations on the Pandas data frame such as merging, reshaping, selecting, data cleaning, and data wrangling. [11.]

NumPy

NumPy is a scientific computing Python package useful for linear algebra, creating arrays, and conducts with mathematical functionalities. NumPy supports large, multidimensional arrays and matrices along with high-level mathematical functions to operate so NumPy is very popular and useful in statistics, data analysis, and machine learning. NumPy is also an open-source library and NumPy can be used freely. Similarly, NumPy can be utilized to do various mathematical operations such as trigonometric, statistical, and algebraic routines. The main use of NumPy will be illustrated in a later session. [12.]

Matplotlib

Matplotlib is an open-source Python library for creating interactive and attractive visualizations. Matplotlib makes easy things easy and hard things possible to solve. Various collections of functions with different functionality are included in matplotlib to make works such as MATLAB. Matplotlib is also known as a plotting library able to produce graphs, histograms, lines, bar diagrams, and other various types of figures to get the visual insight of project data. Matplotlib is also supported to build static, animated, and interactive visualizations in the Python programming language. Matplotlib is an easy and convenient tool to visually analyze data points created by machine learning models. The use case of matplotlib is discussed in the method and material section. [2.]

Scikit-Learn

Like the discussed libraries above, Scikit-learn is also an open-source Python library. The main application behind Scikit-learn is to import different machine learning algorithm models such as logistic regression, KNN, decision tree, support vector machine, and random forest classifier. Scikit-learn is written in the Python language and it is built upon NumPy, SciPy, and matplotlib. Scikit-learn is a simple, usable, and effective tool mainly known for data mining and analysis. Later sections of the thesis will demonstrate how Scikit-learn was used to make predictive analytical models. [2.]

4.2 Data and Data Source

Data is a major part of a machine learning process aiming to solve analytical tasks. Suitable or right data is needed to solve major problems. Data can consist of values, text, numbers, or words. This means data can represent multiple things. This is why data need to be recorded properly and consistently to provide sufficient information to machine learning.

Data is generated every day in large amounts. The platform where the data is being stored and from where it can be obtained is called a data source. In short, the data source is the birthplace of data when information is first uploaded or published. Generally, machine data sources and file data sources are two types of data sources based on the kinds of information stored. Although plenty of information about data exists all around, in some cases the needed information is not available.

In the machine learning process, getting data is the first step, and there are many websites for finding data such as Kaggle, Reddit, Amazon web services, Google, and UCI. For the final project, Kaggle was chosen as a data source. Kaggle enables users to find datasets, upload datasets, interact, and work with co-workers to make models together. Similarly, there are also supportive questions, answers, and different challenges given to the user. LendingClub’s data from 2007-2010 was selected to demonstrate the predictive analytical work on the thesis. A CSV data file that has 14 columns was downloaded and 13 of the columns have their short name. Table 1 includes each column's short name.

Table 1. Description of columns names

Column name	Description
purpose	1: If a customer has similar credit criteria defined on LendingClub.com. 0: If a customer does not meet the criteria.
int.rate	The rate of interest charged on borrowers (a rate of 10% is stored as 0.1). The risky borrower confirmed by the club is assigned higher interest rates.
installment	The monthly repayment is owed by the borrower after the loan is granted.
log.annual.inc	The automatically reported log of the annual income of the borrower.
Dti	The ratio between debt and annual income of the borrowers.
Fico	Credit score refers to the borrowers.
days.with.cr. line	Borrowers having some number of credit lines in days.
revol. bal	The remaining or unpaid amount revolved by borrowers at the end of the credit card billing period.
revol. util	The utilization rate or amount of credit line on available credit revolved by borrowers.
inq.last.6mths	The number of inquiries by investors in the last six months on borrowers about repayment.
delinq.2yrs	The numbers of times delayed on payment made by borrowers in last two years.
pub.rec	The borrower's public records such as bankruptcy filings, tax liens, or judgments may affect the loan approval.

4.3 Data Implementation

The current section and its subsection briefly discuss the complete machine learning workflow and steps that need to be followed to build a predictive thesis problem project. The section clarifies the data performance and stepwise machine learning process done in the case project. Similarly, evaluation of the different models is calculated in the subsequent sections.

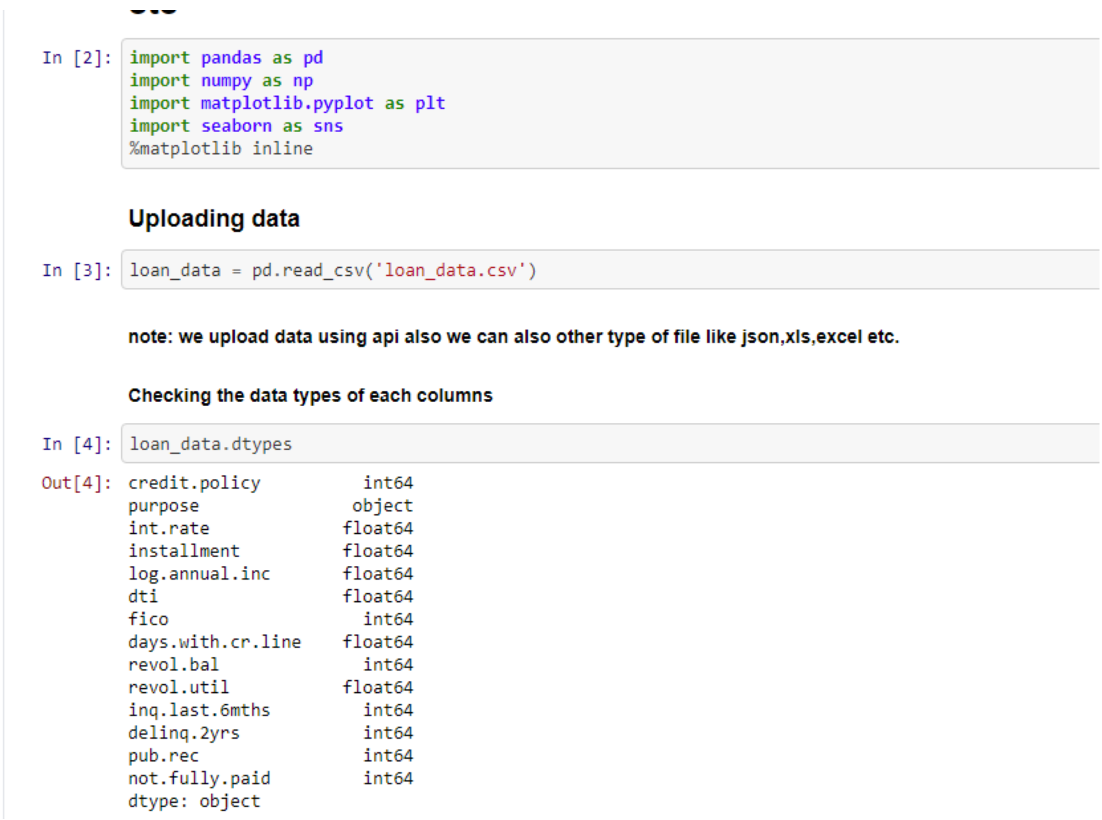
4.3.1 Data Preparation

Data preparation is the first step of machine learning workflow after the right project data has been selected. In simple words, data processing or preparation is similar to washing fresh vegetables to remove unwanted elements to cook clean curry. Similarly, data can be processed to get it into a state where working with it is possible. To work with data, it must be prepared in such a way that addresses missing values and invalid values. Also, duplicates need to be removed to confirm everything is properly formatted. [13.]

Generally, machine learning algorithms need all data to be numbers before modeling the predictive model. Similarly, some data may have statistical noise that should be corrected and complex non-linear relationships may be out of data. Data preparation is required to fit and evaluate machine learning models. Data preparation can also be defined as the transformation of raw data into a suitable form for modeling [13.]. In the upcoming sections, the process is gone through as has been discussed.

4.3.2 Loading the data

To load the sample data, Pandas was imported on Jupyter notebook (text editor). Also, other required libraries for the processing such as Numpy, Matplotli, and Seaborn were imported. Figure 5 shows loading project data into the work state, also called a text editor. A CSV file was downloaded from Kaggle for the project and it was uploaded to the text editor (Jupyter Notebook). Pandas allow uploading files through an API. It also supports other file formats such as JSON, XLS, and XML.



```
In [2]: import pandas as pd
        import numpy as np
        import matplotlib.pyplot as plt
        import seaborn as sns
        %matplotlib inline
```

Uploading data

```
In [3]: loan_data = pd.read_csv('loan_data.csv')
```

note: we upload data using api also we can also other type of file like json,xls,excel etc.

Checking the data types of each columns

```
In [4]: loan_data.dtypes

Out[4]: credit.policy          int64
        purpose               object
        int.rate             float64
        installment          float64
        log.annual.inc       float64
        dti                  float64
        fico                   int64
        days.with.cr.line    float64
        revol.bal              int64
        revol.util           float64
        inq.last.6mths         int64
        delinq.2yrs            int64
        pub.rec                int64
        not.fully.paid         int64
        dtype: object
```

Figure 5. Uploading and checking data types of all columns. Screenshot [14.]

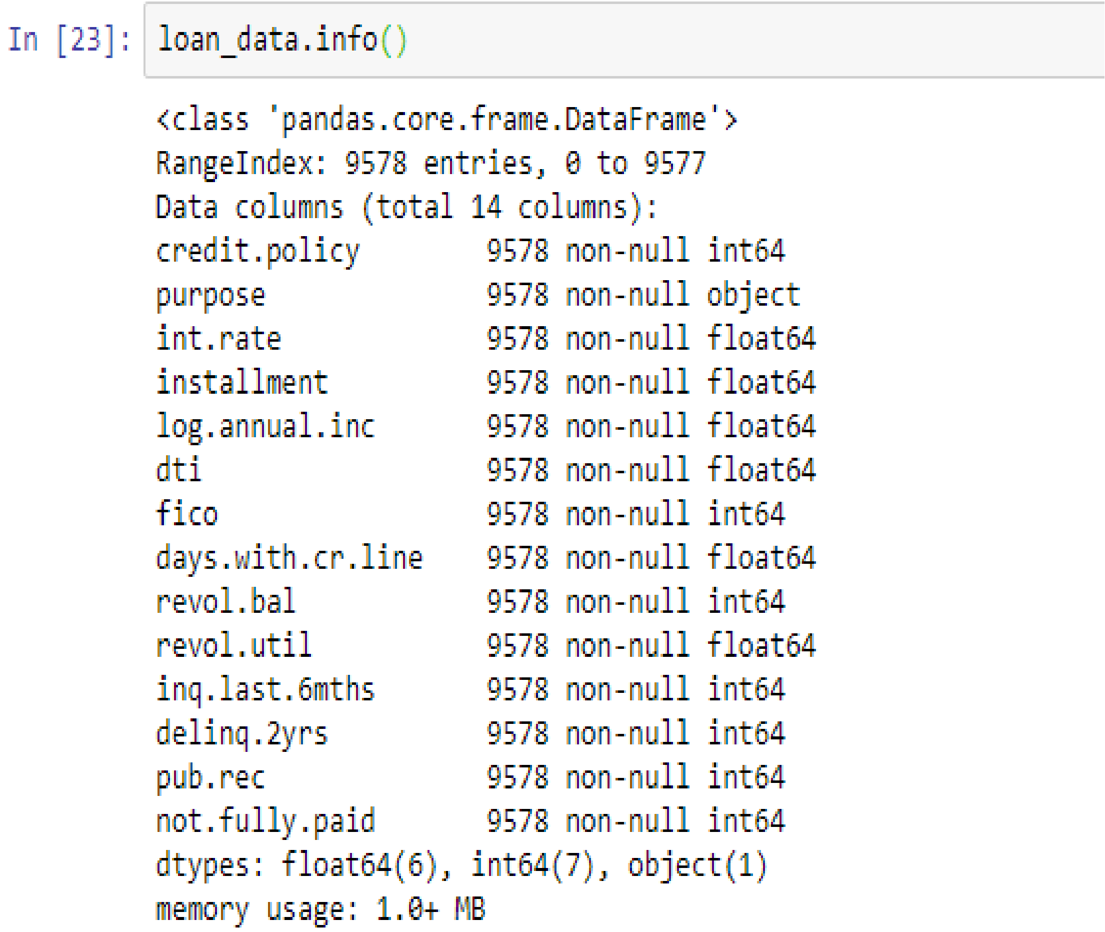
The above figure shows the setup of all libraries for data handling and explanatory analysis. Seaborn and Matplotlib were used for the visualization part and NumPy was used for other mathematical operations. After loading the data into the user system, the data type had to be checked clearly to determine the effective features that influenced the target array. Likewise, the data format could also be changed into the right format. The data ‘purpose’ column was an object, categorical feature, and a part of loan payment potential. However, machine learning only supports numerical values, and all values needed to be changed into numerical.

4.3.3 Descriptive Analysis

The descriptive analysis gives great insight into the shape of the data. In descriptive analysis distribution and quantitative insight of each attribute are summarized with different statistical measurements such as mean, median, mode, min-max value, standard deviation, percentile, and variance. While statistically describing the data, it can be determined whether the data has missing values or not. Finding surprising distribution for attributes is also possible. Similarly, in the case of categorical features or attributes, the distribution class is observed and feature data can be inspected usually using a bar chart, a histogram, a pie chart, and other figures.

4.3.4 Categorical Features

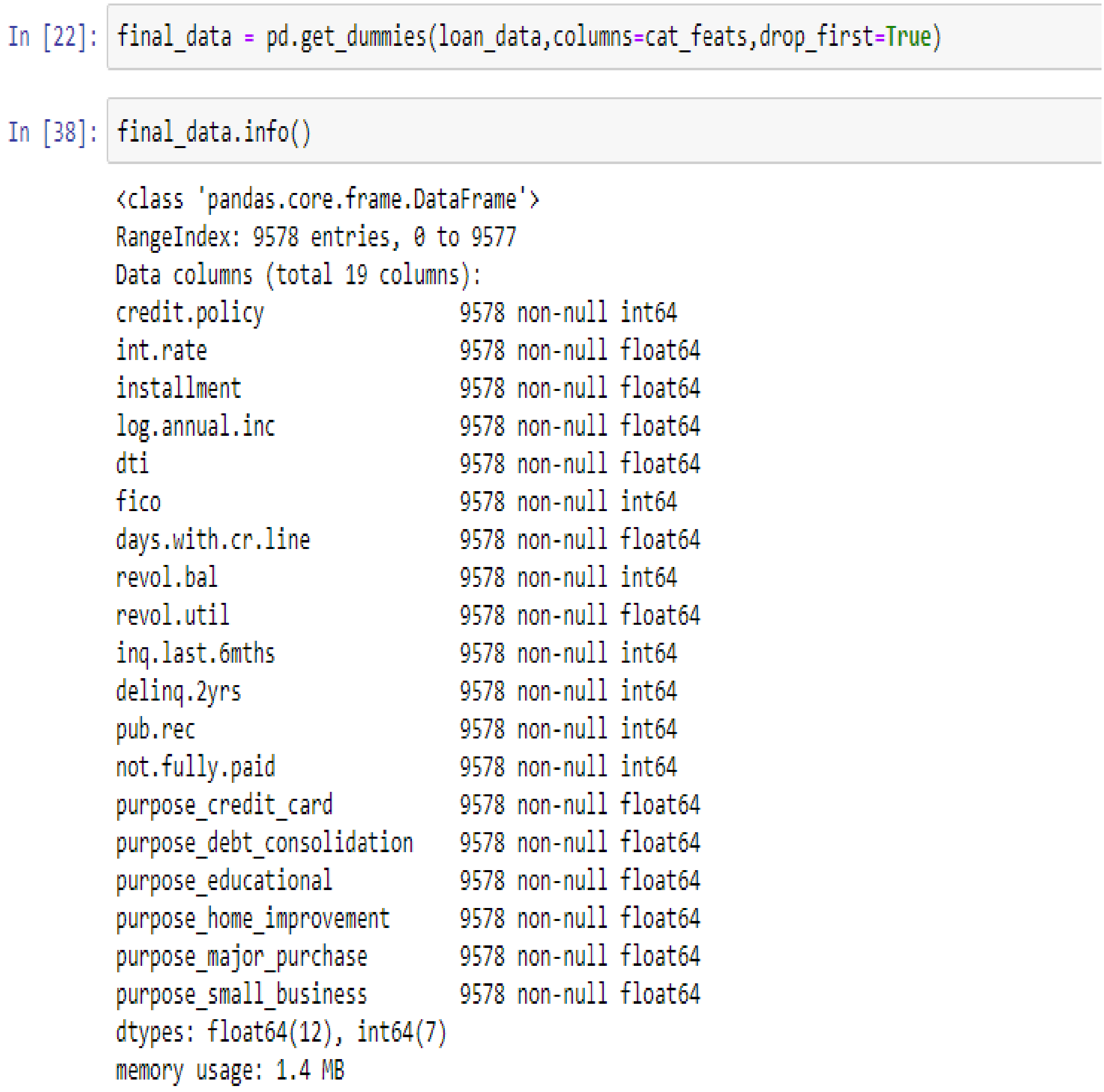
Categorical features or data are defined as the categorical data, representing discrete values belonging to a specific finite set of categories or classes. Discrete value can also be text or numeric and include unstructured data such as images. The concept of machine learning algorithms only computes with numerical data. In simple terms, machine learning algorithms cannot directly work with categorical data so feature engineering and transformations need to be applied before starting to build models. Figure 6 illustrates the process of feature handling of data and how formats required by machine learning are used.



```
In [23]: loan_data.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9578 entries, 0 to 9577
Data columns (total 14 columns):
credit.policy        9578 non-null int64
purpose              9578 non-null object
int.rate             9578 non-null float64
installment          9578 non-null float64
log.annual.inc       9578 non-null float64
dti                  9578 non-null float64
fico                 9578 non-null int64
days.with.cr.line    9578 non-null float64
revol.bal            9578 non-null int64
revol.util           9578 non-null float64
inq.last.6mths       9578 non-null int64
delinq.2yrs          9578 non-null int64
pub.rec              9578 non-null int64
not.fully.paid       9578 non-null int64
dtypes: float64(6), int64(7), object(1)
memory usage: 1.0+ MB
```

Figure 6. Data set before handing categorical values. Screenshot [14.]

The figure shows 14 columns of the project data with the columns' names and their types. As discussed earlier, machine learning only deals with numerical values such as int and float shown in the above figure but not with objects. The column name 'purpose' is an object type and a categorical feature. The 'purpose' column is changed into a number by getting the numerical dummy values shown in the figure below.



```
In [22]: final_data = pd.get_dummies(loan_data,columns=cat_feats,drop_first=True)

In [38]: final_data.info()

         <class 'pandas.core.frame.DataFrame'>
         RangeIndex: 9578 entries, 0 to 9577
         Data columns (total 19 columns):
         credit.policy                 9578 non-null int64
         int.rate                      9578 non-null float64
         installment                   9578 non-null float64
         log.annual.inc                9578 non-null float64
         dti                           9578 non-null float64
         fico                          9578 non-null int64
         days.with.cr.line             9578 non-null float64
         revol.bal                     9578 non-null int64
         revol.util                    9578 non-null float64
         inq.last.6mths                9578 non-null int64
         delinq.2yrs                   9578 non-null int64
         pub.rec                       9578 non-null int64
         not.fully.paid                9578 non-null int64
         purpose_credit_card           9578 non-null float64
         purpose_debt_consolidation    9578 non-null float64
         purpose_educational           9578 non-null float64
         purpose_home_improvement      9578 non-null float64
         purpose_major_purchase        9578 non-null float64
         purpose_small_business        9578 non-null float64
         dtypes: float64(12), int64(7)
         memory usage: 1.4 MB
```

Figure 7. Table after changing categorical values into numbers. Screenshot [14.]

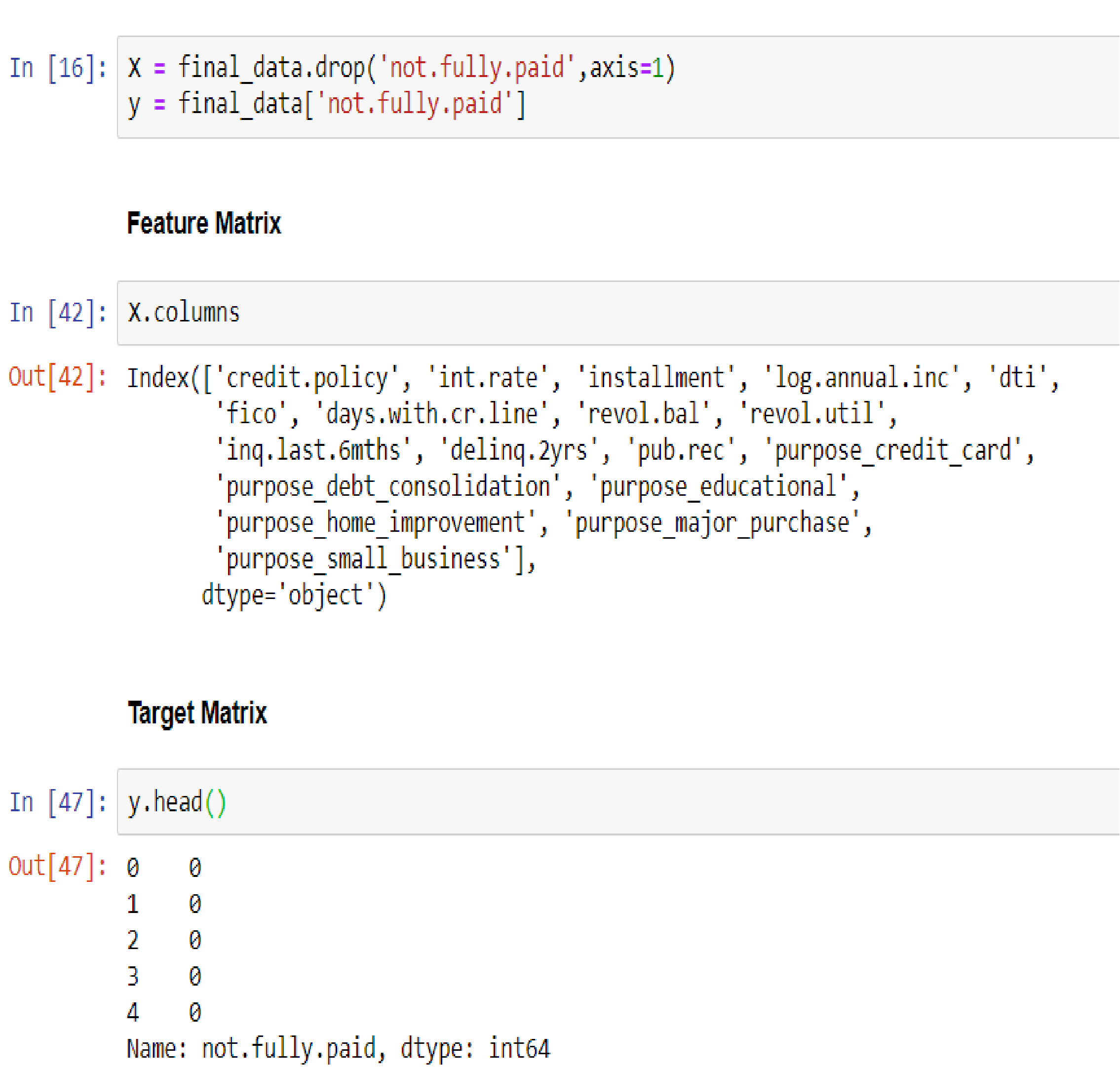
In figure 7, at step 22, the Pandas method named 'get_dummies' was used to change the 'purpose' column that was previously set as 'cat_feats' into its feature's columns. Finally, at the same step 'cat_feats' was deleted by setting parameter 'drop_first' as 'True' and the new file named as 'final_data'. At step 38, the information of final data was checked by applying the 'info()' method. The 'cats_feats' columns were removed and as a result, six different categorical columns were added. Subsequently, all the categorical columns were of ‘float’ data types as they needed to have numerical values for the upcoming procedures.

4.3.5 Feature Matrix and Target Array

Generally, rows of matrices are referred to as samples and the number of rows as n_samples. Similarly, each column of the data belongs to specific information described by each row. Normally, columns of the matrices are referred to as features and the number of columns as n_features.

Feature Matrix

The table layout has a two-dimensional numerical array or matrix called a feature matrix. The feature matrix has shape [n_samples, n_features] most often contained in a NumPy array or a Pandas data frame, and the matrix is stored as a variable name [2.]. X is a feature matrix as shown in figure 8. The sample always describes the individual objects described in the data set and the sample might be a person, an image, a symbol number, or anything else that is described with quantitative measurements. Features or columns are generally real-valued but in some cases, they may be boolean, discrete, or categorical. In figure 8, at step 42, the feature matrix list was made with the name of the column.



```
In [16]: X = final_data.drop('not.fully.paid',axis=1)
         y = final_data['not.fully.paid']
```

Feature Matrix

```
In [42]: X.columns

Out[42]: Index(['credit.policy', 'int.rate', 'installment', 'log.annual.inc', 'dti',
                'fico', 'days.with.cr.line', 'revol.bal', 'revol.util',
                'inq.last.6mths', 'delinq.2yrs', 'pub.rec', 'purpose_credit_card',
                'purpose_debt_consolidation', 'purpose_educational',
                'purpose_home_improvement', 'purpose_major_purchase',
                'purpose_small_business'],
               dtype='object')
```

Target Matrix

```
In [47]: y.head()

Out[47]: 0    0
         1    0
         2    0
         3    0
         4    0
         Name: not.fully.paid, dtype: int64
```

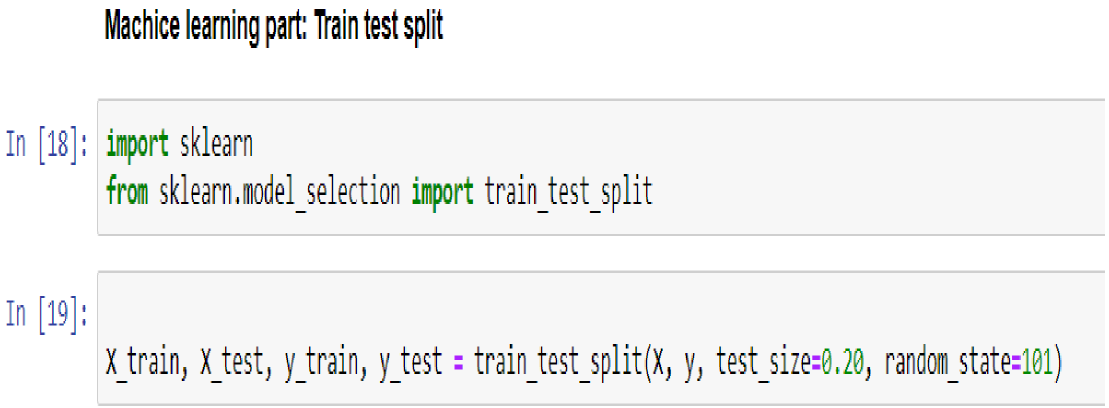
Fig 8. Selecting feature matrix and target array. Screenshot [14.]

Target Array

The target array of the data set is one of the features of the dataset that need to be understood. Moreover, relevant insights need to be made. Target arrays vary depending on the business goal and available data. However, the target array has one dimension with n_samples and an array contained in a NumPy and Pandas Series [2.] as shown in figure 8 at step 47. The values of the target array may be continuous numerical values or discrete classes or labels. The target array represents the quantity needed to made predictions based on the data, depending on the remaining features of the data as shown in step 42 in figure 8. In the case project, 'not.fully.paid' was a target array and was calculated as either 0 or 1 as shown in the same figure.

4.3.6 Data Splitting

In the machine learning process, a training data set is used to build predictive models. Then testing data sets are used to check the model validation. Training and testing data points are separated from each other, and the process is called splitting of data. In machine learning, it is necessary to have a validated model. To start the process, data sets are divided into two parts as training and testing to check accuracies and precisions on the case data. The figure below shows the process of splitting.



Machice learning part: Train test split

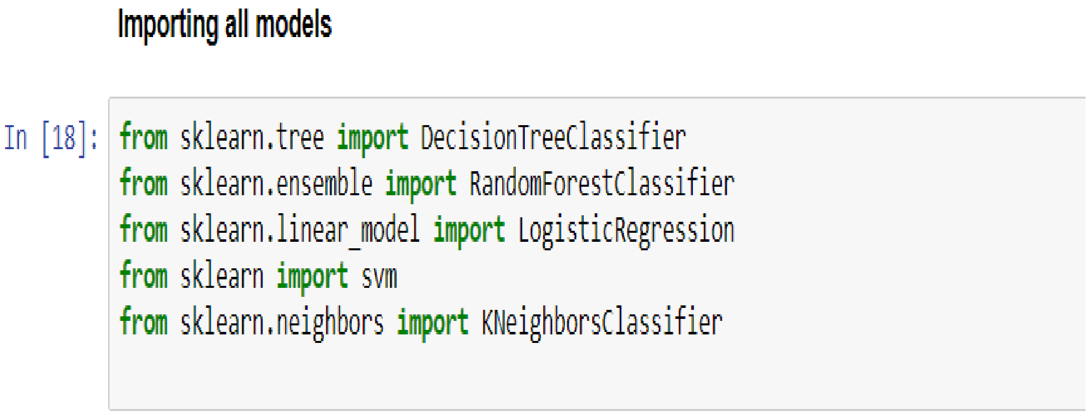
```
In [18]: import sklearn
         from sklearn.model_selection import train_test_split

In [19]:
         X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.20, random_state=101)
```

Fig 9. Diagram of train test split. Screenshot [14.]

In the above figure, at the first step, the ‘train_test_split’ method was imported from the Scikit-learn library. The main idea about selecting training and testing data sets was that the model trained by larger datasets gives better accuracy. Consequently, the above figures show that at the second step data was divided into two sets: 80% for training the model and 20% for testing the model.

4.4 Model Development

To start the model development process, the format of the data set had to be checked and corrected into the right format before training the model. There are many algorithms in the field of machine learning but the goal of the thesis is to deal with classification algorithms of supervised learning only. As a result, five different models as described in section 3.4 were chosen for the project. All the five algorithms are found in the Scikit-learn library and it was imported as shown in the figure below.



Importing all models

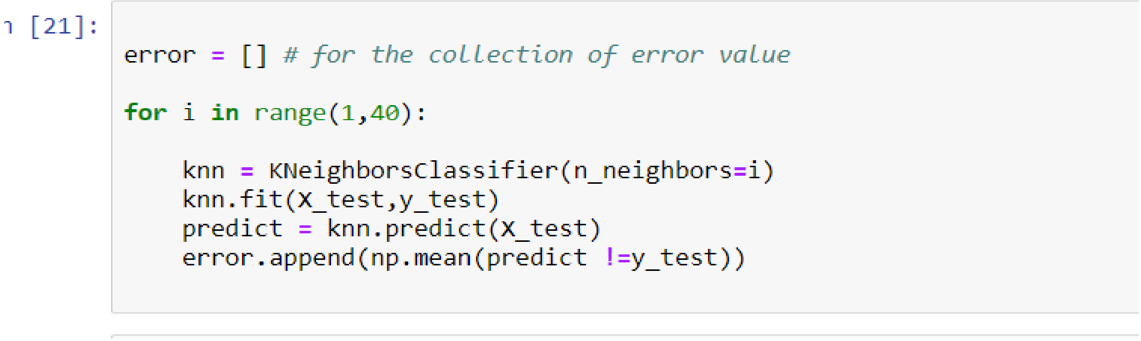
In [18]:

```python
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.linear_model import LogisticRegression
from sklearn import svm
from sklearn.neighbors import KNeighborsClassifier
```

Fig 10. Importation of all models. Screenshot [14.]

4.4.1 Hyperparameter Selection

Before building the Machine learning model value, hyperparameter selection needed to be done. It was an important part of the project, because the value of the parameter name k of KNN algorithms should be the best possible fit for data. With the small value of k, the model produces the most flexible fit with low bias and high variance and vice versa. The figure below clearly shows the selection process of value k.



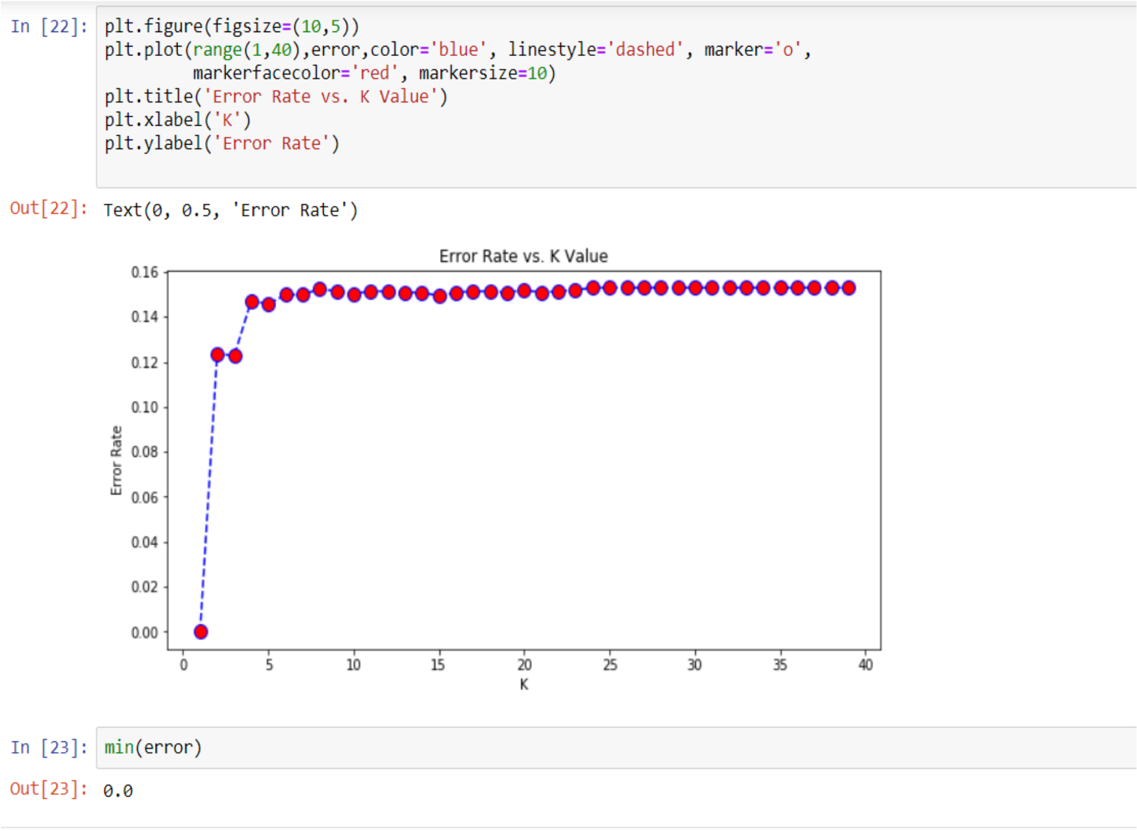
n [21]:

```python
error = [] # for the collection of error value

for i in range(1,40):

    knn = KNeighborsClassifier(n_neighbors=i)
    knn.fit(X_test,y_test)
    predict = knn.predict(X_test)
    error.append(np.mean(predict !=y_test))
```

Fig 11. Creating the errors list. Screenshot [14.]

Figure 11 shows the use case of a Python loop to find suitable values of hyperparameter value k. Firstly, an empty list named ‘error’ was created as a list of error values given by an algorithm. The Python loop had been applied to find the best fit value of k in the range of 1 to 40. After the completion of the Python loop, a list of error values was generated based on the values of k. It can be seen in the following figure 12.

.



```
In [22]: plt.figure(figsize=(10,5))
         plt.plot(range(1,40),error,color='blue', linestyle='dashed', marker='o',
                  markerfacecolor='red', markersize=10)
         plt.title('Error Rate vs. K Value')
         plt.xlabel('K')
         plt.ylabel('Error Rate')

Out[22]: Text(0, 0.5, 'Error Rate')
```

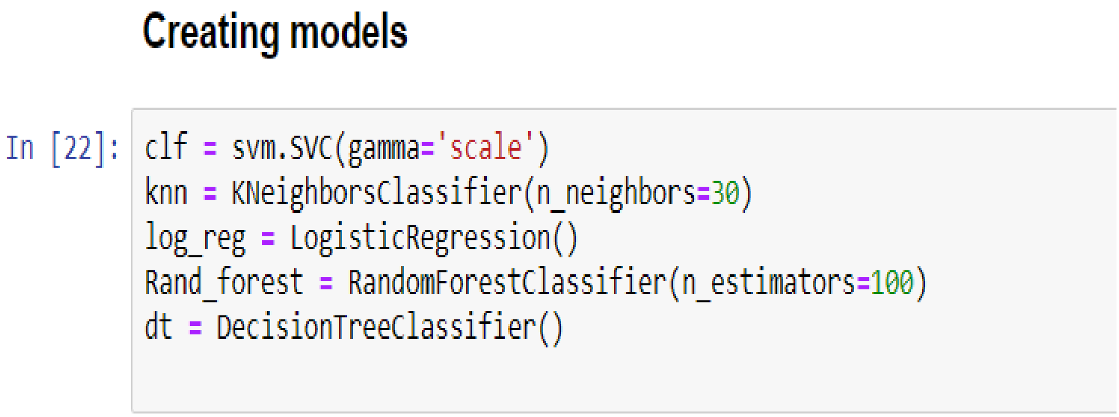
```
In [23]: min(error)

Out[23]: 0.0
```

Fig 12. Plotting the errors against different values of k. Screenshot [14.]

Matplotlib was used in the above figure to show the error rate against the value of hyperparameter k. The value of k having either very small or very large values cannot give good performance results. In figure 12 some values were found without any errors because of the highly accurate value of k as shown in step 23. For the case project, 30 was a suitable value of hyperparameter k. In the next section, the process is described further.

4.4.2 Creating the Models

In the current section, five different predictive models were created. Similarly, the value of hyperparameter for KNN and Random Forest Classifier was chosen as 30 and respectively to make the model more accurate and reliable. “gamma=’scale” in the SVM was set to fix the warning shown in the case of used versions of Python libraries as shown in the figure below. However, the visibility of warning messages had not affected the result of our problem after the model was tested many times. The following figure clearly shows the process of making the model.



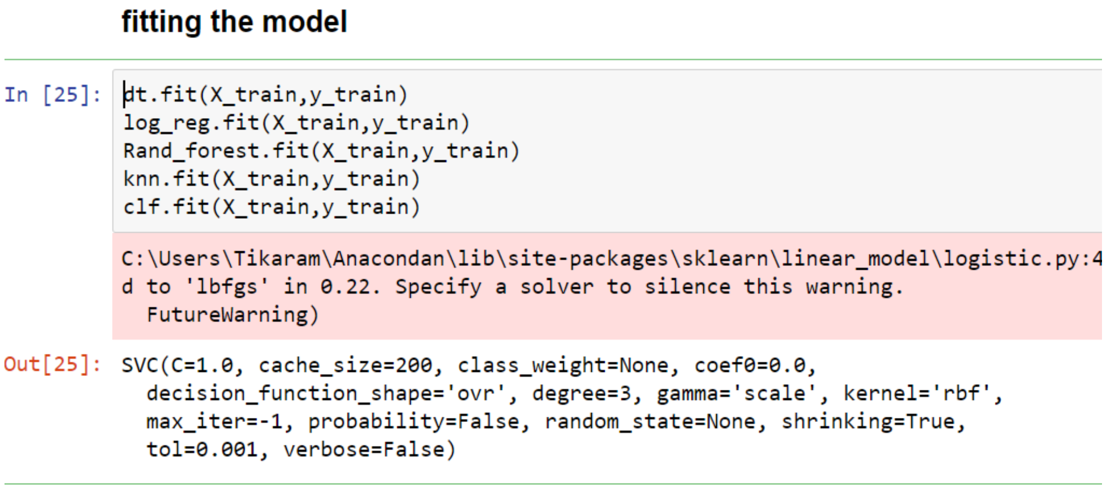
Creating models

```
In [22]: clf = svm.SVC(gamma='scale')
         knn = KNeighborsClassifier(n_neighbors=30)
         log_reg = LogisticRegression()
         Rand_forest = RandomForestClassifier(n_estimators=100)
         dt = DecisionTreeClassifier()
```

Fig 13. Diagram of creating models. Screenshot [14.]

4.4.3 Fitting the Models

Fitting the model is defined as the procedure where our algorithm runs on our training data set to build the predictive model. A well-fitted model with sufficient hyperparameter adjustment produces more accurate outcomes in the case the project data was divided into a training set and testing set. The training data set was used in the form of 'X_train' as the x coordinate and 'y_train' as they coordinate to build the well-fitted model shown in the following figures.



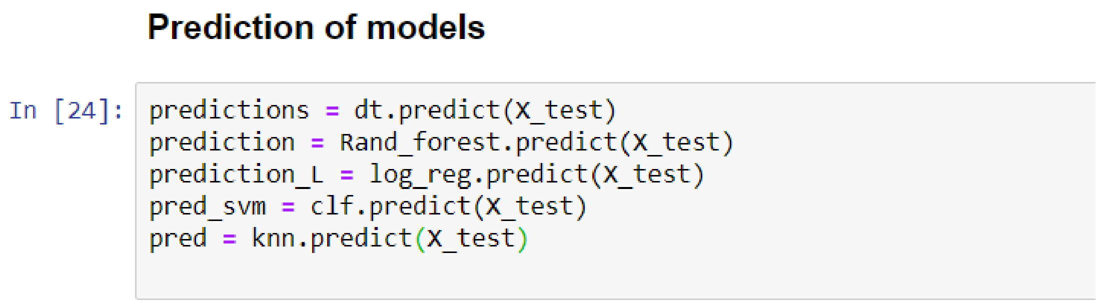
fitting the model

```
In [25]: dt.fit(X_train,y_train)
         log_reg.fit(X_train,y_train)
         Rand_forest.fit(X_train,y_train)
         knn.fit(X_train,y_train)
         clf.fit(X_train,y_train)

         C:\Users\Tikaram\Anacondan\lib\site-packages\sklearn\linear_model\logistic.py:4
         d to 'lbfgs' in 0.22. Specify a solver to silence this warning.
           FutureWarning)

Out[25]: SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
           decision_function_shape='ovr', degree=3, gamma='scale', kernel='rbf',
           max_iter=-1, probability=False, random_state=None, shrinking=True,
           tol=0.001, verbose=False)
```

Figure 14. Fitting of all models. Screenshot [14.]

4.4.4 Model Prediction

After the model had been trained, the term ‘prediction’ refers to the output of the built machine learning algorithm models on training as well as testing data. The fitted model generates all the probable values for an unknown variable record in the test data. For instance, in the project data ‘X_test’ was used to predict the most likely value by the built model. Prediction is an essential step to check the performance of a model and to find the required solution. Figure 15 clearly shows the use case of model prediction that happened in the project.



Prediction of models

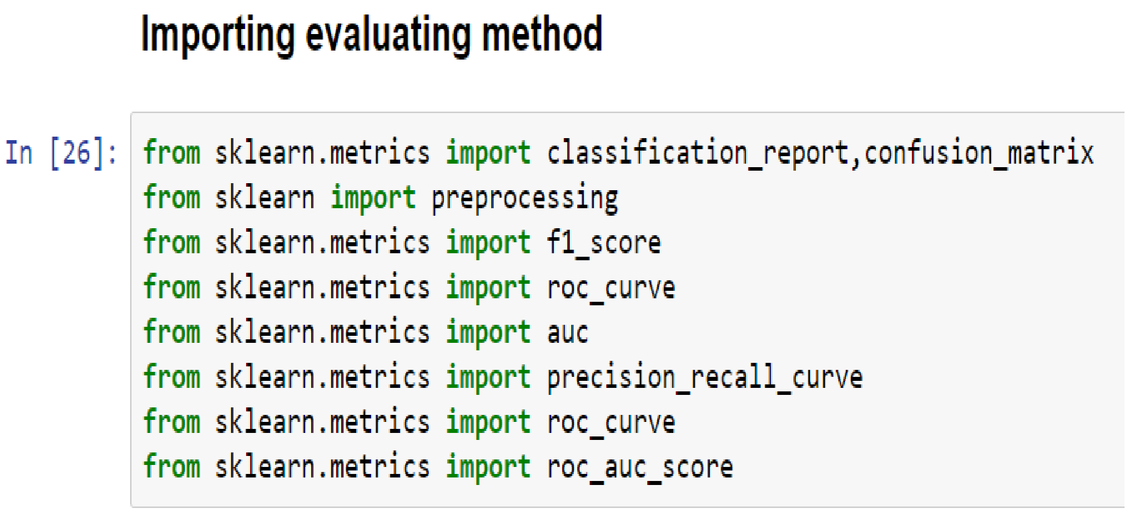
```
In [24]: predictions = dt.predict(X_test)
         prediction = Rand_forest.predict(X_test)
         prediction_L = log_reg.predict(X_test)
         pred_svm = clf.predict(X_test)
         pred = knn.predict(X_test)
```

Fig 15. Diagram showing the predictive values of all models. Screenshot [14.]

4.5 Model Evaluation

Machine learning models are responsible for giving high accuracy to produce real or more likely value to the given problem. After training the model it is important to know whether the built model works or not and whether the model provides trusted predictions. Similarly, the model can only memorize the data, unable to make good predictions or performance. Thus, the evaluation is essential in machine learning to estimate the generalized accuracy of the model on future data. [15.]

For the case project, mainly f1-score and Jaccard similarity score were calculated to compare the results of all the built models. Besides that, the ROC curve and the AUC curve were also imported as evaluation matrices. The confusion matrix was made as a base chart board to get an idea about applying other evaluation matrices. The major evaluation methods imported for the project are shown in detail in figure 16.



Importing evaluating method

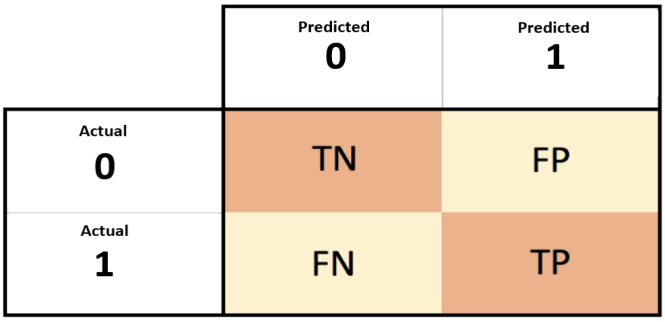
```
In [26]: from sklearn.metrics import classification_report,confusion_matrix
         from sklearn import preprocessing
         from sklearn.metrics import f1_score
         from sklearn.metrics import roc_curve
         from sklearn.metrics import auc
         from sklearn.metrics import precision_recall_curve
         from sklearn.metrics import roc_curve
         from sklearn.metrics import roc_auc_score
```

Fig 16. Importing all evaluation methods. Screenshot [14.]

In this section, the predictions made by the built model to new and previously unseen data or tested data is evaluated by different evaluation methods. The important methods and used methods in the project are described below.

Confusion Matrix

The confusion matrix is the most powerful analytical tool in machine learning and data science and is capable of summarizing the classification algorithms. The confusion matrix helps to understand how the classification model is working, whether giving right or wrong predictions. The key point towards the confusion matrix is to check the summarized correct and incorrect predictions with count values belonging to their respective own classes as shown in the figure below.



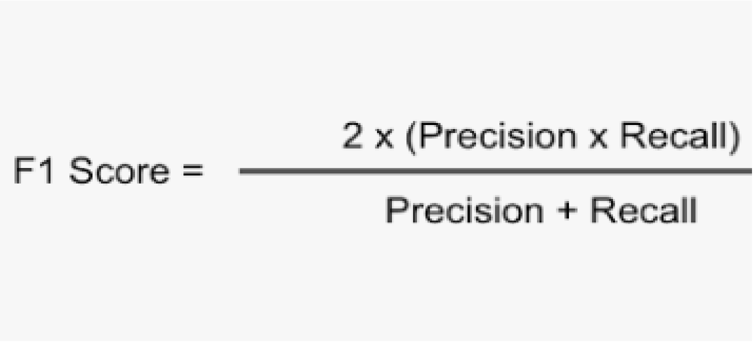
	Predicted 0	Predicted 1
Actual 0	TN	FP
Actual 1	FN	TP

Fig 17. Confusion matrix. Screenshot [16.]

Figure 17 clearly shows the confusion matrix for classification problems having four boxes. The picture illustrates the prediction in comparison with actual outcomes. Similarly, the first row gives the outcome as falsely and correctly predictive negative values whereas the second row show the outcome as falsely and correctly predicted positive values. [15.]

F1-score

F1-score is also known as F-measure. The values of precision and recall are required to find the value of the F1-score, and the harmonic mean of precision and recall is an F1-score. The values of the F1-score lie between 0 to 1 and the F1-score ignores the true negative values. However, the F1-score is a very useful evaluation tool, and the score can be calculated using the following formula shown in figure 18.



$$\text{F1 Score} = \frac{2 \text{ x } (\text{Precision x Recall})}{\text{Precision} + \text{Recall}}$$

Fig 18. The formula of f1-score. Screenshot [15.]

Roc Curve

Roc stands for receiver operating characteristic curve to illustrate the diagnostic ability of a binary classifier. In machine learning, the Roc curve is created by plotting the true positive rate against the false-positive rate at different threshold settings. In simple words, Roc shows the performance of the classification model at all classification thresholds. [15.]

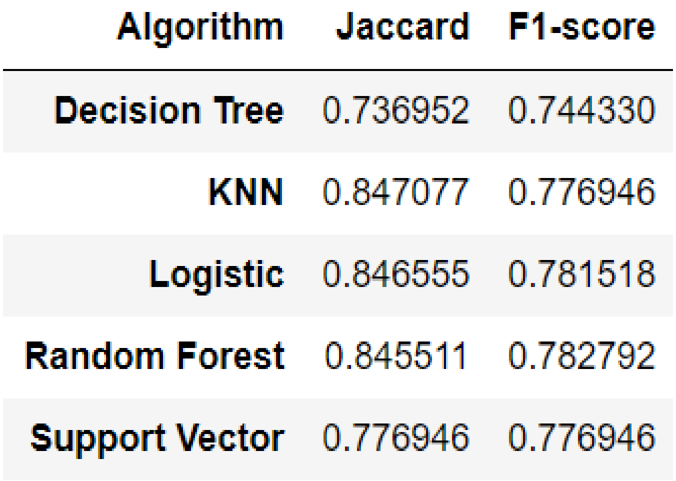
AUC Curve

AUC is known as the area under the Roc curve that measures the aggregate performance across all possible classification thresholds. AUC measures how well predictions are ranked instead of their actual value. Similarly, AUC measures the quality of the model predictions irrespective of the chosen threshold. The values of AUC ranges from 0 to 1 [15.]. For instance, the value of AUC is 0 for a prediction of the model, which is completely wrong. The value of AUC is 1 a correct prediction.

5 Results and Discussion

As stated in the previous section, predictive analytical features were integrated into the used text editor, Jupyter Notebook, including machine learning tools. Similarly, a complete workflow of machine learning was explained in detail, predictive models were constructed, and different evaluation methods were imported and their theoretical application was discussed in section 4. The results of the models found after the use of different machine learning algorithms are summarized and described in section 3.4, and their performance on test data using the Jaccard similarity score and f1-score was evaluated during the project.

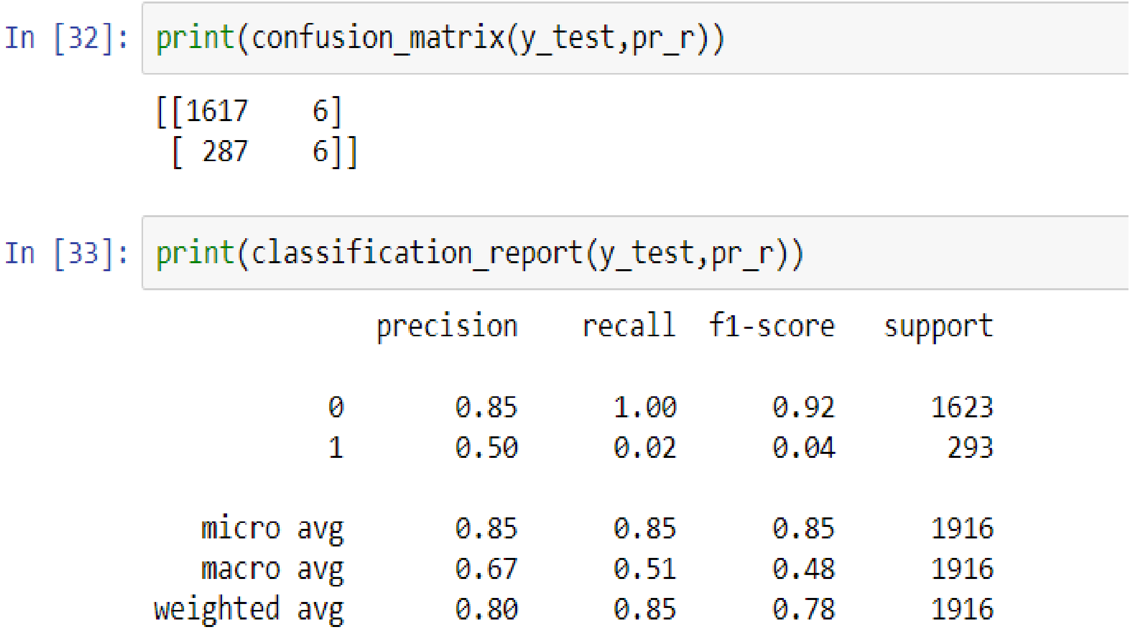
In the project, different machine learning algorithms were used, and two evaluation methods were chosen for the model performance evaluation. The values of the Jaccard similarity score and the f1-score of five different algorithms were calculated as shown in figure 19.



Algorithm	Jaccard	F1-score
Decision Tree	0.736952	0.744330
KNN	0.847077	0.776946
Logistic	0.846555	0.781518
Random Forest	0.845511	0.782792
Support Vector	0.776946	0.776946

Fig 19. Performance score table. Screenshot [14.]

As shown in the figure above, all the models except Decision Tree could be concluded to be highly accurate models. As the figure shows, four out of five models had 78% as an average f1-score, and three models named KNN, logistic and random scored almost 84% as Jaccard similarity. To achieve the result data was first divided into training as testing dataset 80% and 20% respectively. Similarly, the value of the parameter of KNN was 30, and the numbers of estimators in the random forest model n for 300 were selected. In comparison, the performance KNN, random forest, and logistic regression models had almost the same high accuracy as the other two algorithms in the case project. Support vector had a good f1-score as three highly accurate models have but the Jaccard similarity score is almost 7% lower. Regarding the decision tree models, the model only had a Jaccard similarity score of 73% and an f1-score of 74%. Therefore, one of the models among KNN, Logistic regression, and Random Forest classifier was concluded to be a better option for the case project. The classification report and confusion matrix of the Random forest models are shown in the figure below.



```
In [32]: print(confusion_matrix(y_test,pr_r))

         [[1617    6]
          [ 287    6]]

In [33]: print(classification_report(y_test,pr_r))

                       precision    recall  f1-score   support

                    0       0.85      1.00      0.92      1623
                    1       0.50      0.02      0.04       293

            micro avg       0.85      0.85      0.85      1916
            macro avg       0.67      0.51      0.48      1916
         weighted avg       0.80      0.85      0.78      1916
```

Fig 20. Confusion matrix and classification reports. Screenshot [14.]

As shown in the confusion matrix, the model predicted that 1616 out of 1623 are 'not.full.paid' as false with a precision value of 0.77 and a recall value of 0.85. The table clearly shows that the model is highly accurate and useful to make decisions. Also, the use of at least five different algorithms is a very good option to make comparable analytical tasks to make good decisions. Similarly, the process can be run through more categorical analysis to improve and secure decisions even though the project was done for each kind of category.

6 Conclusion

The project was carried out to show a use case of machine learning algorithms in a sample of open-source data. The project aimed to understand the concept, application, and workflow of machine learning both theoretically and practically. Besides, the thesis has explained the data analysis and data visualization tools and processes that were needed in the project.

The purpose of the thesis was to apply predictive analysis to the chosen sample data. The data was downloaded from Kaggle and processed by the Python language. Pandas was used to upload the data into the work platform on Jupyter Notebook. This was followed by a data cleaning and formatting process. Similarly, categorical columns were changed to numerical values as needed for the machine learning workflow. Also, a target array and a feature array were selected, and the data set was divided into the training and testing sets by using the ‘train_test_split’ method imported from the Scikit-learn library. Five different models – Decision Tree, KNN, logistic regression, random forest, and support vector machine – were imported and constructed as predictive models. The figures included in every section were taken as snip shots of supportive project programming code and content was searched during the thesis writing phase. After applying the test data on the five different models and evaluating the performance score, K-nearest-neighbor (KNN) and Logistic regression were found to be highly accurate models.

Analyzing the potentiality of ‘not.fully.paid’ by people belongs to the project case data, it is clear that machine learning provides a huge benefit to the banking or financial sector and in other similar issues. Machine learning is very easy to learn, and implement, and it is useful in the case of complex situations such as facial recognition and the case of simple techniques when solving real problems. The random forest model was one of the highly accurate models and it was chosen for the case project.

To sum up, the thesis was able to achieve the major objectives defined in the beginning and able to provide new insight. Also, the thesis describes the complete workflow of machine learning, including a proper theoretical background. The high accuracy of the model was found almost 84% and it can be applied to solving the predictive problem in a real-life scenario.

References

1. LendingClub [online].
   URL: https://en.wikipedia.org/wiki/LendingClub.
   Accessed 15 January 2020.

2. VanderPlas J. Python Data Science Handbook. Published by O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472, The United States of America [online]. 17 November 2016.
   URL: https://books.google.fi/books/about/Python_Data_Science_Handbook.htML?id=6omNDQAAQBAJ&printsec=frontcover&source=kp_read_button&redir_esc=y#v=onepage&q&f=false.
   Accessed 10 January 2020.

3. Introduction to Machine Learning: The Wikipedia Guide [online].
   URL: http://www.datascienceassn.org/sites/default/files/Introduction%20to%20Machine%20Learning.pdf.
   Accessed 20 February 2020.

4. Haffner P. What Is Machine Learning - and Why Is It Important? [online].
   7 July 2016.
   URL: https://www.interactions.com/blog/technology/machine-learning-important/.
   Accessed 19 November 2020.

5. Foote KD. A Brief History of Machine Learning[Online]. 26 March 2019.
   URL: https://www.dataversity.net/a-brief-history-of-machine-learning/.
   Accessed 27 November 2020.

6. Isha S. SuperVize Me: What's the Difference Between Supervised, Unsupervised, Semi-supervised and Reinforcement Learning Unsupervised Learning [online]. 2 August 2018.
   URL: https://blogs.nvidia.com/blog/2018/08/02/supervised-unsupervised-learning/.
   Accessed 12 October 2020.

7. Java T point. Classification Algorithms in Machine Learning [online].
   URL: https://www.javatpoint.com/classification-algorithm-in-machine-learning.
   Accessed on 14 August 2020.

8. Ranjan R. Overfitting and Underfitting in Machine Learning [online]. 20 April 2020.
   URL: https://towardsdatascience.com/overfitting-and-underfitting-in-machine-learning-89738c58f610.
   Accessed 22 October 2020.

9. Joshi P. Decision When to Use Linear Regression, Clustering, or Decision Trees [online]. 4 October 2017.
   URL: https://dzone.com/articles/decision-trees-vs-clustering-algorithms-vs-linear.
   Accessed 15 November 2020.

10. Python Documentation [online].
    URL: https://www.Python.org/doc/.
    Accessed 26 July 2020.

11. Pandas Official Documentation [online].
    URL: http://pandas.pydata.org/.
    Accessed 28 May 2020.

12. NumPy Official Documentation [online].
    URL: http://www.numpy.org/
    Accessed 16 April 2020.

13. Tawfik S. Data Preparation for Machine Learning: 5 Critical Steps to Ensure AI Success [online]. 25 September 2019.
    URL: https://blogs.informatica.com/2019/09/25/data-preparation-for-machine-learning-5-critical-steps-to-ensure-ai-success/.
    Accessed 10 December 2020.

14. Git hub. Accuracy comparison of different supervised machine algorithms [online]. 23 October 2020.
    URL: https://github.com/TikaramAcharya/Power-Bi-tool-project/blob/master/Machine_learning_project_thesis_topic.ipynb.
    Accessed on 27 October 2020.

15. The Ultimate Guide to Evaluation and Selection of Models in Machine Learning [online]. 2 November 2020.
    URL: https://neptune.ai/blog/the-ultimate-guide-to-evaluation-and-selection-of-models-in-machine-learning.
    Accessed 26 December 2020.

16. Packt. Confusion Matrix [online].
    URL:https://subscription.packtpub.com/book/big_data_and_business_intelligence/9781838555078/6/ch06lvl1sec34/confusion-matrix.
    Accessed on 5 January 2021.