***Group Members:***

Ankita Ghosh, Techno Engineering College Banipur, 172440110060

Atanu De, Bengal College of Engineering, 172600110019

Atul Kumar Singh, Bengal College of Engineering, 172600110021

Rahul Chakraborty, Techno India Salt Lake, 13001316083

Sarbodarshi Mitra, Techno Engineering College Banipur, 172440110091

Triasha Karmakar, Bengal Institute of Business Studies, PG-VUOAP01-MBA-IS-290

**Predicting Loan Repayment using Machine Learning Algorithm**

# **Table Of Contents**

* Acknowledgement
* Project Objective
* Data Description
* Data Analysis
* Code
* Future Scope of Improvements

# **ACKNOWLEDGEMENT**

I take this opportunity to express my profound gratitude and deep regards to my faculty Mr. Titas Roychowdhury for his exemplary guidance, monitoring and constant encouragement throughout the course of this project. The blessing, help and guidance given by him time to time shall carry me along way in the journey of life on which I am about to embark.

I am obliged to my project team members for the valuable information provided by them in their respective fields. I am grateful for their cooperation during the period of my assignment.

ANKITA GHOSH

# **ACKNOWLEDGEMENT**

I take this opportunity to express my profound gratitude and deep regards to my faculty Mr. Titas Roychowdhury for his exemplary guidance, monitoring and constant encouragement throughout the course of this project. The blessing, help and guidance given by him time to time shall carry me along way in the journey of life on which I am about to embark.

I am obliged to my project team members for the valuable information provided by them in their respective fields. I am grateful for their cooperation during the period of my assignment.

ATANU DE

# **ACKNOWLEDGEMENT**

I take this opportunity to express my profound gratitude and deep regards to my faculty Mr. Titas Roychowdhury for his exemplary guidance, monitoring and constant encouragement throughout the course of this project. The blessing, help and guidance given by him time to time shall carry me along way in the journey of life on which I am about to embark.

I am obliged to my project team members for the valuable information provided by them in their respective fields. I am grateful for their cooperation during the period of my assignment.

ATUL KUMAR SINGH

# **ACKNOWLEDGEMENT**

I take this opportunity to express my profound gratitude and deep regards to my faculty Mr. Titas Roychowdhury for his exemplary guidance, monitoring and constant encouragement throughout the course of this project. The blessing, help and guidance given by him time to time shall carry me along way in the journey of life on which I am about to embark.

I am obliged to my project team members for the valuable information provided by them in their respective fields. I am grateful for their cooperation during the period of my assignment

SARBODARSHI MITRA

# **ACKNOWLEDGEMENT**

I take this opportunity to express my profound gratitude and deep regards to my faculty Mr. Titas Roychowdhury for his exemplary guidance, monitoring and constant encouragement throughout the course of this project. The blessing, help and guidance given by him time to time shall carry me along way in the journey of life on which I am about to embark.

I am obliged to my project team members for the valuable information provided by them in their respective fields. I am grateful for their cooperation during the period of my assignment.

RAHUL CHAKRABORTY

# **ACKNOWLEDGEMENT**

I take this opportunity to express my profound gratitude and deep regards to my faculty Mr. Titas Roychowdhury for his exemplary guidance, monitoring and constant encouragement throughout the course of this project. The blessing, help and guidance given by him time to time shall carry me along way in the journey of life on which I am about to embark.

I am obliged to my project team members for the valuable information provided by them in their respective fields. I am grateful for their cooperation during the period of my assignment.

TRIASHA KARMAKAR

**INTRODUCTION & PROJECT OBJECTIVE**

There used to be a time not long ago when people used to be really scared of the word “ loan “ let alone thinking of applying for it.But gone are the days,in recent years people often consider loans as their go to option when eager to buy something let it be a house or a mobile phone. With the expansion of this industry came the risk of the lenders knowing the capability of a person to repay the loan taken, due to which many losses were invited. Thereby a new system was necessary to detect the capability of a bearer repaying the loan to predict it before giving them the loan in the first place.Thats what the project is all about.

**OBJECTIVE:**Using machine learning algorithm to develop a model that predicts the capability of a person to repay a loan if taken.

But before we get to technical part of the project we need to have a look at the basic parts first.

With a simple question like:

## 

## **What Is a Loan?**

Well A loan is the money you receive from a bank or financial institution in exchange for a commitment to repay the principal amount with interest.

Since lenders take the risk of a possible default, they charge a fee to offset this risk – and this fee is known as the interest.

Loans typically are secured or unsecured. In a secured loan, you need to pledge collateral to get the loan. So, if you default or do not pay back the loan, the lender has a right to take possession of the asset that had been pledged as collateral.

An unsecured loan doesn’t ask for collateral. If you do not pay back the unsecured loan, the lender has no right to take anything in return.

Common types of loans people take are home loans, car loans, personal loans, education loans, business loans, personal line of credit, debt consolidation loans, etc.

## **What Is Loan Repayment?**

Loan repayment is the act of paying back the borrowed money to the lender. The repayment occurs through a series of scheduled payments, also known as EMIs, which include both principal and interest.

## **How Loan Repayment Works?**

Loan repayment generally occurs through equated monthly installments (EMIs). These installments are the amount of money that is repaid to the lender every month. It is made up of two components – the principal amount and the interest on the principal amount, paid to the bank or lender on a fixed date each month until the total amount due is paid up over the loan tenure.

Now, you might assume that the principal and interest components are divided equally in an EMI. However, that’s not the case. In the initial loan period, the interest component in an EMI is higher. And in the latter period of the loan tenure, the interest component reduces, and the principal component gets higher.

Let’s explain this with an example.

## **Why Is Loan Repayment Important?**

Loan Repayment should be taken seriously because not only do they reduce your loan liability and interest accrued, they are also reflected in your credit history. The immediate financial implication would be anywhere from higher interest component (for missed instalment payments) to declaring bankruptcy (in the event of failing to repay altogether). There is also a long-term implication on your credit health which is reflected on your credit history.

## **Types of Loan Repayment Methods**

Listed below are some of the loan repayment options; however, the loan repayment option available to you may depend upon your lender and the type of loan that’s issued:

1. **EMIs** –Equated Monthly Instalments or EMIs, are the most popular loan repayment option. Every instalment involves a part of the principal and a part of the interest, which is scheduled to be paid every month over a fixed tenure.

That said, some banks allow their borrowers to repay the loan after a certain number of instalments have been made. Some banks may charge a prepayment fee, if you want to prepay your loan. Pre-payment can be done in two ways:

* **Partial or Part Pre-Paymen**t: This is when you pay off your loan in part, it helps you reduce the principal. This saves money on interest as the interest is applied on the new reduced principal.
* **Full Prepayment or Pre-Closure**: This is when you completely pay off your loan before the loan tenure.

2. **BULLET REPAYMENT** – Some loan products may allow the bearer to repay the loan through bullet loan repayment method. In this option, you need to pay only the interest component every month. When the loan tenure ends, you need to make one bullet repayment that pays off the entire principal loan.

## 

## **How Is Interest Calculated?**

The interest rate is the proportion of a loan that borrower pays in addition to the principal due. Think of it as the fee you pay to the lender for using its money. As with types of loans, there are many different flavors of interest rates offered:

### **1) Simple interest:**

The most clear-cut, simple rates are just multiplied to the principal at each payment period to find the interest due. For example, if you borrow rs2,000 from a family member and they ask for 5% interest when you repay them for the loan in a year, at the end of the repayment term you would owe them rs2100.

### **2)Compound:**

Common for credit cards and savings accounts, compound rates charge interest on the principal and on previously earned interest. For example, if you borrow 2,000rs at a rate of 5% over a year, you would owe $100 in interest in the first year. In the second year, you would owe 2,205rs, as you would calculate a 5% interest payment on 2,100rs that year.

### **3)Amortized:**

Amortized loans are designed so the borrower pays a larger amount of interest, rather than the principal, at the beginning of the loan. Over time the amount of principal in each payment will increase, whittling down the principal and amount of interest charged on the principal. While the payments due stay the same over the years, what the payment goes toward (principal vs. interest) shifts during the life of the loan. These are popular for car or home loans.

### **4)Fixed:**

A fixed interest rate will be defined upfront and stay the same over the term of the loan. This makes budgeting for payments predictable.

### **5)Variable:**

Variable (or adjustable) rates change over the life of the loan to reflect changes in the market interest rate. This means that the interest rate for the loan could go down or up over the term of your loan.

## 

## **How Does One's Credit History Impact Your Interest Rate?**

Before you can take out a loan, secured on unsecured, you first have to apply. Financial institutions and lenders will do a soft credit pull first to confirm you meet the minimum requirements to apply. If you move forward with an application, the lender will do a hard credit check to review your credit history.

If you want to review your own credit history you can request a credit report from one of the major credit agencies; Experian, Transunion, and Equifax. You can request a free report each year from each lender, so you can see what a lender will be reviewing.

Your creditworthiness will play a role in the interest rate offered. If you have a good credit score, the lender will have more peace of mind that you will repay your loan, and offer you a lower interest rate or maybe a larger amount of money. If you have a lower credit score you might want to build your score back up before submitting a loan application to see a better loan offer.

**Lender’s risk:**

All financing institutions know that they must accept a certain amount of lender risk when lending to factors, asset-based lenders, and entrepreneurs. But in the fast-paced environment of lending to factors and asset-based lenders, evaluating potential clients requires thorough investigations of their assets and company holdings to identify and mitigate potential lender risk. Because of the pace in which funds are dispersed by factors, it is incredibly important that factors understand everything at play in a company’s books before lending.

## **Lender’s Risk Factors**

There are several different categories of lender risk that factors must take into account before lending to a fellow factor. First, conducting basic background and financial checks can help to mitigate risk and avoid high-risk agreements.

Before entering an A/R lending agreement, factors should take multiple aspects of a company’s overall state of financial health and viability into consideration:

**Counterparty Credit Risk**

Counterparty risk is defined as the possibility that a debtor you do business with will be unable to meet the obligations that they have agreed to. If a debtor is unable to fulfill their obligations in some way, it is important that a plan is set in motion to mitigate and minimize losses. Counterparty risk can present a serious problem for factors and can be difficult to foresee due to its technical nature. Typically, A/R lenders see increases in counterparty risk when customers and those with outstanding invoices start behaving differently than they have in the past. They may begin paying late, avoid paying at all, or have sudden changes in their credit status. Counterparty risk is always present for factors, who should remain vigilant in identifying potential situations that could increase counterparty risk.

**Fraud Risk**

For factors, there is always the risk that a company you reach an agreement with may commit fraud in an effort to avoid paying the agreed upon amounts. Reducing the risk of fraud begins with evaluating the company’s character, but even that can only produce so much faith. For larger agreements, lenders may want to consider performing in-depth audits, as well as ensuring that your organization has the proper fraud insurance policies in place to mitigate risks.

According to an IFA Business Profile and Performance Survey for Factoring Firms, in the US, 83% of factors reported that they had encountered some sort of fraud within the last five years of operation. Only 17% of all factoring firms reported that they had never encountered fraud at all. There are a few ways in which factors typically see debtors committing fraud:

**Fake invoicing**

Creating invoices for services or products that were not actually delivered in an attempt to secure larger sums of cash from a lender. This is a common practice among fraudulent borrowers and can typically be spotted with an audit, or by digging deeper into their accounts receivable history. Fake invoicing is only worthwhile to fraudsters when done to facilitate large increases from a factor. Keep an eye out for clients with large invoices that are out of character for their clients, based on their history with a company.

**Misdirected payments**

Misdirected payment fraud typically takes place when a debtor instructs their clients to post their payments for products or services rendered to someone other than the lender to whom they sold their accounts receivable obligations. These misguided attempts are often easily spotted as the factor begins to reach out to parties to settle their invoices, only to find that they believe that they have already paid.

**Pre-invoicing**

Pre-invoicing is a very common form of fraud that factors deal with on a regular basis. This occurs when a company creates invoices for future products or services before they have been delivered and before they have officially reached an agreement with a company. In the case of a manufacturing company, they may create invoices for customers that have yet to place their order, but are planning on placing it in the near future. Pre-invoicing can also include real invoices that have been backdated to fall within the scope of the agreement with a factor.

There are multiple steps that any lender finance company can take to mitigate fraud and lender risk.

**How to solve the problem?**

**Data cleansing**

Data scientists spend a large amount of their time cleaning datasets and getting them down to a form with which they can work. ... In this project, we'll leverage Python's Pandas and NumPy libraries to clean data. We'll cover the following: Dropping unnecessary columns in a DataFrame. Changing the index of a DataFrame.

## 

## **What is Data Cleaning In Python?**

The meaning is rather simple than you must be thinking. Just as the two words suggest, data that has been collected for analysis is cleaned to get the relevant information out of it. The process of removing the kind of data that is incorrect or incomplete or duplicate and can affect the end results of the analysis is called data cleaning.

This does not mean that data cleaning is about the removal of certain kinds of irrelevant data. It is a process for ensuring dependability and increasing the accuracy of the data which has been collected.

## 

## **How To Do Data Cleaning in Python?**

Let’s take the example of a survey in which a particular form is filled by a number of people. Now, this data which has been entered by people is to be processed and there are good chances of finding some cases of this data being irrelevant or incomplete due to fields left blank or forms not filled at all.

But the data collected has to be processed and in order to avoid any further degradation of it, programs are written. One of the most preferred languages to do the task uses Python and let’s get back to the forms we were talking about in the example and learn how to run a python program. This will enable us to understand how to do data cleaning in Python much better.

Now, in a programming language, there are certain parameters to be filled and certain dependencies to be met to make sure the process if time-efficient as well. Already counting the factors in the picture, right?

So the parameters of the programming languages are called data types. Just like we categorize matter into solid, liquid and gas, Python also categorizes data entered into data types like integer, float, Boolean and others.

Once this classification is done, the first step towards building a Python program is completed. But are you thinking that how does a declaration of data types works? It works with the help of dependencies. There are generally called the libraries and contain the basic definition of all predefined terms of any programming language like Python.

Another aspect that comes into play while creating a program the size of it. Think of reading a book, would it be better divided into chapters or just continued text to interpret it better?

Similar to this, the codes for data cleaning in python can be stored into several files which are together called a module and then interpreted by software like Eclipse or Jupiter. They read the instructions mentioned in the Python program and apply them to the data collected to produce the accountable data.

**Analysis - which column is important**

## **Why Do We Care About Selecting Columns?**

In many standard data science examples, there are a relatively small number of columns

Real-life data sets are messy and often include a lot of extra (potentially unnecessary) columns.In data science problems you may need to select a subset of columns for one or more of the following reasons:

* Filtering the data to only include the relevant columns can help shrink the memory footprint and speed up data processing.
* Limiting the number of columns can reduce the mental overhead of keeping the data model in your head.
* When exploring a new data set, it might be necessary to break to task into manageable chunks.
* In some cases, you may need to loop through columns and perform calculations or cleanups in order to get the data in the format you need for further analysis.
* Your data may just contain extra or duplicate information which is not needed.

This brings us to the next topic,

## **Feature Importance**

Feature importance refers to a class of techniques for assigning scores to input features to a predictive model that indicates the relative importance of each feature when making a prediction.

Feature importance scores can be calculated for problems that involve predicting a numerical value, called regression, and those problems that involve predicting a class label, called classification.

The scores are useful and can be used in a range of situations in a predictive modeling problem, such as:

* Better understanding the data.
* Better understanding a model.
* Reducing the number of input features.

**Feature importance scores can provide insight into the dataset**. The relative scores can highlight which features may be most relevant to the target, and the converse, which features are the least relevant. This may be interpreted by a domain expert and could be used as the basis for gathering more or different data.

**Feature importance scores can provide insight into the model**. Most importance scores are calculated by a predictive model that has been fit on the dataset. Inspecting the importance score provides insight into that specific model and which features are the most important and least important to the model when making a prediction. This is a type of model interpretation that can be performed for those models that support it.

**Feature importance can be used to improve a predictive model**. This can be achieved by using the importance scores to select those features to delete (lowest scores) or those features to keep (highest scores). This is a type of feature selection and can simplify the problem that is being modeled, speed up the modeling process (deleting features is called dimensionality reduction), and in some cases, improve the performance of the model.

**Model building**

## **1. Steps of Data Exploration and Preparation:**

## Below are the steps involved to understand, clean and prepare your data for building your predictive model:

1. **Variable Identification**
2. **Univariate Analysis**
3. **Bi-variate Analysis**
4. **Missing values treatment**
5. **Outlier treatment**
6. **Variable transformation**
7. **Variable creation**

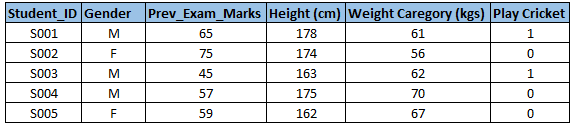
Finally, we will need to iterate over steps 4 – 7 multiple times before we come up with our refined model.

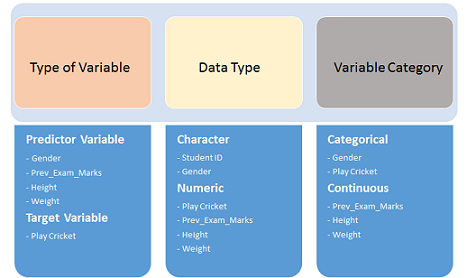
Let’s now study each stage in detail:-

**1.Variable Identification:**

First, identify **Predictor** (Input) and **Target** (output) variables. Next, identify the data type and category of the variables.

Let’s understand this step more clearly by taking an example.

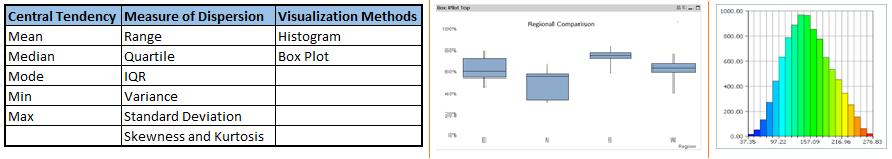
Example:- Suppose, we want to predict, whether the students will play cricket or not (refer below data set). Here you need to identify predictor variables, target variables, data type of variables and category of variables.Below, the variables have been defined in different category:



### **2.Univariate Analysis:**

At this stage, we explore variables one by one. Methods to perform univariate analysis will depend on whether the variable type is categorical or continuous. Let’s look at these methods and statistical measures for categorical and continuous variables individually:

**Continuous Variables:-** In case of continuous variables, we need to understand the central tendency and spread of the variable. These are measured using various statistical metrics visualization methods as shown below:

**Note:** Univariate analysis is also used to highlight missing and outlier values. In the upcoming part of this series, we will look at methods to handle missing and outlier values. To know more about these methods, you can refer to course descriptive statistics from Udacity.

**Categorical Variables:-** For categorical variables, we’ll use frequency tables to understand the distribution of each category. We can also read the percentage of values under each category. It can be measured using two metrics, **Count** and **Count%** against each category. Bar charts can be used as visualization.

### **3.Bi-variate Analysis:**

Bi-variate Analysis finds out the relationship between two variables. Here, we look for association and dissociation between variables at a pre-defined significance level. We can perform bi-variate analysis for any combination of categorical and continuous variables. The combination can be: Categorical & Categorical, Categorical & Continuous and Continuous & Continuous. Different methods are used to tackle these combinations during the analysis process.

Let’s understand the possible combinations in detail:

**Continuous & Continuous:** While doing bi-variate analysis between two continuous variables, we should look at the scatter plot. It is a nifty way to find out the relationship between two variables. The pattern of scatter plot indicates the relationship between variables. The relationship can be linear or nonlinear.

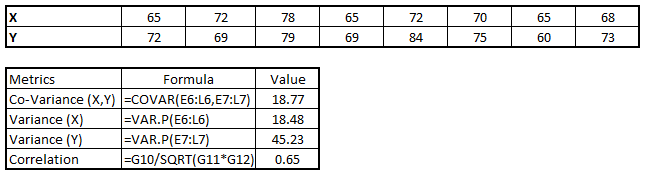
Scatter plot shows the relationship between two variables but does not indicate the strength of relationship amongst them. To find the strength of the relationship, we use Correlation. Correlation varies between -1 and +1.

* -1: perfect negative linear correlation
* +1:perfect positive linear correlation and
* 0: No correlation

Correlation can be derived using following formula:

**Correlation = Covariance(X,Y) / SQRT( Var(X)\* Var(Y))**

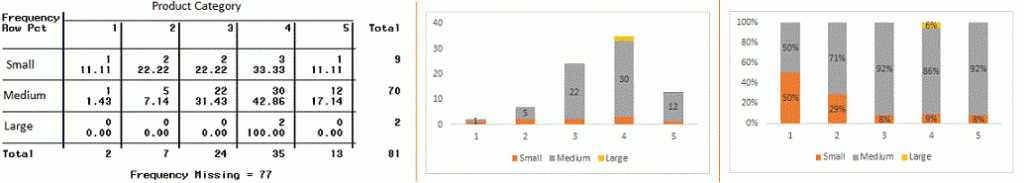
Various tools have function or functionality to identify correlation between variables. In Excel, function CORREL() is used to return the correlation between two variables and SAS uses procedure PROC CORR to identify the correlation. These function returns Pearson Correlation value to identify the relationship between two variables:



In the above example, we have a good positive relationship(0.65) between two variables X and Y.

**Categorical & Categorical:** To find the relationship between two categorical variables, we can use following methods:

* **Two-way table:** We can start analyzing the relationship by creating a two-way table of count and count%. The rows represent the category of one variable and the columns represent the categories of the other variable. We show count or count% of observations available in each combination of row and column categories.
* **Stacked Column Chart:** This method is more of a visual form of Two-way table.



* **Chi-Square Test:** This test is used to derive the statistical significance of relationship between the variables. Also, it tests whether the evidence in the sample is strong enough to generalize the relationship for a larger population as well. Chi-square is based on the difference between the expected and observed frequencies in one or more categories in the two-way table. It returns probability for the computed chi-square distribution with the degree of freedom.

Probability of 0: It indicates that both categorical variable are dependent

Probability of 1: It shows that both variables are independent.

Probability less than 0.05: It indicates that the relationship between the variables is significant at 95% confidence. The chi-square test statistic for a test of independence of two categorical variables is found by:

Data Exploration, Chi Square, Business Analyticswhere *O* represents the observed frequency. *E* is the expected frequency under the null hypothesis and computed by:

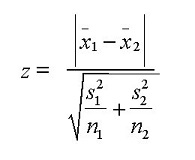
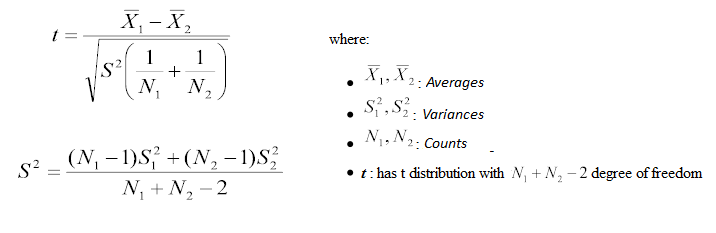
Data Exploration, Chi Square, Business Analytics

From the previous two-way table, the expected count for product category 1 to be of small size is 0.22. It is derived by taking the row total for Size (9) times the column total for Product category (2) then dividing by the sample size (81). This procedure is conducted for each cell. Statistical Measures used to analyze the power of relationship are:

* Cramer’s V for Nominal Categorical Variable
* Mantel-Haenszed Chi-Square for ordinal categorical variable.

Different data science languages and tools have specific methods to perform chi-square tests. In SAS, we can use **Chisq** as an option with **Proc freq** to perform this test.

**Categorical & Continuous:** While exploring relation between categorical and continuous variables, we can draw box plots for each level of categorical variables. If levels are small in number, it will not show statistical significance. To look at the statistical significance we can perform a Z-test, T-test or ANOVA.

* **Z-Test/ T-Test:-** Either test assess whether the mean of two groups are statistically different from each other or not.If the probability of Z is small then the difference of two averages is more significant. The T-test is very similar to Z-test but it is used when the number of observations for both categories is less than 30.  
  
* **ANOVA:-** It assesses whether the average of more than two groups is statistically different.

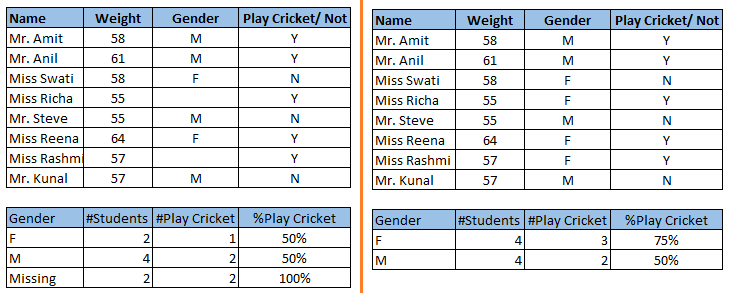
**Example:** Suppose, we want to test the effect of five different exercises. For this, we recruit 20 men and assign one type of exercise to 4 men (5 groups). Their weights are recorded after a few weeks. We need to find out whether the effect of these exercises on them is significantly different or not. This can be done by comparing the weights of the 5 groups of 4 men each.

Till here, we have understood the first three stages of Data Exploration, Variable Identification, Univariate and Bi-Variate analysis. We also looked at various statistical and visual methods to identify the relationship between variables.

Now, we will look at the methods of Missing values Treatment. More importantly, we will also look at why missing values occur in our data and why treating them is necessary.

## **2. Missing Value Treatment**

Missing data in the training data set can reduce the power / fit of a model or can lead to a biased model because we have not analysed the behavior and relationship with other variables correctly. It can lead to wrong predictions or classification.

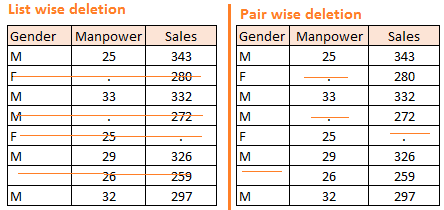


Notice the missing values in the image shown above: In the left scenario, we have not treated missing values. The inference from this data set is that the chances of playing cricket by males is higher than females. On the other hand, if you look at the second table, which shows data after treatment of missing values (based on gender), we can see that females have higher chances of playing cricket compared to males.

We looked at the importance of treatment of missing values in a dataset. Now, let’s identify the reasons for the occurrence of these missing values. They may occur at two stages:

1. **Data Extraction**: It is possible that there are problems with the extraction process. In such cases, we should double-check for correct data with data guardians. Some hashing procedures can also be used to make sure data extraction is correct. Errors at data extraction stage are typically easy to find and can be corrected easily as well.
2. **Data collection**: These errors occur at time of data collection and are harder to correct. They can be categorized in four types:
   * **Missing completely at random:** This is a case when the probability of missing a variable is the same for all observations. For example: respondents of the data collection process decide that they will declare their earnings after tossing a fair coin. If a head occurs, respondent declares his / her earnings & vice versa. Here each observation has an equal chance of missing value.
   * **Missing at random:** This is a case when a variable is missing at random and the missing ratio varies for different values / levels of other input variables. For example: We are collecting data for age and female has a higher missing value compared to male.
   * **Missing that depends on unobserved predictors:** This is a case when the missing values are not random and are related to the unobserved input variable. For example: In a medical study, if a particular diagnostic causes discomfort, then there is a higher chance of dropping out from the study. This missing value is not at random unless we have included “discomfort” as an input variable for all patients.
   * **Missing that depends on the missing value itself:** This is a case when the probability of missing value is directly correlated with missing value itself. For example: People with higher or lower income are likely to provide non-response to their earnings.

### **Which are the methods to treat missing values ?**

1. **Deletion:**  It is of two types: List Wise Deletion and PairWise Deletion.
   * In list wise deletion, we delete observations where any of the variables is missing. Simplicity is one of the major advantages of this method, but this method reduces the power of the model because it reduces the sample size.
   * In pairwise deletion, we perform analysis with all cases in which the variables of interest are present. Advantage of this method is, it keeps as many cases available for analysis. One of the disadvantages of this method, it uses different sample sizes for different variables.  
       
     
   * Deletion methods are used when the nature of missing data is “**Missing completely at random**” else non random missing values can bias the model output.
2. **Mean/ Mode/ Median Imputation:** Imputation is a method to fill in the missing values with estimated ones. The objective is to employ known relationships that can be identified in the valid values of the data set to assist in estimating the missing values. Mean / Mode / Median imputation is one of the most frequently used methods. It consists of replacing the missing data for a given attribute by the mean or median (quantitative attribute) or mode (qualitative attribute) of all known values of that variable. It can be of two types:-
   * **Generalized Imputation:** In this case, we calculate the mean or median for all non missing values of that variable then replace missing value with mean or median. Like in above table, variable “**Manpower”** is missing so we take average of all non missing values of “**Manpower”** (**28.33**) and then replace missing value with it.
   * **Similar case Imputation:** In this case, we calculate average for gender “**Male”** (29.75) and “**Female**” (25) individually of non missing values then replace the missing value based on gender. For “**Male**“, we will replace missing values of manpower with 29.75 and for “**Female**” with 25.
3. **Prediction Model**: Prediction model is one of the sophisticated methods for handling missing data. Here, we create a predictive model to estimate values that will substitute the missing data. In this case, we divide our data set into two sets: One set with no missing values for the variable and another one with missing values. First data set becomes the training data set of the model while the second data set with missing values is the test data set and the variable with missing values is treated as the target variable. Next, we create a model to predict target variables based on other attributes of the training data set and populate missing values of the test data set.We can use regression, ANOVA, Logistic regression and various modeling techniques to perform this. There are 2 drawbacks for this approach:
   * The model estimated values are usually more well-behaved than the true values
   * If there are no relationships with attributes in the data set and the attribute with missing values, then the model will not be precise for estimating missing values.
4. **KNN Imputation:** In this method of imputation, the missing values of an attribute are imputed using the given number of attributes that are most similar to the attribute whose values are missing. The similarity of two attributes is determined using a distance function. It is also known to have certain advantages & disadvantages.
   * **Advantages:**
     + k-nearest neighbour can predict both qualitative & quantitative attributes
     + Creation of predictive model for each attribute with missing data is not required
     + Attributes with multiple missing values can be easily treated
     + Correlation structure of the data is taken into consideration
   * **Disadvantage:**
     + KNN algorithm is very time-consuming in analyzing large databases. It searches through all the dataset looking for the most similar instances.
     + Choice of k-value is very critical. Higher value of k would include attributes which are significantly different from what we need whereas lower value of k implies missing out of significant attributes.

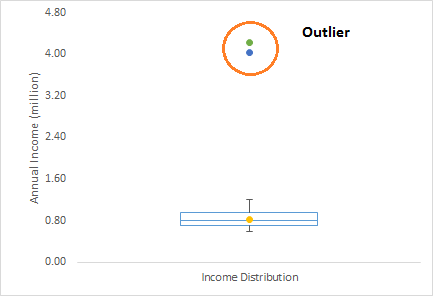
After dealing with missing values, the next task is to deal with outliers. Often, we tend to neglect outliers while building models. This is a discouraging practice. Outliers tend to make your data skewed and reduce accuracy. Let’s learn more about outlier treatment.

## **3. Techniques of Outlier Detection and Treatment**

## **What is an Outlier?**

Outlier is a commonly used terminology by analysts and data scientists as it needs close attention else it can result in wildly wrong estimations. Simply speaking, Outlier is an observation that appears far away and diverges from an overall pattern in a sample.

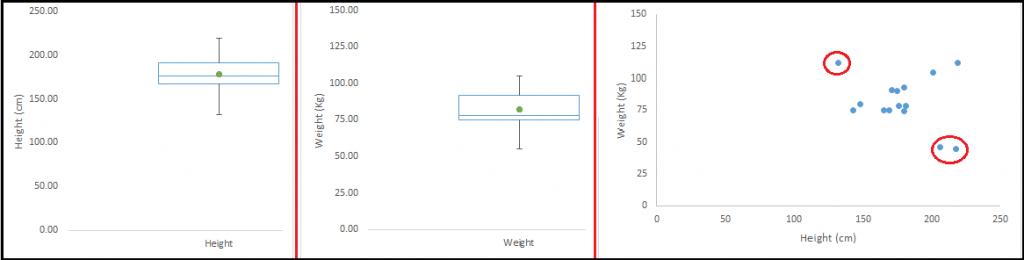
Let’s take an example, we do customer profiling and find out that the average annual income of customers is $0.8 million. But, there are two customers having annual income of $4 and $4.2 million. These two customers' annual income is much higher than the rest of the population. These two observations will be seen as Outliers.



### **What are the types of Outliers?**

Outlier can be of two types: **Univariate** and **Multivariate**. Above, we have discussed the example of a univariate outlier. These outliers can be found when we look at the distribution of a single variable. Multivariate outliers are outliers in an n-dimensional space. In order to find them, you have to look at distributions in multi-dimensions.

Let us understand this with an example. Let us say we are understanding the relationship between height and weight. Below, we have a univariate and bivariate distribution for Height, Weight. Take a look at the box plot. We do not have any outlier (above and below 1.5\*IQR, most common method). Now look at the scatter plot. Here, we have two values below and one above the average in a specific segment of weight and height.



### **What causes Outliers?**

Whenever we come across outliers, the ideal way to tackle them is to find out the reason for having these outliers. The method to deal with them would then depend on the reason for their occurrence. Causes of outliers can be classified in two broad categories:

1. **Artificial (Error) / Non-natural**
2. **Natural**.

Let’s understand various types of outliers in more detail:

* **Data Entry Errors:-** Human errors such as errors caused during data collection, recording, or entry can cause outliers in data. For example: Annual income of a customer is $100,000. Accidentally, the data entry operator puts an additional zero in the figure. Now the income becomes $1,000,000 which is 10 times higher. Evidently, this will be the outlier value when compared with the rest of the population.
* **Measurement Error:** It is the most common source of outliers. This is caused when the measurement instrument used turns out to be faulty. For example: There are 10 weighing machines. 9 of them are correct, 1 is faulty. Weight measured by people on the faulty machine will be higher / lower than the rest of people in the group. The weights measured on faulty machines can lead to outliers.
* **Experimental Error:** Another cause of outliers is experimental error. For example: In a 100m sprint of 7 runners, one runner missed out on concentrating on the ‘Go’ call which caused him to start late. Hence, this caused the runner’s run time to be more than other runners. His total run time can be an outlier.
* **Intentional Outlier*:*** This is commonly found in self-reported measures that involve sensitive data. For example: Teens would typically under report the amount of alcohol that they consume. Only a fraction of them would report actual value. Here actual values might look like outliers because the rest of the teens are under-reporting the consumption.
* **Data Processing Error:** Whenever we perform data mining, we extract data from multiple sources. It is possible that some manipulation or extraction errors may lead to outliers in the dataset.
* **Sampling error:** For instance, we have to measure the height of athletes. By mistake, we include a few basketball players in the sample. This inclusion is likely to cause outliers in the dataset.
* **Natural Outlier:** When an outlier is not artificial (due to error), it is a natural outlier. For instance: In my last assignment with one of the renowned insurance company, I noticed that the performance of top 50 financial advisors was far higher than rest of the population. Surprisingly, it was not due to any error. Hence, whenever we perform any data mining activity with advisors, we used to treat this segment separately.

**Precision, Recall and Confusion Matrix:**

Evaluation classification models have to be done very carefully.Accuracy is very important, but it might not be the best metric all the time.

Let’s have a dummy model that always predicts that a loan will not default.even if the accuracy of the model is 99 percent, the chances of a bank buying this model is really low.While our model has stunning accuracy, this is an apt example where accuracy is definitely not the right metric.

Now the question arises, If not accuracy, what else?

Along with accuracy, there are a bunch of other methods to evaluate the performance of a classification model

**1)CONFUSION MATRIX**

**2)PRECISION AND RECALL**

**3)ROC AND AUC**

Before moving forward, we will look into some terms which will be constantly repeated and might make the whole thing an incomprehensible maze if not understood clearly.

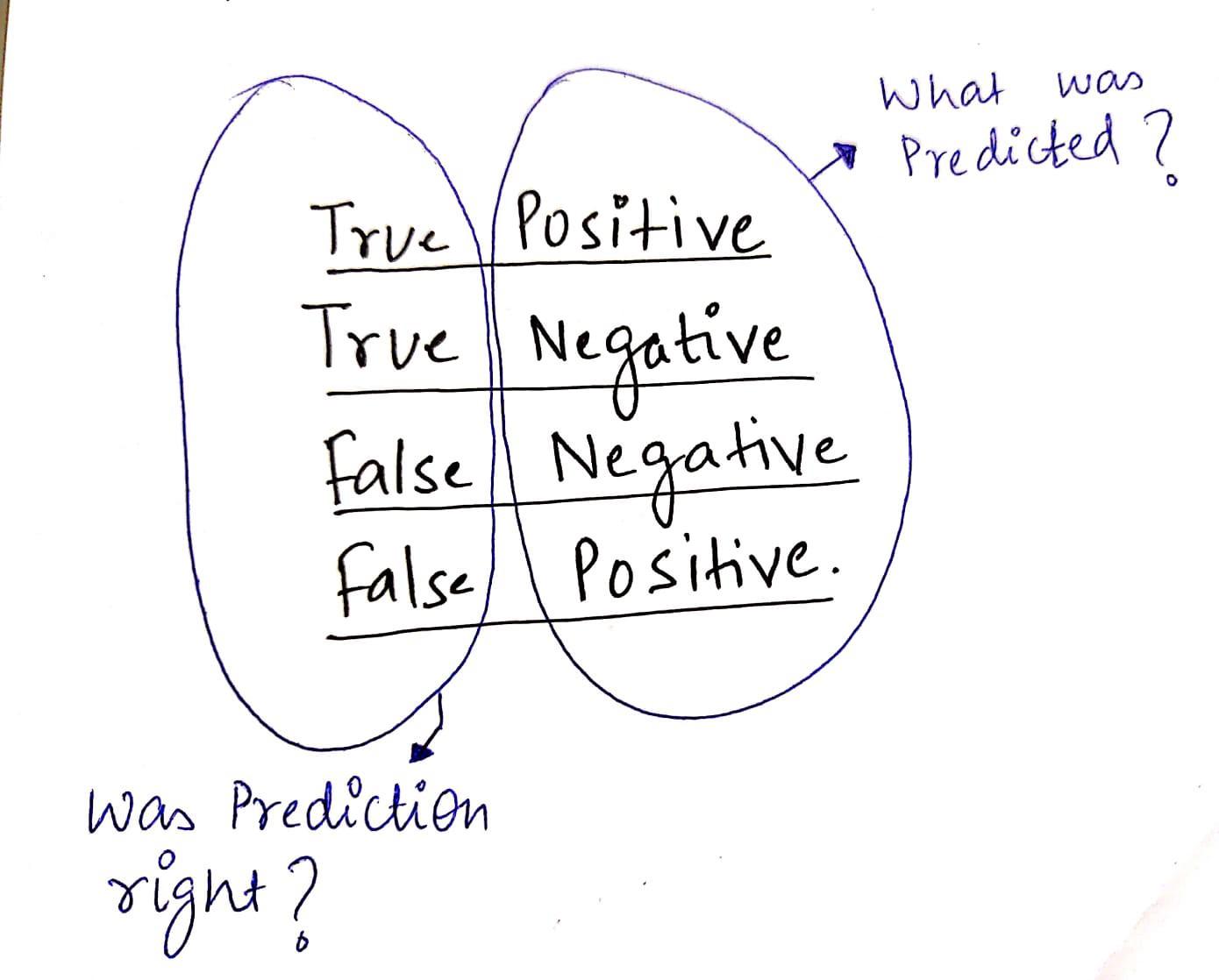
### 

### 

### **The Positives and Negatives — TP, TN, FP, FN**

***True Negative*: We were right when we predicted that a loan would not default.**

***False Positive*: We falsely predicted that a loan would default.**

****

**T**ypes of errors that can cause which are

**false positive**, where the model falsely detects someone capable of repaying the loan .

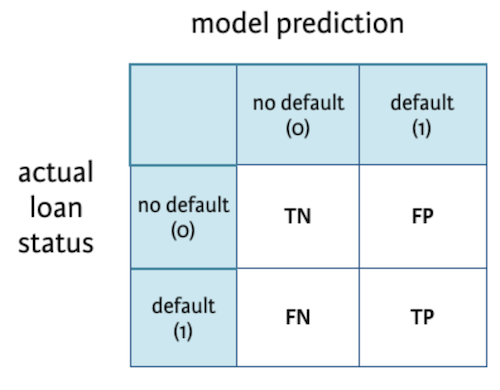
**false negative** , where the model falsely detects someone who can easily repay the loan as someone incapable.

### 

### ***Confusion Matrix:***

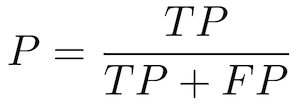
*As now we are familiar with TP, TN, FP, FN — It will be very easy to understand what a confusion matrix is.*

*It is a summary table showing how good our model is at predicting examples of various classes. Axes here are predicted-labels vs actual-labels.*

******

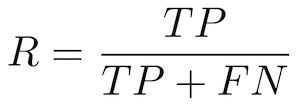
### **Precision and Recall:**

The ratio of correct positive predictions to the total predicted positives.

****

**Recall —** Also called Sensitivity, Probability of Detection, True Positive Rate

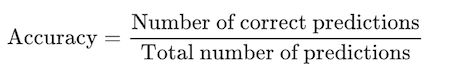
The ratio of correct positive predictions to the total positive examples.

******

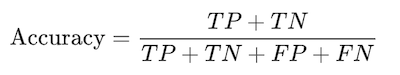
### 

### ***Accuracy:***

Accuracy is defined as the ratio of correctly predicted examples by the total examples.

******

In terms of confusion matrix it is given by:

******

**Why is recall matrices important in our case?**

In our case of predicting if a loan would default It would be better to have a high Recall as the banks don’t want to lose money and would be a good idea to alarm the bank even if there is a slight doubt about the defaulter.

Low precision, in this case, might be okay.

Mostly, we have to pick one over the other. It’s almost impossible to have both high Precision and Recall.

accuracy is a very useful metric when all the classes are equally important. But this might not be the case if we are predicting if a patient has cancer. In this example, we can probably tolerate FPs but not FNs.

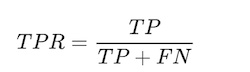
### **ROC curve:**

A ROC curve (receiver operating characteristic curve) graph shows the performance of a classification model at all classification thresholds.

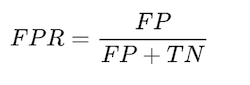
(Using thresholds: Say, if you want to compute TPR and FPR for the threshold equal to 0.7, you apply the model to each example, get the score, and, if the score if higher than or equal to 0.7, you predict the positive class; otherwise, you predict the negative class)

It plots 2 parameters:

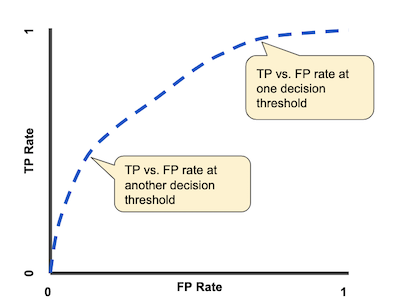
**True positive rate (Recall)**



**False Positive rate**



Tells what % of people who were not defaulters were identified as defaulters***.***

******

A typical ROC curve.

Lowering the classification threshold classifies more items as positive, thus increasing both False Positives and True Positives.

### **AUC:**

AUC stands for Area under the ROC Curve. It provides an aggregate measure of performance across all possible classification thresholds.

The higher the area under the ROC curve (AUC), the better the classifier. A perfect classifier would have an AUC of 1. Usually, if your model behaves well, you obtain a good classifier by selecting the value of the threshold that gives TPR close to 1 while keeping FPR near 0.

**DATA DESCRIPTION**

Here we got a manual data provided by our organization (globsyn kolkata) just for testing and achieving our project result.

The two most critical questions in the lending industry are:

1) How risky is the borrower?

2) Given the borrower’s risk, should we lend him/her?

The answer to the first question determines the interest rate the borrower would have. Interest rate measures among other things (such as time value of money) the riskness of the borrower, i.e. the riskier the borrower, the higher the interest rate.

With interest rate in mind, we can then determine if the borrower is eligible for the loan.

* Therefore, so we’ll address the second question indirectly by trying to predict if the borrower will repay the loan by its mature date or not. Through this excerise we’ll illustrate three modeling concepts:
* What to do with missing values.
* Techniques used with imbalanced classification problems.
* Illustrate how to build an ensemble model using two methods: blending and stacking, which most likely gives us a boost in performance.
* Below is a short description of each feature in the data set:
* **credit\_policy**: 1 if the customer meets the credit underwriting criteria of LendingClub.com, and 0 otherwise.
* **purpose**: The purpose of the loan such as: credit\_card, debt\_consolidation, etc.
* **int\_rate**: The interest rate of the loan (proportion).
* **installment**: The monthly installments ($) owed by the borrower if the loan is funded.
* **log\_annual\_inc**: The natural log of the annual income of the borrower.
* **dti**: The debt-to-income ratio of the borrower.
* **fico**: The FICO credit score of the borrower.
* **days\_with\_cr\_line**: The number of days the borrower has had a credit line.
* **revol\_bal**: The borrower’s revolving balance.
* **revol\_util**: The borrower’s revolving line utilization rate.
* **inq\_last\_6mths**: The borrower’s number of inquiries by creditors in the last 6 months.
* **delinq\_2yrs**: The number of times the borrower had been 30+ days past due on a payment in the past 2 years.
* **pub\_rec**: The borrower’s number of derogatory public records.
* **not\_fully\_paid**: indicates whether the loan was not paid back in full (the borrower either defaulted or the borrower was deemed unlikely to pay it back).

In the given data set when we analyze the data then there we have total of 5978 rows and 14 column.

Total 11 numeric data sets, and those are int.rate, installment, log.annual.inc, dti, fico, days.with.cr.line, revol.bal, revol.util, in.last.6mths, delinq.2yrs, pub.rec.

* 3 categorical data sets,credit.policy, puspose & not.fully.paid
* credit.policy int64
* purpose int64
* 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 float64
* delinq.2yrs float64
* pub.rec float64
* not.fully.paid int64
* purpose1 object
* for numeric data types the discrete types are fico, inq.last.6mths, delinq.2yrs, pub.rec and the continuous types are int.rate, installment, log.annual.inc, dti, days.with.cr.line, revol.bal, revol.util.
* for all categorical there labels are:

purpose - {'debt\_consolidation': 1,

'credit\_card': 2,

'all\_other': 3,

'home\_improvement': 4,

'small\_business': 5,

'major\_purchase': 6,

'educational': 7}

* From the given data set in the not.fully.paid colum positive case(1) means NOT FULLY PAID and negative case(0) means FULLY PAID.

And here the number of positive cases are 1533 and negative cases are 8045.

The proportion of positive to negative case is 19.06%.Percentage of positive case in the given data set is 16.01%.

* Percentage of Null data sets are :

log.annual.inc (4) = 0.0418%

days.with.cr.line (29) = 0.3028%

revol.util (62) = 0.6473%

inq.last.6mths (29) = 0.3028%

delinq.2yrs (29) = 0.3028%

pub.rec (29) = 0.3028%

After dropping null values we have 9516 rows and 14 column.

**Five-Number Statistics**

The five-number statistics, or 5-number statistics for short, is a non-parametric data summarization technique.

It is sometimes called the Tukey 5-number summary because it was recommended by John Tukey. It can be used to describe the distribution of data samples for data with any distribution.

*As a standard summary for general use, the 5-number summary provides about the right amount of detail.*

The five-number statistics involves the calculation of 5 summary statistical quantities: namely:

* **Median**: The middle value in the sample, also called the 50th percentile or the 2nd quartile.
* **1st Quartile**: The 25th percentile.
* **3rd Quartile**: The 75th percentile.
* **Minimum**: The smallest observation in the sample.
* **Maximum**: The largest observation in the sample.

A quartile is an observed value at a point that aids in splitting the ordered data sample into four equally sized parts. The median, or 2nd Quartile, splits the ordered data sample into two parts, and the 1st and 3rd quartiles split each of those halves into quarters.

A percentile is an observed value at a point that aids in splitting the ordered data sample into 100 equally sized portions. Quartiles are often also expressed as percentiles.

Both the quartile and percentile values are examples of rank statistics that can be calculated on a data sample with any distribution. They are used to quickly summarize how much of the data in the distribution is behind or in front of a given observed value. For example, half of the observations are behind and in front of the median of a distribution.

This is the mean values of each features in our data set.

credit.policy 0.804970

int.rate 0.122640

installment 319.089413

log.annual.inc 10.931874

dti 12.606679

fico 710.846314

days.with.cr.line 4562.026085

revol.bal 16913.963876

revol.util 46.865677

inq.last.6mths 1.571578

delinq.2yrs 0.163787

pub.rec 0.062101

not.fully.paid 0.160054

This the minimum values of each features -

credit.policy 0

purpose all\_other

int.rate 0.06

installment 15.67

log.annual.inc 7.5475

dti 0

fico 612

days.with.cr.line 178.958

revol.bal 0

revol.util 0

inq.last.6mths 0

delinq.2yrs 0

pub.rec 0

not.fully.paid 0

dtype: object

This the maximum values of each features -

credit.policy 1

purpose small\_business

int.rate 0.2164

installment 940.14

log.annual.inc 14.5284

dti 29.96

fico 827

days.with.cr.line 17640

revol.bal 1207359

revol.util 119

inq.last.6mths 33

delinq.2yrs 13

pub.rec 5

not.fully.paid 1

so by analyzing the mean,max and min value of the features of each data set ,now we can understand the ranges of values of each column in our given data set.

Quantile plays a very important role in Statistics when one deals with the Normal Distribution.

When quantile=25%-

credit.policy 1.000000

int.rate 0.103900

installment 163.770000

log.annual.inc 10.558414

dti 7.212500

fico 682.000000

days.with.cr.line 2820.000000

revol.bal 3187.000000

revol.util 22.700000

inq.last.6mths 0.000000

delinq.2yrs 0.000000

pub.rec 0.000000

not.fully.paid 0.000000

When quantile=75%-

credit.policy 1.000000

int.rate 0.140700

installment 432.762500

log.annual.inc 11.289832

dti 17.950000

fico 737.000000

days.with.cr.line 5730.000000

revol.bal 18249.500000

revol.util 71.000000

inq.last.6mths 2.000000

delinq.2yrs 0.000000

pub.rec 0.000000

not.fully.paid 0.000000

In the given data set quantile function return values at the given quantile over requested axis.It divides the whole values into eqal frequency distribution , each containing the same fraction of the total datasets.

**DATA ANALYSIS**

What is EDA?

Exploratory Data Analysis (EDA) is an approach to analyse the data, to summarize its characteristics, often with visual methods. Every machine learning problem solving starts with EDA. It is probably the most important part of a machine learning project.

In this project we have performed EDA using the most popular programming languages python.

Why EDA?

With data growing in large amounts from day to day, business organisations want to make proper use of data, so that they can make strategic decisions by studying the data. So in this project we have got a large amount of data. To understand and comprehend the data we have used various graphs and methods.

According to **Tukey** (data analysis in 1961)

*“Procedures for analysing data, techniques for interpreting the results of such procedures, ways of planning the gathering of data to make its analysis easier, more precise or more accurate, and all the machinery and results of (mathematical) statistics which apply to analysing data.*

Here we have used two types of method to comprehend our data.

* **Univariate visualization** — provides summary statistics for each field in the raw data set.
* **Bivariate visualization** — is performed to find the relationship between each variable in the dataset and the target variable of interest.

### **Univariate visualization**

Univariate data visualization plots help us comprehend the enumerative properties as well as a descriptive summary of the particular data variable. These plots help in understanding the **location/position** of observations in the data variable, its **distribution**, and **dispersion**.

Here we have used univariate visualization through Boxplot and histogram on the continuous columns to understand the given data set.

1. **Boxplot**

A [boxplot](https://www.statology.org/boxplots/) is a plot that shows the five-number summary of a dataset.

The five-number summary includes:

* The minimum value
* The first quartile
* The median value
* The third quartile
* The maximum value

**2. Histogram**

A histogram is a type of chart that uses vertical bars to display frequencies. This type of chart is a useful way to visualize the distribution of values in a dataset.

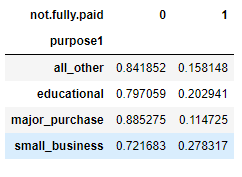
**3.Countplot**

Here we have used univariate visualization through countplot on the categorical columns to understand the given data set.

**seaborn.countplot()** method is used to Show the counts of observations in each categorical bin using bars.

The count plot is a univariate data visualization plot on a two-dimensional axis. One axis is the category axis indicating the category, while the second axis is the value axis that shows the numeric value of that category.

Here we used countplot in the fico column and it shows count value of fully paid which is 0 first increases and then decreses with respect to fico columns values.And the count values of not fully paid which is 1 are very low than the count value of fully paid.So the count plot helps us to visualize the fico column values and understand the data.



**4.Crosstab**

This method is used to compute a simple cross-tabulation of two (or more) factors. By default, computes a frequency table of the factors unless an array of values and an aggregation function are passed.

purpose column as this is a categorical column.Firstly we replaced all the categorical values with 1,2,3,4…..values.And then found the unique values of the column.Now we applied crosstab with respect to the target column and as a result we got many columns of same values for 0(fully paid) .

**Bivariate visualization**

Bivariate analysis is an analysis that is performed to determine the relationship between 2 variables. In this analysis, two measurements were made for each observation.

In bivariate analysis we have used many types of feature selection method.

Why Feature Selection is important in ML ?

This process reduces physical intervention in data analysis. It makes the feature interpretation easy and ready to use. The technique helps us to select the most targeted variable correlating with other variables. This reduces the dimension of the set and improves the accuracy of the selected features. Hence the model performance is increased with the selected features.

Firstly we applied here ANOVA feature selection process. **An**alysis **o**f **Va**riance(ANOVA) is a statistical method, used to check the means of two or more groups that are significantly different from each other.

We checked the relationship between categorical predictor vs continuous response.So after applying anova method to the continues columns we found the columns in this respective order,

days.with.cr.line

log.annual.inc

installment

revol.bal

dti

revol.util

int rate.

And we found discrete columns in the respective order,

Purpose

inq.last.6mths

pub.rec

credit policy

fico

delinq.2yrs

So here we selected first three columns from conitinuous feature selection process and two column from discrete feature selection process.After applying this columns to the main program(the program where we applied different type of Algorithm like Random Forest Algorithm,Decision Tree Algorithm,Logistic Regression Algorithm,K-nearest neighbours algorithm in the selective columns).And we get some result which was not so good.

Now we need to get rid of the extra value of the column Pub.rec

Because we already seen through value\_count method there are too much low values which we can easily attach and make shorter dataset which will be legible.

0.0 8960

1.0 530

2.0 19

3.0 5

5.0 1

4.0 1

After applying algorithm we make the pub.rec column more readible and the result is -

1. 8960
2. 556

And for the same reason we also made credit.policy and inq.last.6mths columns shorter.After reducing the data the new name of inq.last.6mths and pub.rec column are inq and pub.rec1 respectively.

So For betterment the result of Recall value we applied Outlier detection Method.In this case we need to know first –

What is outlier?

Outliers are the values that look different from the other values in the data. Below is a plot highlighting the outliers in ‘red’ and outliers can be seen in both the extremes of data.

# **Reasons for outliers in data**

1. Errors during data entry or a faulty measuring device (a faulty sensor may result in extreme readings).
2. Natural occurrence (salaries of junior level employees vs C-level employees)

# **Problems caused by outliers**

1. Outliers in the data may causes problems during model fitting (esp. linear models).
2. Outliers may inflate the error metrics which give higher weights to large errors (example, mean squared error, RMSE).

If we able to remove those outliers from the data then we may able to find some important column in determining the target.

Here we applied the getoutlier method in int.rate column as it is a continues column.

So after observing the result we got rid of some outlying observation in the Inq column .

After that we applied Random Forest Classiier for Feature importance to get rid of the extra columns. And we got the following columns-

inq

purpose1

credit.policy

delinq.2yrs

pub.rec1

For this columns also we are not getting the good result of recall score.

So now we applied permutation combination through Random Forest Classifier.But it’s also not helped too much to improve the score of Recall value.

Finally we chose the best five columns which are giving comparetively better result in Random Forest Algorithm and those columns are –

delinq.2yrs

installment

purpose1

inq

pub.rec1

We also applied different types of algorithm but we got the best result in Random Forest Classifier Algorithm.

So we got Recall score for training data set 95% and for testing data set 35%.

We also checked auc,accuracy and precision scores and their

scores are comparatively far more better than recall score.

The final result we got after analysing –

Training Testing

Accuracy Score- 98% 78%

Precision Score- 94% 67%

Recall Score- 95% 35%

Area under the curve score-96% 57%

**CODE**

1 . import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

from sklearn import linear\_model

from sklearn import model\_selection

from sklearn import metrics

df=pd.read\_csv("d:/datasets/loans.csv")

df[:5]

df.shape

df.dtypes

df.info()

pos = df[df["not.fully.paid"] == 1].shape[0]

neg = df[df["not.fully.paid"] == 0].shape[0]

df["not.fully.paid"].value\_counts()/df.shape[0]

pos

neg

per=(pos / neg) \* 100

round(per,2)

q=df.shape[0]

perc=(pos / q) \* 100 # percentege of positive case in given dataset is

round(perc,2)

df.isnull().sum()

df.dropna()

df.shape()

df.describe()

plt.figure(figsize=(8, 6))

sns.countplot(df["not.fully.paid"])

plt.xticks((0, 1), ["Paid fully", "Not paid fully"])

plt.xlabel("")

plt.ylabel("Count")

plt.title("Class counts", y=1, fontdict={"fontsize": 20});

*Here we have analyze the data set and understand the columns of the the data set.*

2. df.purpose.value\_counts()

pd.crosstab(index=df.purpose,columns=df["not.fully.paid"],normalize="index")

ser=pd.Series(np.where(df["purpose"]=="credit\_card","major\_purchase",df["purpose"]),index=df.index)

ser=pd.Series(np.where(df["purpose"]=="home\_improvement","all\_other",ser),index=ser.index)

ser=pd.Series(np.where(df["purpose"]=="debt\_consolidation","all\_other",ser),index=ser.index)

pd.crosstab(ser,df["not.fully.paid"],normalize="index")

df["purpose1"]=ser.values

df["credit.policy"].value\_counts()

df["inq.last.6mths"].value\_counts()

ser=pd.Series(np.where(df["inq.last.6mths"]>=3.0,3.0,df["inq.last.6mths"]),index=df.index)

df["inq.last.6mths1"]=ser.values

df["inq.last.6mths1"].value\_counts()

df["installment"].value\_counts()

sns.countplot(y="fico",hue="not.fully.paid",data=df)

df["pub.rec"].value\_counts()

arr=np.where(df["pub.rec"]>=1,1,0)

ser=pd.Series(arr,index=df.index)

df["pub.rec1"]=ser.values

df["pub.rec1"].value\_counts()

*Here we analyse each columns of the data set and reduced the values of some of the columns, which helps us to calculate and analyse the data set and get our desired result.*

3. X=df[

["credit.policy","purpose1","inq","delinq.2yrs","pub.rec1",["int.rate","installment","log.annual.inc","dti","days.with.cr.line","revol.bal","revol.util”] ]

y=df["not.fully.paid"]

Xtrain,Xtest,ytrain,ytest=model\_selection.train\_test\_split(X,y,test\_size=.25,random\_state=42)

*Here we generalize the data by dividing it to the training and testing data sets.*

4. def getoutliers(ser,type="both"):

iqr=ser.quantile(q=.75)-ser.quantile(q=.25)

thup=ser.quantile(q=.75)+1.5\*iqr

thlow=ser.quantile(q=.25)-1.5\*iqr

if type=="upper":

return ser[ser>thup].index.values

if type=="both":

return ser[(ser>thup) | (ser<thlow)].index.values

if type=="lower":

return ser[ser<thlow].index.values

arr=getoutliers(Xtrain["inq"],type="both")

print(arr)

from sklearn.feature\_selection import SelectFromModel

from sklearn import ensemble

model=ensemble.RandomForestClassifier(n\_estimators = 100)

model1=SelectFromModel(model,max\_features=5)

model1.fit(Xtrain,ytrain)

model1.get\_support()

selcol= Xtrain.columns[(model1.get\_support())]

*In our program we tried many feature selection process but at last we found some of them were useful to get the better result which have given in the above.*

5. selcol=df[["delinq.2yrs","installment","purpose1","inq","pub.rec1"]]

X=selcol

y=df["not.fully.paid"]

Xtrain,Xtest,ytrain,ytest=model\_selection.train\_test\_split(X,y,test\_size=.25,random\_state=42)

*So finally we selected some of the features which helped us to get the better of value of recall(specially) & precision , Auc & accuracy score.*

6. from sklearn import linear\_model

from sklearn import model\_selection

from sklearn import utils

from sklearn import metrics

from sklearn import tree

from sklearn.feature\_selection import RFECV

from sklearn import feature\_selection

from sklearn import naive\_bayes

from sklearn import neighbors

from sklearn import ensemble

def modelstats1(Xtrain,Xtest,ytrain,ytest):

stats=[]

modelnames=["LR","DecisionTree","KNN","NB","RF"]

models=list()

models.append(linear\_model.LogisticRegression())

models.append(tree.DecisionTreeClassifier())

models.append(neighbors.KNeighborsClassifier())

models.append(naive\_bayes.GaussianNB())

models.append(ensemble.RandomForestClassifier())

for name,model in zip(modelnames,models):

if name=="KNN":

k=[l for l in range(5,17,2)]

grid={"n\_neighbors":k}

grid\_obj=model\_selection.GridSearchCV(estimator=model,param\_grid=grid,scoring="f1")

grid\_fit=grid\_obj.fit(Xtrain,ytrain)

model =grid\_fit.best\_estimator\_

model.fit(Xtrain,ytrain)

name=name+"("+str(grid\_fit.best\_params\_["n\_neighbors"])+")"

else:

model.fit(Xtrain,ytrain)

trainprediction=model.predict(Xtrain)

testprediction=model.predict(Xtest)

scores=list()

scores.append(name+"-train")

scores.append(metrics.accuracy\_score(ytrain,trainprediction))

scores.append(metrics.precision\_score(ytrain,trainprediction))

scores.append(metrics.recall\_score(ytrain,trainprediction))

scores.append(metrics.roc\_auc\_score(ytrain,trainprediction))

stats.append(scores)

scores=list()

scores.append(name+"-test")

scores.append(metrics.accuracy\_score(ytest,testprediction))

scores.append(metrics.precision\_score(ytest,testprediction))

scores.append(metrics.recall\_score(ytest,testprediction))

scores.append(metrics.roc\_auc\_score(ytest,testprediction))

stats.append(scores)

colnames=["MODELNAME","ACCURACY","PRECISION","RECALL","AUC"]

return pd.DataFrame(stats,columns=colnames)

def modelstats2(Xtrain,Xtrainohe,Xtest,Xtestohe,ytrain,ytest):

stats=[]

modelnames=["LR","DecisionTree","KNN","NB","RF"]

models=list()

models.append(linear\_model.LogisticRegression())

models.append(tree.DecisionTreeClassifier())

models.append(neighbors.KNeighborsClassifier())

models.append(naive\_bayes.GaussianNB())

models.append(ensemble.RandomForestClassifier())

for name,model in zip(modelnames,models):

if name=="KNN":

k=[l for l in range(5,17,2)]

grid={"n\_neighbors":k}

grid\_obj=model\_selection.GridSearchCV(estimator=model,param\_grid=grid,scoring="f1")

grid\_fit=grid\_obj.fit(Xtrain,ytrain)

model =grid\_fit.best\_estimator\_

model.fit(Xtrain,ytrain)

name=name+"("+str(grid\_fit.best\_params\_["n\_neighbors"])+")"

elif name=="LR":

model.fit(Xtrainohe,ytrain)

trainprediction=model.predict(Xtrainohe)

testprediction=model.predict(Xtestohe)

else:

model.fit(Xtrain,ytrain)

trainprediction=model.predict(Xtrain)

testprediction=model.predict(Xtest)

scores=list()

scores.append(name+"-train")

scores.append(metrics.accuracy\_score(ytrain,trainprediction))

scores.append(metrics.precision\_score(ytrain,trainprediction))

scores.append(metrics.recall\_score(ytrain,trainprediction))

scores.append(metrics.roc\_auc\_score(ytrain,trainprediction))

stats.append(scores)

scores=list()

scores.append(name+"-test")

scores.append(metrics.accuracy\_score(ytest,testprediction))

scores.append(metrics.precision\_score(ytest,testprediction))

scores.append(metrics.recall\_score(ytest,testprediction))

scores.append(metrics.roc\_auc\_score(ytest,testprediction))

stats.append(scores)

colnames=["MODELNAME","ACCURACY","PRECISION","RECALL","AUC"]

return pd.DataFrame(stats,columns=colnames)

modelstats1(Xtrain,Xtest,ytrain,ytest)

*This is the main algorithm where we have added different types of algorithm(Random Forest,Logistic Regression,KNN,Naïve bayes..etc.) altogether and checked which algorithm is working good for our data set.And finally we got that Random Forest algorithm works good for the given data set.*

**FUTURE SCOPE OF IMPROVEMENT**

## **Support Vector Machine:**

The objective of the support vector machine algorithm is to find a hyper plane in an N-dimensional space (N — the number of features) that distinctly classifies the data points.

To separate the two classes of data points, there are many possible hyper planes that could be chosen. Our objective is to find a plane that has the maximum margin, i.e. the maximum distance between data points of both classes. Maximizing the margin distance provides some reinforcement so that future data points can be classified with more confidence.

Support vector machine is an elegant and powerful algorithm.



Importance:

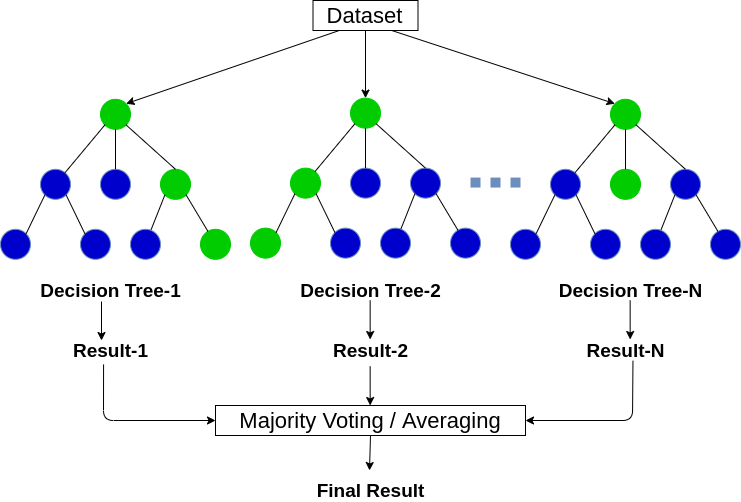
SVM is a supervised **machine** learning algorithm which can be used for classification or **regression** problems. It uses a technique called the kernel trick to transform your data and then based on these transformations it finds an optimal boundary between the possible outputs.

**Random Forest:**

Ensemble model made of many **decision** trees using bootstrapping, **random** subsets of features, and average voting to make predictions. This is an **example** of a bagging ensemble.

**Random Forest** is a **great** algorithm, for both classification and regression problems, to produce a predictive model. Its default hyper parameters already return **great** results and the system is **great** at avoiding over fitting. Moreover, it is a pretty good indicator of the importance it assigns to your features.

The most common answer I get is that the **Random Forest** are so **called** because each tree in the **forest** is built by **randomly** selecting a sample of the data. ... Another important paper that Leo refers to is **called** “The **random** Subspace Method for constructing **Decision Forest**” by Tin Kan Ho.

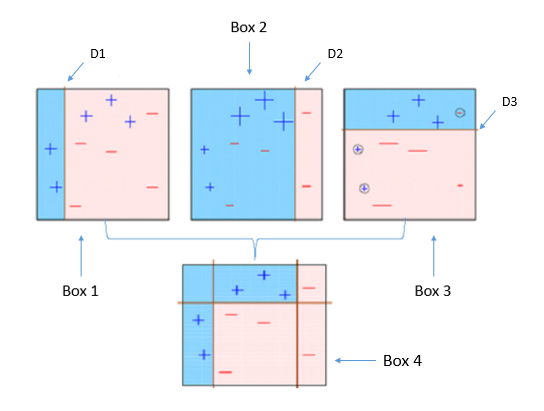


 Importance**:**

It is a set of **Decision** Trees. Each **Decision** Tree is a set of internal nodes and leaves. In the internal node, the selected feature is used to make **decision** how to divide the data set into two separate sets with similar responses within.

**AdaBoost** :

Adaboost is one of the first boosting algorithms to be adapted in solving practices. **Adaboost** helps you combine multiple “weak classifiers” into a single “strong classifier”. Here are some (fun) facts about **Adaboost** is the weak learners in **AdaBoost** are decision trees with a single split, called decision stumps.



Importance:

**AdaBoost** can be used to boost the performance of any machine learning algorithm. It is best used with weak learners. These are models that achieve accuracy just above random chance on a classification problem. The most suited and therefore most common algorithm used with **AdaBoost** is decision trees with one level.

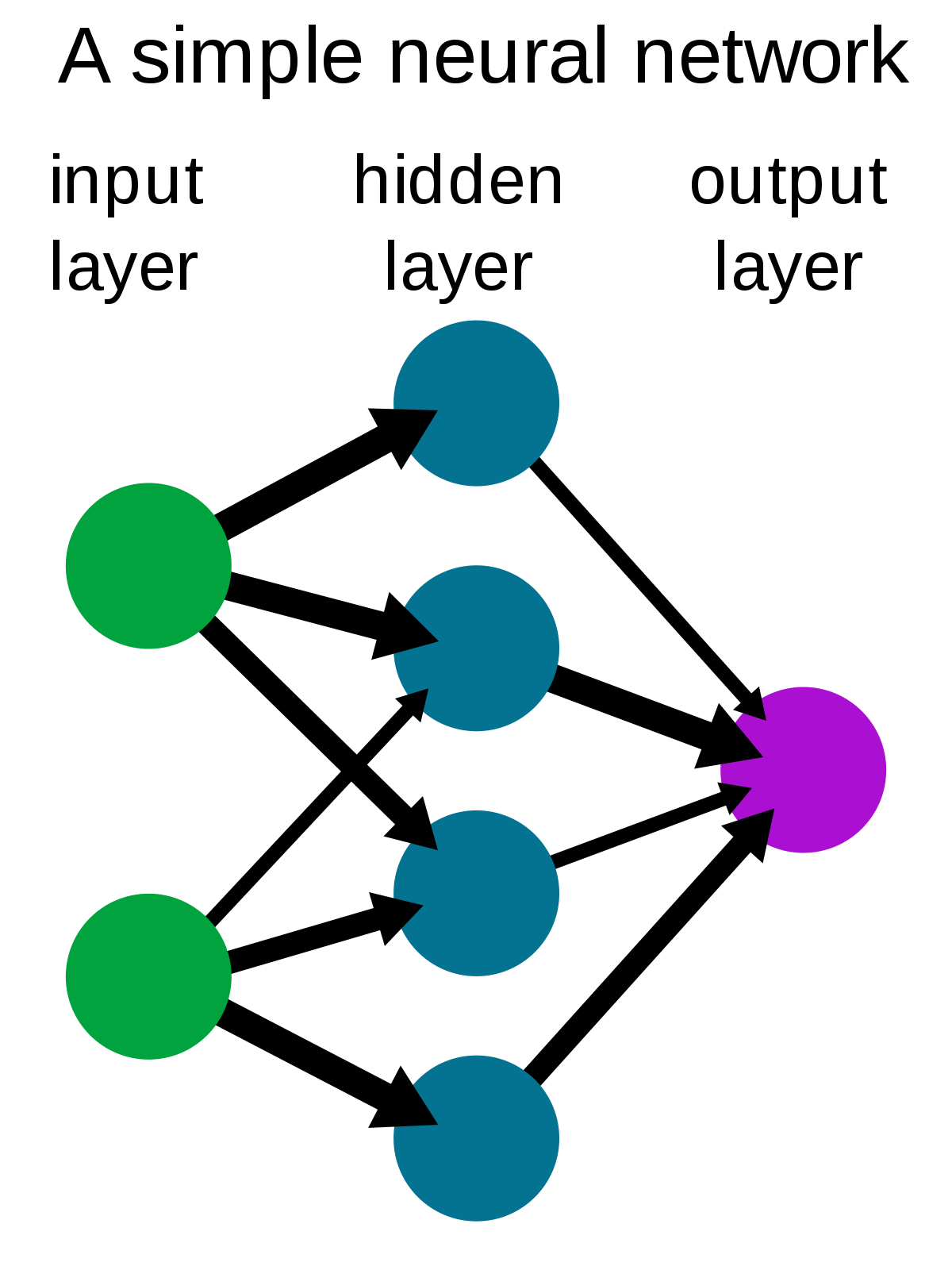
**Gradient Boosting:**

**Gradient boosting** is a machine learning technique for regression and classification problems, which produces a prediction model in the form of an ensemble of weak prediction models, typically decision trees.

Importance:

Generally, **importance** provides a score that indicates how useful or valuable each feature was in the construction of the **boosted** decision trees within the model. The more an attribute is used to make key decisions with decision trees, the higher its relative **importance**.

**Neural Network:**

A **neural network** is a series of algorithms that endeavors to recognize underlying relationships in a set of data through a process that mimics the way the human brain operates. In this sense, **neural networks** refer to systems of neurons, either organic or artificial in nature.

Importance:

Their main aim is to solve complex problems like pattern recognition or facial recognition, and several other applications include -- speech-to-text transcription, data analysis, handwriting recognition for check processing, weather prediction, and signal processing.