APPENDIX I (Cover Page)

A PROJECT REPORT

ON

MACHINE LEARNING ALGORITHMS

For

EISYSTEM SERVICES

By

NEERAJ SHARMA

IN INTERNSHIP PROGRAMME OF DATA ANALYTICS AND MACHINE LEARNING(BATCH-2)

MADAN MOHAN MALVIYA UNIVERSITY OF TECHNOLOGY(2018-2022)

GORAKHPUR,UTTAR PRADESH

Acknowledgements

A summer project is a golden opportunity for learning and self development. I consider myself very lucky and honored to have so many wonderful people lead me through in completion of this project.. My grateful thanks to (EISYSTEMS , Mr. Chandan Verma,Head of Training, Computer Science Domain,Eisystems Services) who in spite of being extraordinarily busy with her/his duties, took time out to hear, guide and keep me on the correct path. . I do not know where I would have been without him. A humble ‘Thank you’ Sir. Mr. Mayur Dev Sewak,General Manager, Operations,Eisystems Services (monitored my progress and arranged all facilities to make life easier.. I choose this moment to acknowledge his contribution gratefully. Prof. Mr. Chandan Verma,Head of Training, Computer Science Domain,Eisystems Services whose patience I have probably tested to the limit. He was always so involved in the entire process, shared his knowledge, and involved in the entire process, shared his knowledge, and encouraged me to think. Thank you, Dear Sir. I would like to thanks Technex faculty members for their efforts and help provided to me to get such an excellent opportunity.. Last but not the least there were so many who shared valuable information that helped in the successful completion of this project.

NEERAJ SHARMA

BTECH 2 YEAR

COMPUTER SCIENCE AND ENGINEERING

MADAN MOHAN MALVIYA UNIVERSITY OF TECHNOLOGY,GORAKHPUR,U.P

**Executive Summary**

Abstract

In this project, we were asked to experiment with a real world dataset, and to explore how machine learning algorithms can be used to find the patterns in data. We were expected to gain experience using a common data-mining and machine learning library, scikit,pandas,numpy,matplotlib and seaborn, and were expected to submit a report about the dataset and the algorithms used. After performing the required tasks on a dataset of my choice, herein lies my final report.

Keywords: Machine Learning, Classification, Supervised learning, Unsupervised learning

APPENDIX 2 (Title Page)

MACHINE LEARNING ALGORITHM

For

EISYSTEM SERVICES

IN INTERNSHIP PROGRAMME OF DATA ANALYTICS AND MACHINE LEARNING(BATCH-2)

SUBMITTED BY

NEERAJ SHARMA

MADAN MOHAN MALVIYA UNIVERSITY OF TECHNOLOGY(2018-2022)

GORAKHPUR,UTTAR PRADESH

Introduction:

Machine learning is a sub-domain of computer science which evolved from the study of pattern recognition in data, and also from the computational learning theory in artificial intelligence. It is the first-class ticket to most interesting careers in data analytics today[1]. As data sources proliferate along with the computing power to process them, going straight to the data is one of the most straightforward ways to quickly gain insights and make predictions. Machine Learning can be thought of as the study of a list of sub-problems, viz: decision making, clustering, classification, forecasting, deep-learning, inductive logic programming, support vector machines, reinforcement learning, similarity and metric learning, genetic algorithms, sparse dictionary learning, etc. Supervised learning, or classification is the machine learning task of inferring a function from a labeled data [2]. In Supervised learning, we have a training set, and a test set. The training and test set consists of a set of examples consisting of input and output vectors, and the goal of the supervised learning algorithm is to infer a function that maps the input vector to the output vector with minimal error. In an optimal scenario, a model trained on a set of examples will classify an unseen example in a correct fashion, which requires the model to generalize from the training set in a reasonable way. In layman’s terms, supervised learning can be termed as the process of concept learning, where a brain is exposed to a set of inputs and result vectors and the brain learns the concept that relates said inputs to outputs. A wide array of supervised machine learning algorithms are available to the machine learning enthusiast, for example Neural Networks, Decision Trees, Support Vector Machines, Random Forest, Naïve Bayes Classifier, Bayes Net, Majority Classifier[4,7,8,9] etc., and they

each have their own merits and demerits. There is no single algorithm that works for all cases, as merited by the No free lunch theorem [3]. In this project, we try and find patterns in a dataset [2], which is a sample of males in a heart-disease high risk region of South Africa, and attempt to throw various intelligently-picked algorithms at the data, and see what sticks. Problems and Issues in Supervised learning: Before we get started, we must know about how to pick a good machine learning algorithm for the given dataset. To intelligently pick an algorithm to use for a supervised learning task, we must consider the following factors [4]:

1. Heterogeneity of Data: Many algorithms like neural networks and support vector machines like their feature vectors to be homogeneous numeric and normalized. The algorithms that employ distance metrics are very sensitive to this, and hence if the data is heterogeneous, these methods should be the afterthought. Decision Trees can handle heterogeneous data very easily.

2. Redundancy of Data: If the data contains redundant information, i.e. contain highly correlated values, then it’s useless to use distance based methods because of numerical instability. In this case, some sort of Regularization can be employed to the data to prevent this situation.

3. Dependent Features: If there is some dependence between the feature vectors, then algorithms that monitor complex interactions like Neural Networks and Decision Trees fare better than other algorithms.

4. Bias-Variance Tradeoff: A learning algorithm is biased for a particular input x if, when trained on each of these data sets, it is systematically incorrect when predicting the correct output for x, whereas a learning algorithm has high variance for a particular input x if it predicts different output values when trained on different training sets. The prediction error of a learned classifier can be related to the sum of bias and variance of the learning algorithm, and neither can be high as they will make the prediction error to be high. A key feature of machine learning algorithms is that they are able to tune the balance between bias and variance automatically, or by manual tuning using bias parameters, and using such algorithms will resolve this situation.

5. Curse of Dimensionality: If the problem has an input space that has a large number of dimensions, and the problem only depends on a subspace of the input space with small dimensions, the machine learning algorithm can be confused by the huge number of dimensions and hence the variance of the algorithm can be high. In practice, if the data scientist can manually remove irrelevant features from the input data, this is likely to improve the accuracy of the learned function. In addition, there are many algorithms for feature selection that seek to identify the relevant features and discard the irrelevant ones, for instance Principle Component Analysis for unsupervised learning. This reduces the dimensionality.

6. Overfitting: The programmer should know that there is a possibility that the output values may constitute of an inherent noise which is the result of human or sensor errors. In this case, the algorithm must not attempt to infer the function that exactly matches all the data. Being too careful in fitting the data can cause overfitting, after which the model will answer perfectly for all training examples but will have a very high error for unseen samples. A practical way of preventing this is stopping the learning process prematurely, as well as applying filters to the data in the pre-learning phase to remove noises.

**Overview of the Workflow of ML**

Understanding the machine learning workflow

We can define the machine learning workflow in 3 stages.

1. Gathering data
2. Data pre-processing
3. Researching the model that will be best for the type of data
4. Training and testing the model
5. Evaluation

Okay but first let’s start from the basics

**What is the machine learning Model?**

The machine learning model is nothing but a piece of code; an engineer or data scientist makes it smart through training with data. So, if you give garbage to the model, you will get garbage in return, i.e. the trained model will provide false or wrong predictions.

**1. Gathering Data**

The process of gathering data depends on the type of project we desire to make, if we want to make an ML project that uses real-time data, then we can build an IoT system that using different sensors data. The data set can be collected from various sources such as a file, database, sensor and many other such sources but the collected data cannot be used directly for performing the analysis process as there might be a lot of missing data, extremely large values, unorganized text data or noisy data. Therefore, to solve this problem Data Preparation is done.

**2. Data pre-processing**

Data pre-processing is one of the most important steps in machine learning. It is the most important step that helps in building machine learning models more accurately. In machine learning, there is an 80/20 rule. Every data scientist should spend 80% time for data pre-processing and 20% time to actually perform the analysis.

As we know that data pre-processing is a process of cleaning the raw data into clean data, so that can be used to train the model. So, we definitely need data pre-processing to achieve good results from the applied model in machine learning and deep learning projects.

Most of the real-world data is messy, some of these types of data are:

1. **Missing data:** Missing data can be found when it is not continuously created or due to technical issues in the application (IOT system). 2. **Noisy data:** This type of data is also called outliners, this can occur due to human errors (human manually gathering the data) or some technical problem of the device at the time of collection of data. 3. **Inconsistent data:** This type of data might be collected due to human errors (mistakes with the name or values) or duplication of data.

**Three Types of Data**

1. Numeric e.g. income, age 2. Categorical e.g. gender, nationality 3. Ordinal e.g. low/medium/high

**How can data pre-processing be performed?**

These are some of the basic pre — processing techniques that can be used to convert raw data.

1. **Conversion of data:** As we know that Machine Learning models can only handle numeric features, hence categorical and ordinal data must be somehow converted into numeric features.

2. **Ignoring the missing values:** Whenever we encounter missing data in the data set then we can remove the row or column of data depending on our need. This method is known to be efficient but it shouldn’t be performed if there are a lot of missing values in the dataset.

3. **Filling the missing values:** Whenever we encounter missing data in the data set then we can fill the missing data manually, most commonly the mean, median or highest frequency value is used.

4.**Machine learning:** If we have some missing data then we can predict what data shall be present at the empty position by using the existing data.

5. **Outliers detection:** There are some error data that might be present in our data set that deviates drastically from other observations in a data set. [Example: human weight = 800 Kg; due to mistyping of extra 0] **3. Researching the model that will be best for the type of data**

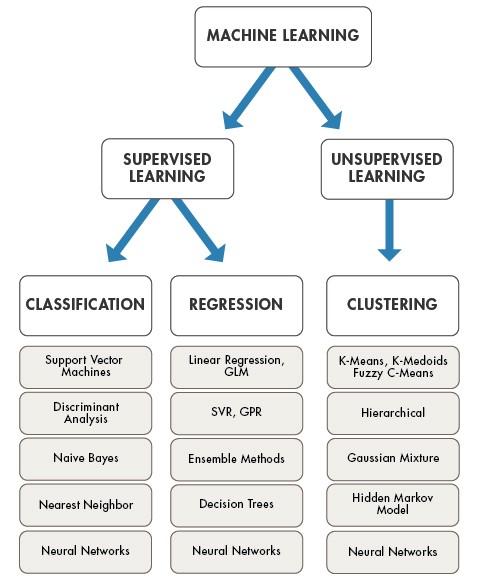
Our main goal is to train the best performing model possible, using the pre-processed data.

These some most used classification algorithms.

* **K-Nearest Neighbor**
* **Naive Bayes**
* **Decision Trees/Random Forest**
* **Support Vector Machine**
* **Logistic Regression**Clustering

Methods used for clustering are:

* **Gaussian mixtures**
* **K-Means Clustering**
* **Boosting**
* **Hierarchical Clustering**
* **K-Means Clustering**
* **Spectral Clustering**

**Overview of models under categories**

**4. Training and testing the model on data**

For training a model we initially split the model into 3 three sections which are ‘**Training data**’ ,‘**Validation data**’ and ‘**Testing data**’.You train the classifier using ‘**training data set**’, tune the parameters using ‘**validation set**’ and then test the performance of your classifier on unseen ‘**test data set**’. An important point to note is that during training the classifier only the training and/or validation set is available. The test data set must not be used during training the classifier. The test set will only be available during testing the classifier.

**Training set:** The training set is the material through which the computer learns how to process information. Machine learning uses algorithms to perform the training part. A set of data used for learning, that is to fit the parameters of the classifier.

**Validation set:** Cross-validation is primarily used in applied machine learning to estimate the skill of a machine learning model on unseen data. A set of unseen data is used from the training data to tune the parameters of a classifier.

**Test set:** A set of unseen data used only to assess the performance of a fully-specified classifier.

We can find out the accuracy of the model using the confusion matrix.

*Accuracy = (True Positives +True Negatives) / (Total number of classes)* **5. Evaluation**

Model Evaluation is an integral part of the model development process. It helps to find the best model that represents our data and how well the chosen model will work in the future.

To improve the model we might tune the hyper-parameters of the model and try to improve the accuracy and also looking at the confusion matrix to try to increase the number of true positives and true negatives.

**Conclusion**

we have discussed the workflow a Machine learning project and gives us a basic idea of how a should the problem be tackled.

**LIST OF CONTENTS**

SNO. ALGORITHM PG NO.

|  |  |  |
| --- | --- | --- |
| 1 | KMEANS | 16 |
| 2 | LINEAR REGRESSION | 20 |
| 3 | LOGISTIC REGRESSION | 24 |
| 4 | NAÏVE BAYES | 27 |
| 5 | SVM | 30 |
| 6 | KNN | 34 |

1.KMEANS ALGORITHM:-

**Kmeans** algorithm is an iterative algorithm that tries to partition the dataset into *K*pre-defined distinct non-overlapping subgroups (clusters) where each data point belongs to **only one group**. It tries to make the intra-cluster data points as similar as possible while also keeping the clusters as different (far) as possible. It assigns data points to a cluster such that the sum of the squared distance between the data points and the cluster’s centroid (arithmetic mean of all the data points that belong to that cluster) is at the minimum.

**Applications**

kmeans algorithm is very popular and used in a variety of applications such as market segmentation, document clustering, image segmentation and image compression, etc.

**ALGORITHM:-**

1. Specify number of clusters *K*.
2. Initialize centroids by first shuffling the dataset and then randomly selecting *K*data points for the centroids without replacement.
3. Keep iterating until there is no change to the centroids. i.e assignment of data points to clusters isn’t changing.

* Compute the sum of the squared distance between data points and all centroids.
* Assign each data point to the closest cluster (centroid).
* Compute the centroids for the clusters by taking the average of the all data points that belong to each cluster.

Data set:-cars.csv

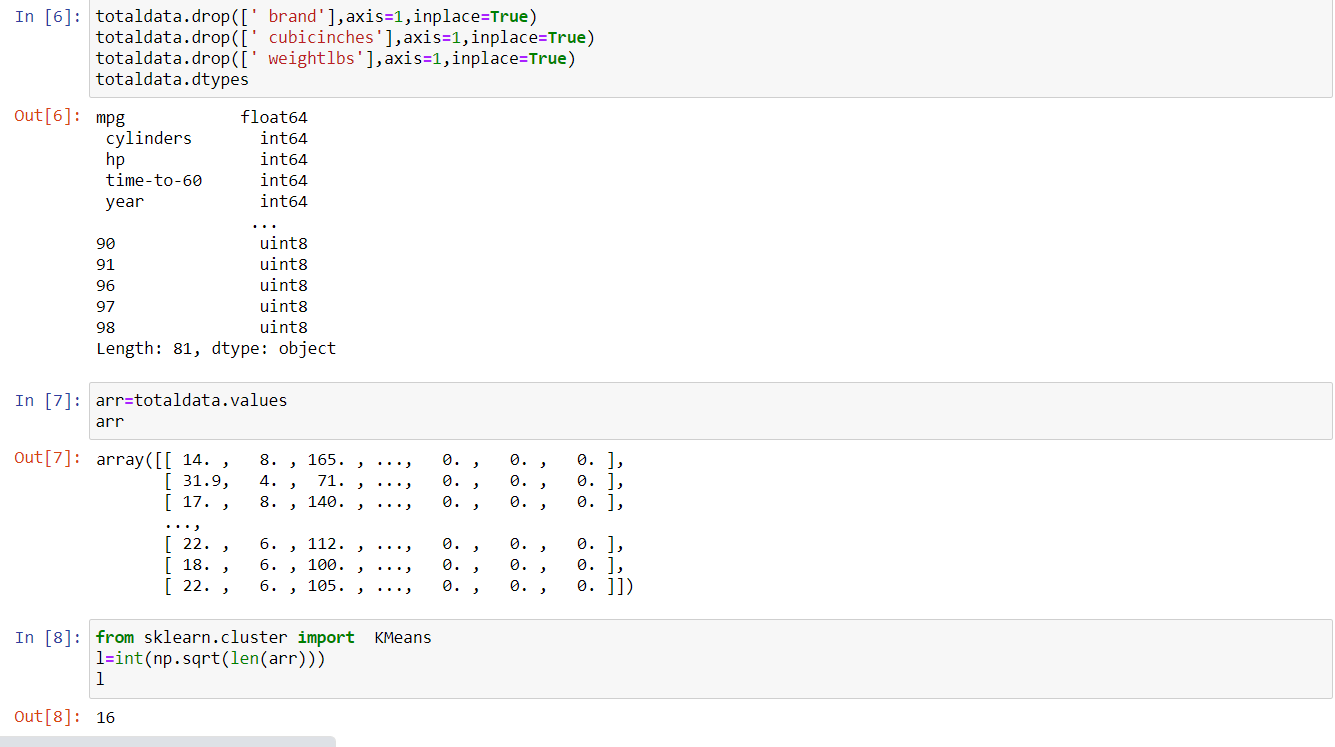
Jupyter file link:-

https://drive.google.com/file/d/15pGA4a6Pv-oLFrLHt3IiKVkYiMKFf1Fs/view?usp=sharing

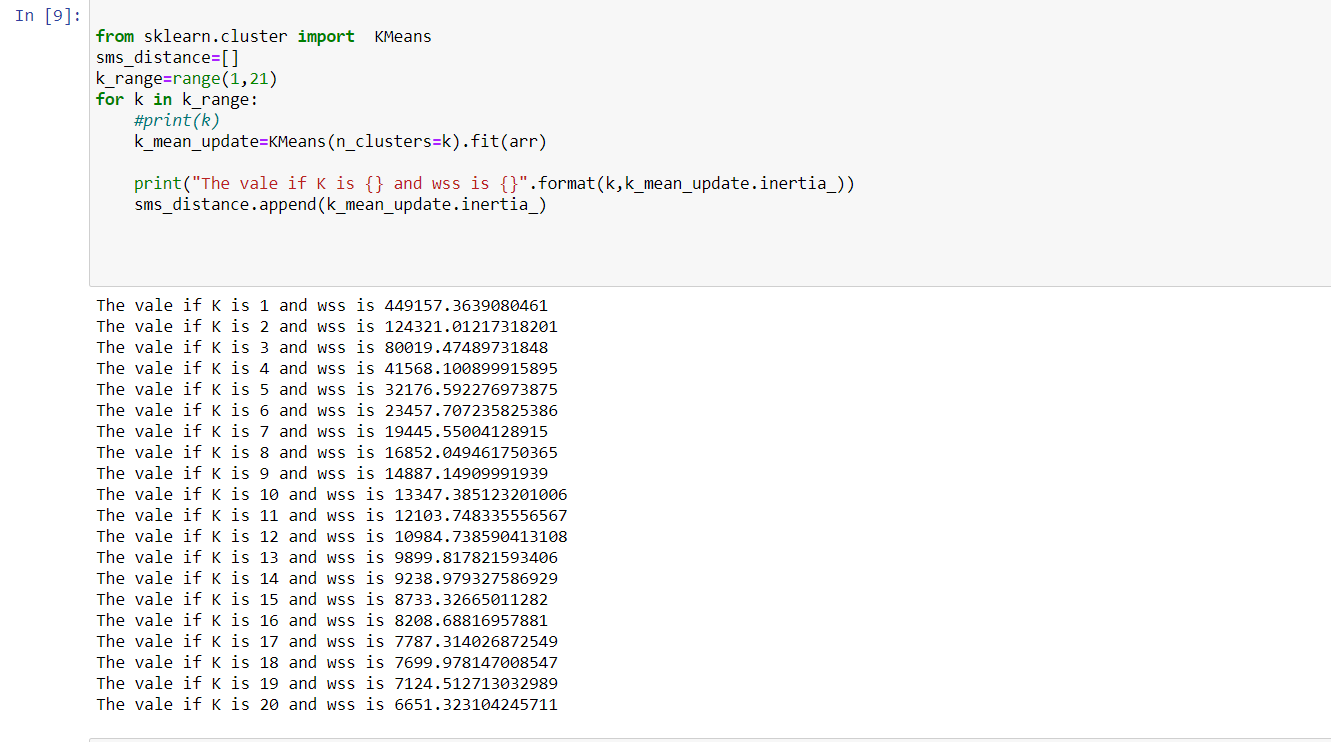
1.LOAD THE DATA



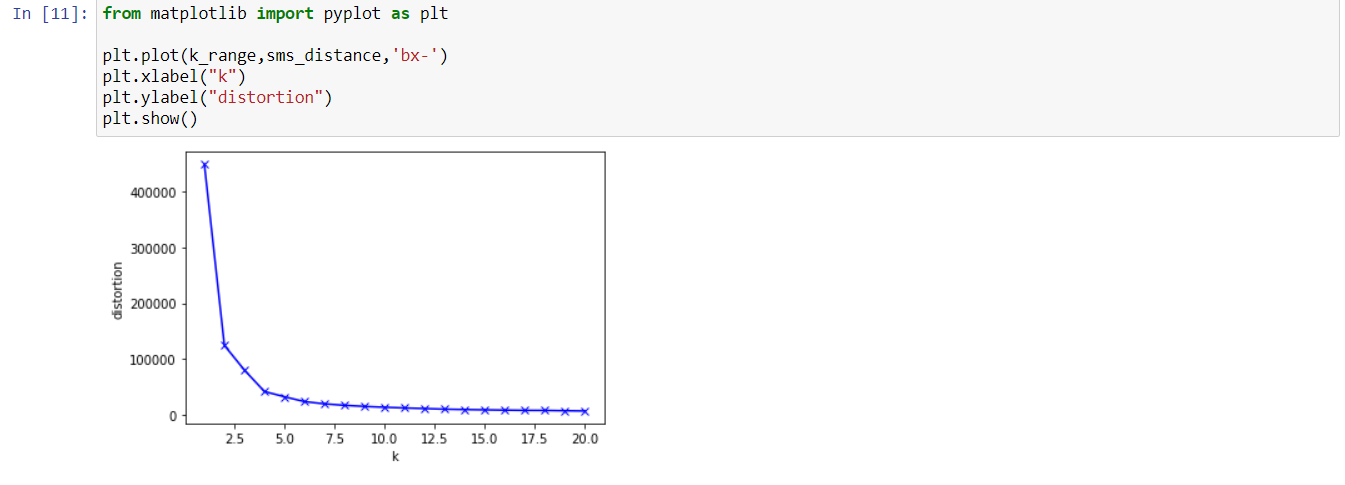
2.DATA PREPROCESSING



3.KMEAN ALGO FITTING FOR DIFFERENT K VALUES AND THEIR CORRESPONDING ACCURACIES



4.GRAPH PLOT :-K VS DISTORTION



**2.LINEAR REGRESSION**

Linear regression performs the task to predict a dependent variable value (y) based on a given independent variable (x). So, this regression technique finds out a linear relationship between x (input) and y(output).

The core idea is to obtain a line that best fits the data. The best fit line is the one for which total prediction error (all data points) are as small as possible. Error is the distance between the point to the regression line.

Y(pred) = b0 + b1\*x

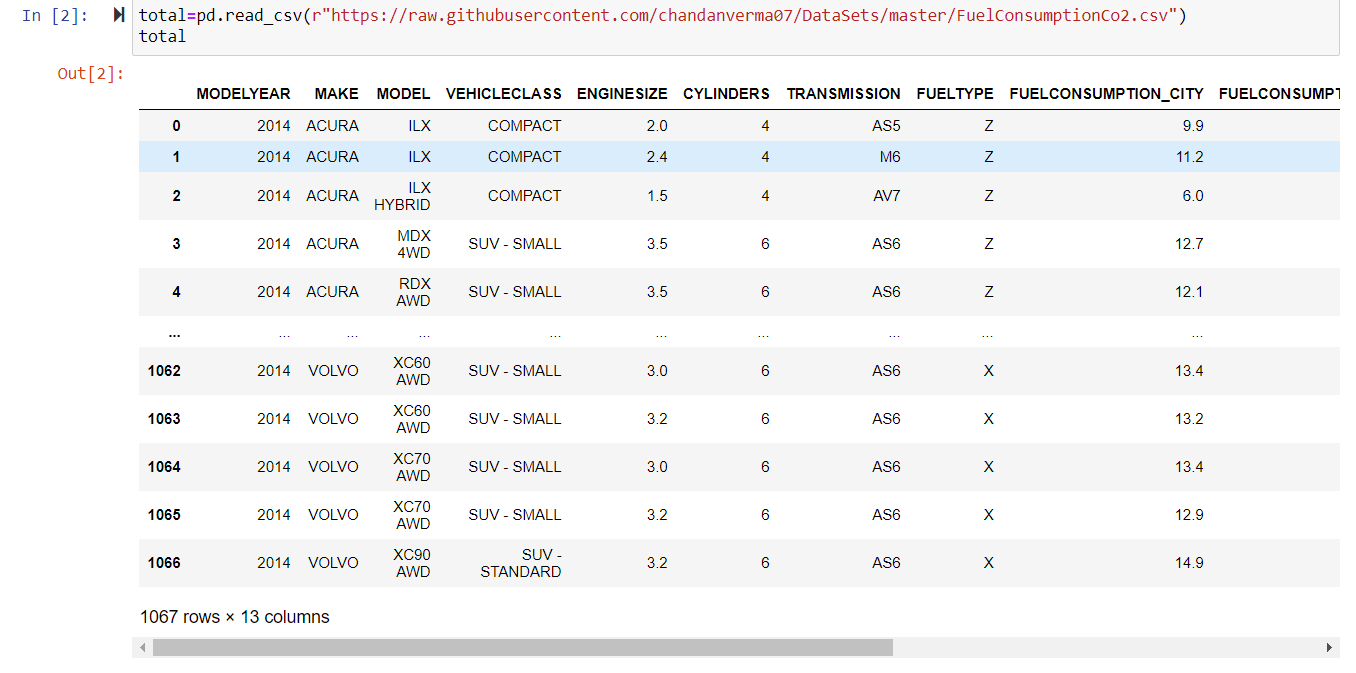
https://miro.medium.com/max/613/1*Utp8sgyLk7H39qOQY9pf1A.png

Linear regression is widely used in biological, behavioral and social sciences to describe possible relationships between variables. It ranks as one of the most important tools used in these disciplines.

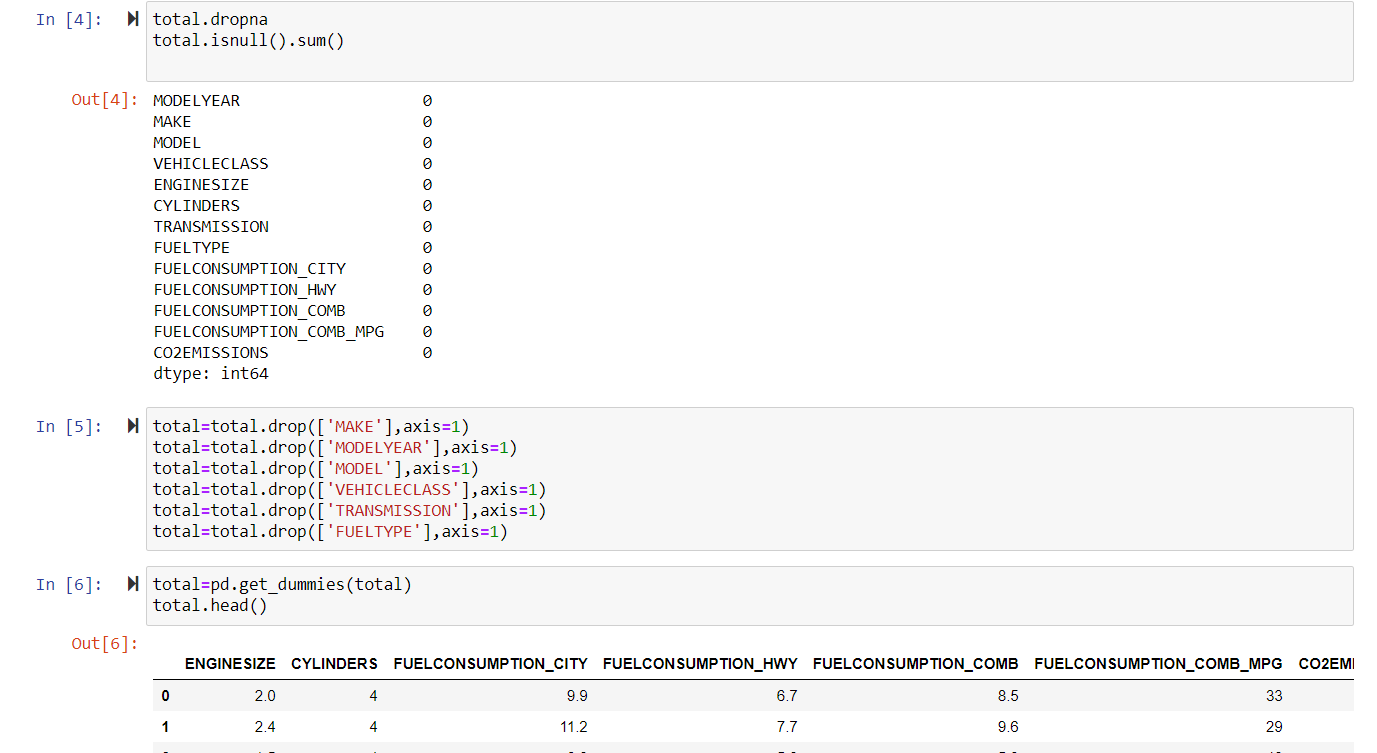
Dataset:-Fuel Consumption CO2

Jupyter file link:- https://drive.google.com/file/d/1iCieLKMaUVijuQDf-eo5aHwRJgcGY3ZA/view?usp=sharing

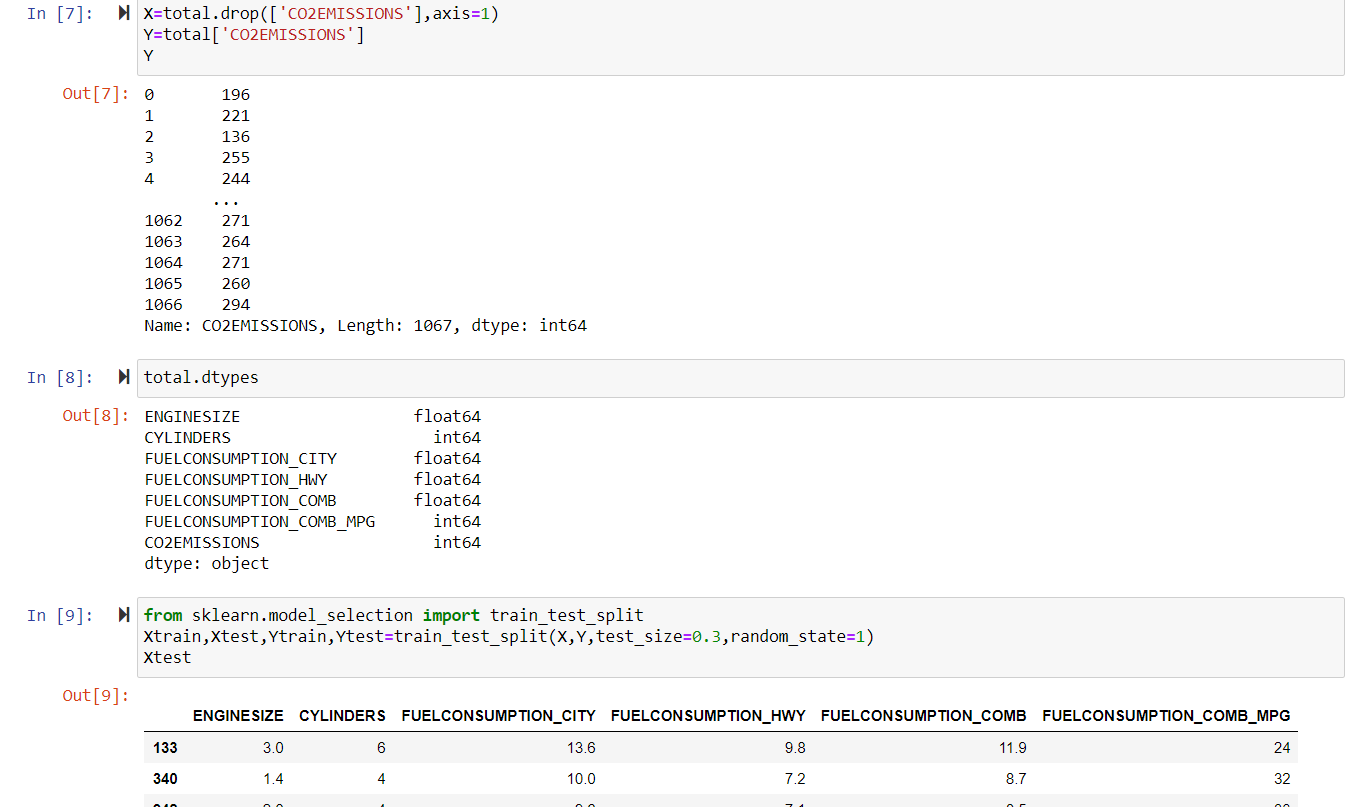
1.LOAD THE DATASET



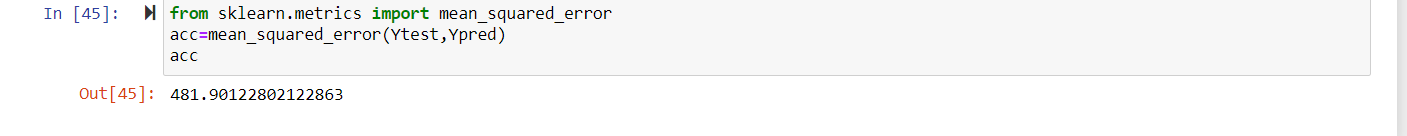
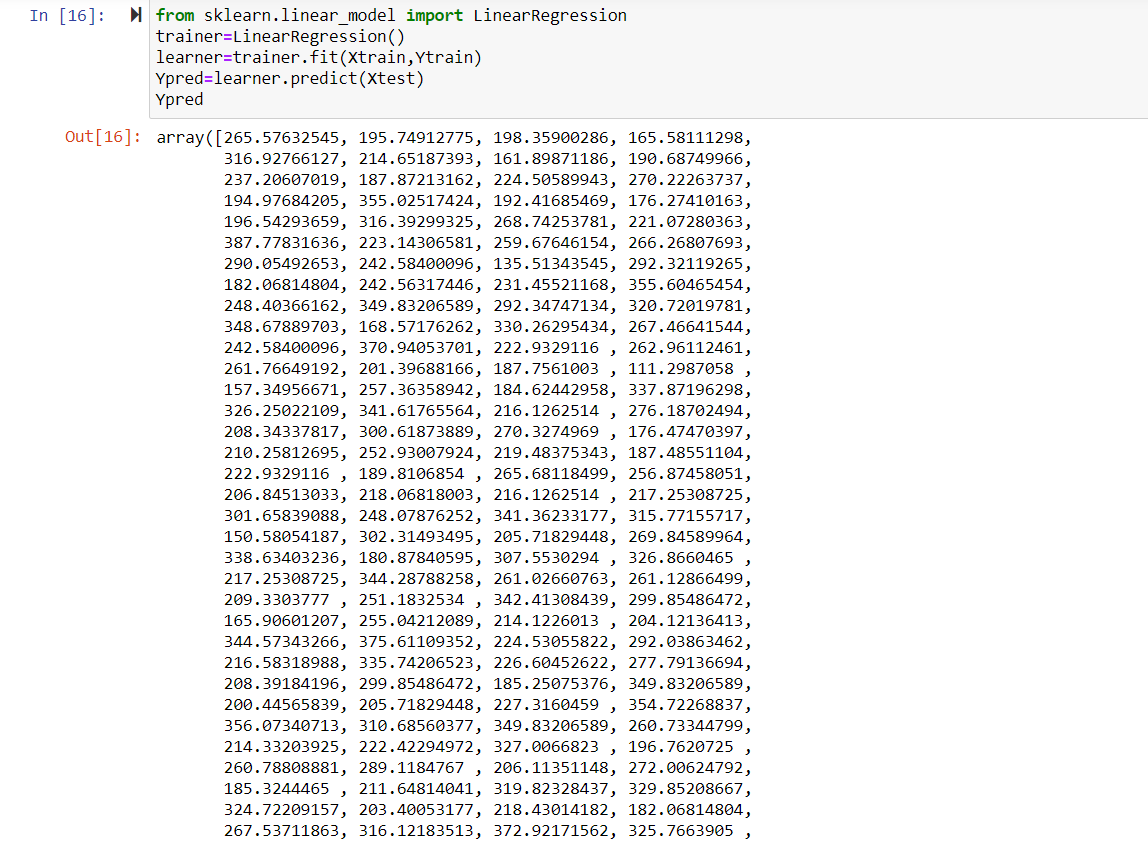
2.DATA PREPROCESSING



3.SPLITTING THE DATA



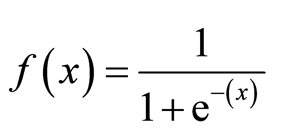
4.FITTING THE LINEAR REGRESSION ALGORITHM



**3.LOGISTIC REGRESSION**

 logistic regression IS a regression model. The model builds a regression model to predict the probability that a given data entry belongs to the category numbered as “1”. Just like Linear regression assumes that the data follows a linear function, Logistic regression models the data using the sigmoid function.

Logistic regression is used in various fields, including machine learning, most medical fields, and social sciences. For example, the Trauma and Injury Severity Score , which is widely used to predict mortality in injured patients, was originally developed by Boyd *et al.* using logistic regression.[[4]](https://en.wikipedia.org/wiki/Logistic_regression#cite_note-4) Many other medical scales used to assess severity of a patient have been developed using logistic regression.

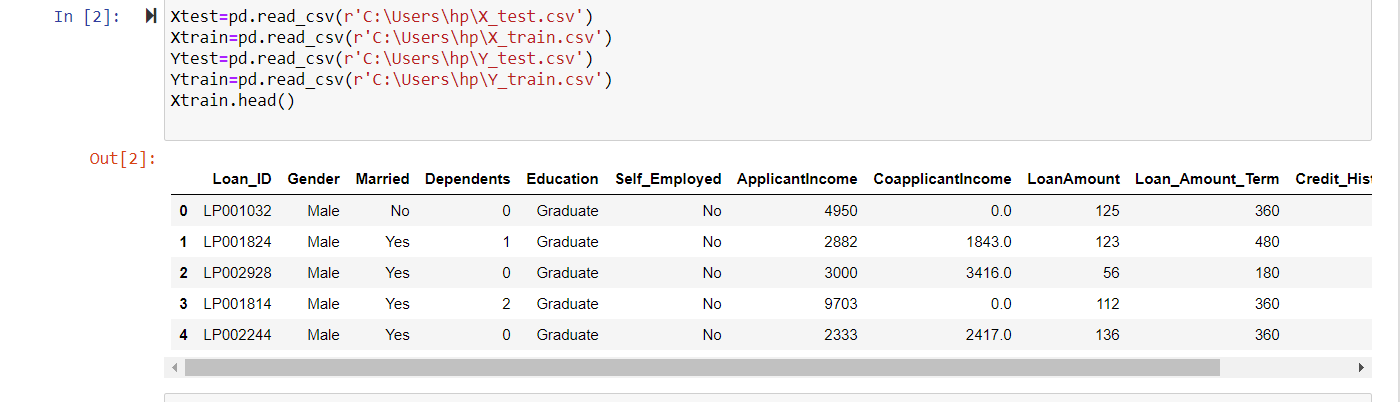


Dataset:-Titanic

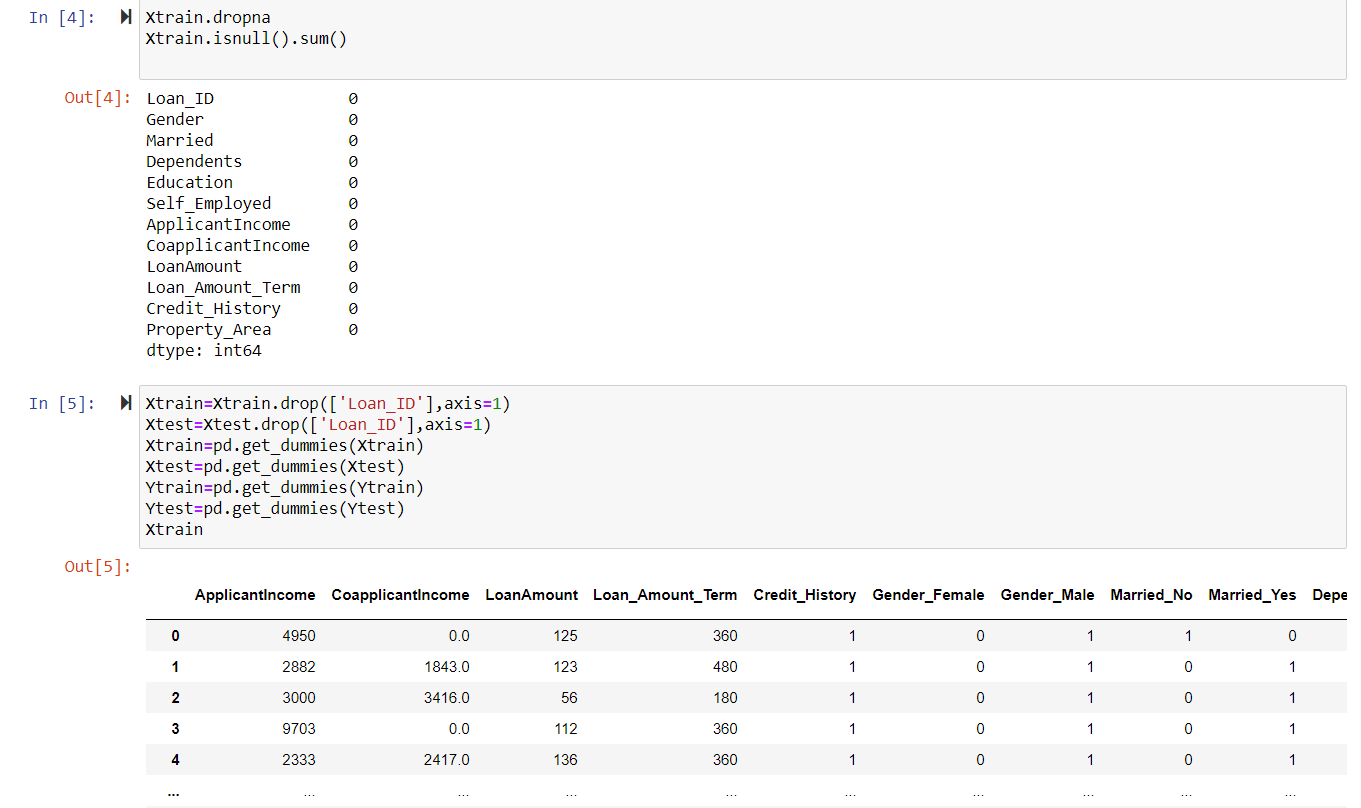
Jupyter file link:-

<https://drive.google.com/file/d/1rGPXs91akS-tEsAgskKTRfMZk3ysgw1V/view?usp=sharing>

1.LOAD THE DATASET



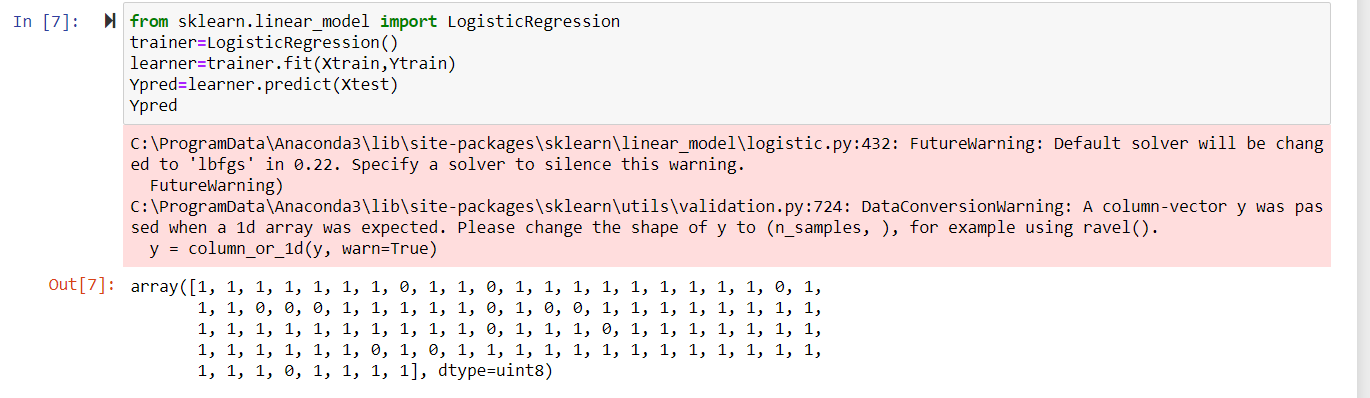
2.DATA PREPROCESSING



3.SPLITTING THE DATA



4.FITTING THE LOGISTIC REGRESSION



5.ACCURACY OF THE MODEL

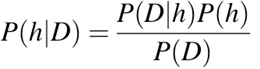


**4.NAIVEBAYES**

Naive Bayes is a statistical classification technique based on Bayes Theorem. It is one of the simplest supervised learning algorithms. Naive Bayes classifier is the fast, accurate and reliable algorithm. Naive Bayes classifiers have high accuracy and speed on large datasets.

Naive Bayes classifier is successfully used in various applications such as spam filtering, text classification, sentiment analysis, and recommender systems.

Naive Bayes classifier assumes that the effect of a particular feature in a class is independent of other features.

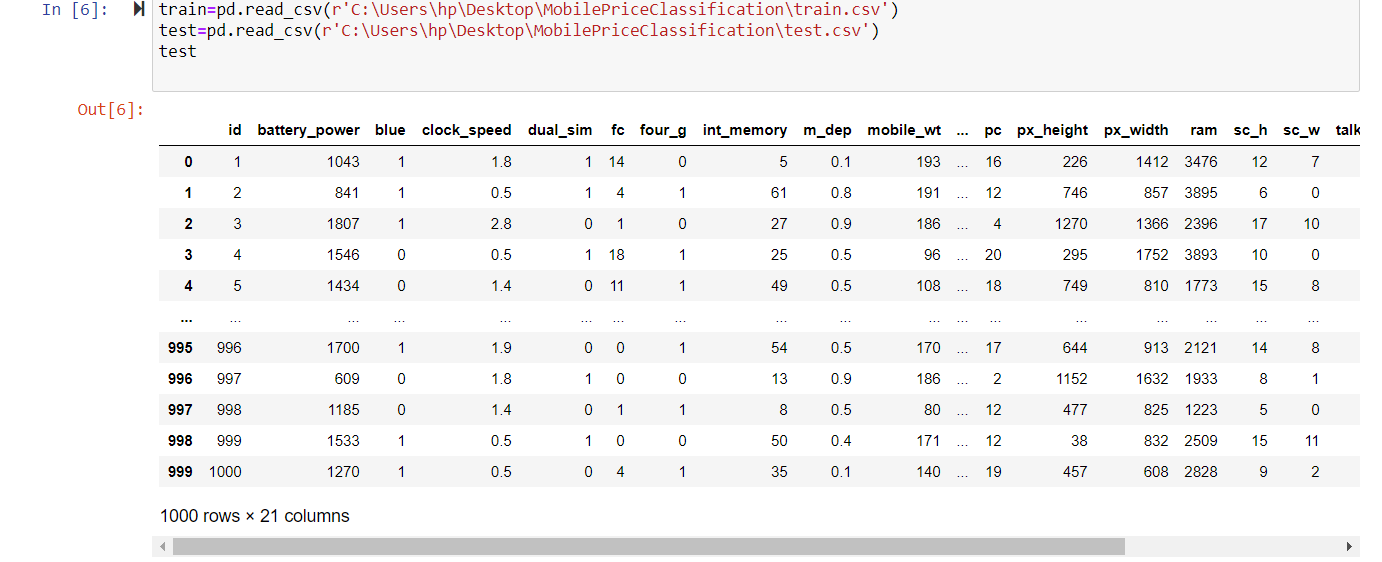


Dataset:-MobilePriceClassification

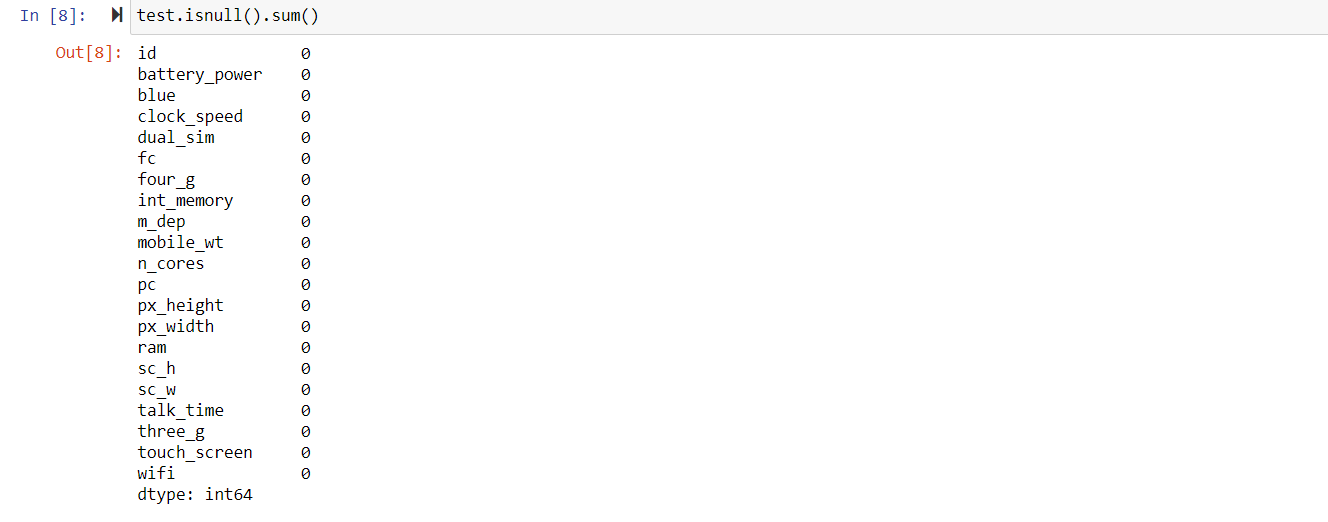
Jupyter file link:-

https://drive.google.com/file/d/1xCQtq9glvn6C1\_X3hpw\_XAlp3nZfuG6H/view?usp=sharing

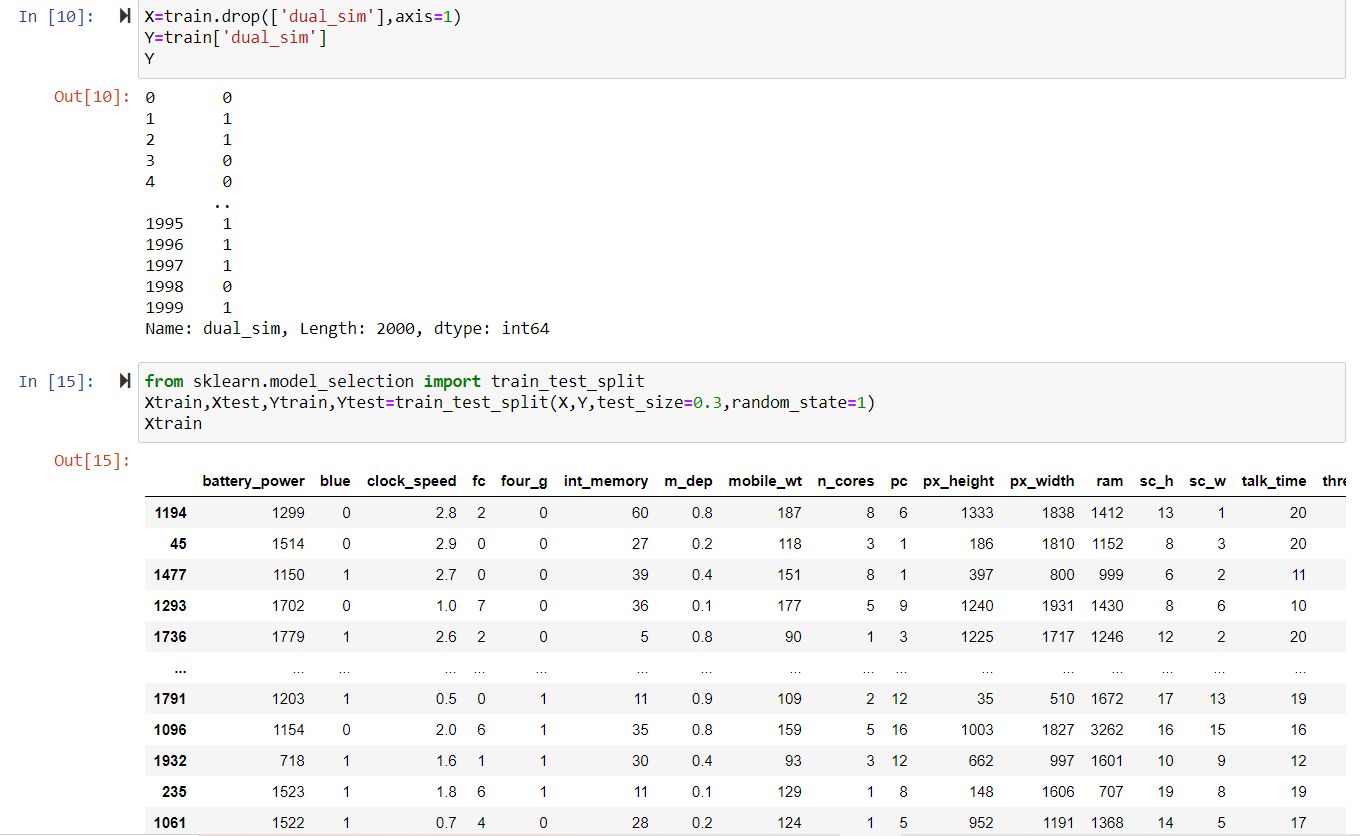
1.LOAD THE DATASET



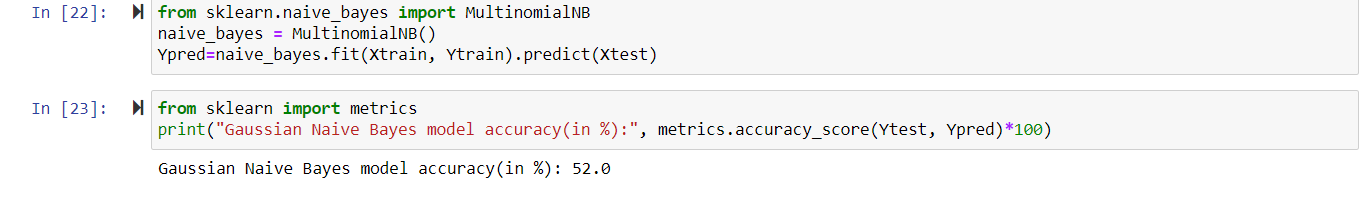
2.DATA PREPROCESSING



3.SPLITTING THE DATA



4.FITTING THE NAIVEBAYES ALGORITHM



**5.SVM ALGORITHM**

Support Vector Machine” (SVM) is a supervised machine learning algorithm which can be used for both classification or regression challenges. However,  it is mostly used in classification problems. In the SVM algorithm, we plot each data item as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiates the two classes very well (look at the below snapshot).

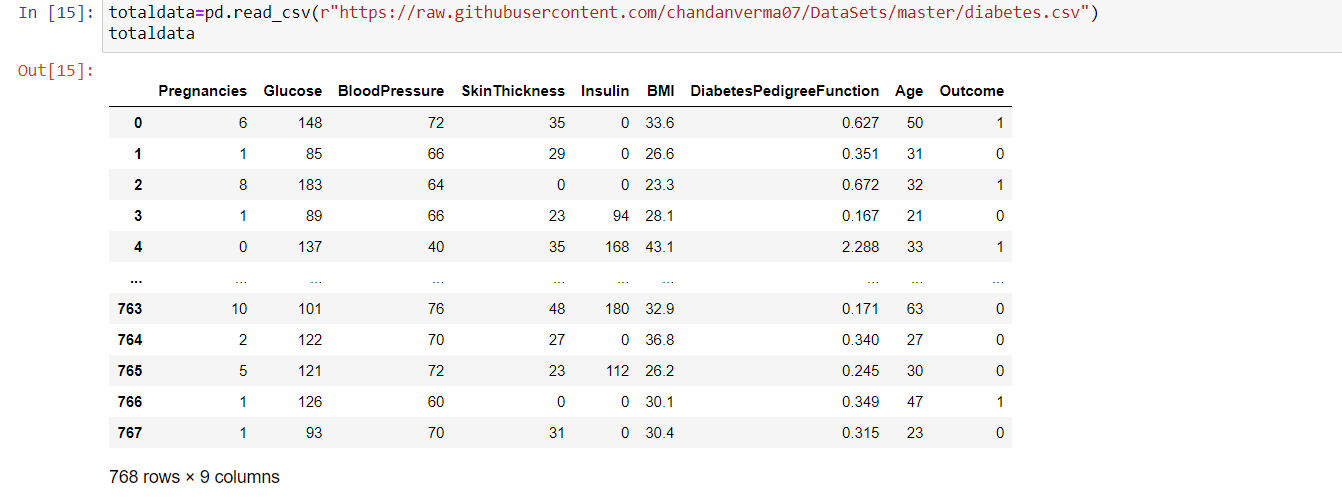
The objective of the support vector machine algorithm is to find a hyperplane in an N-dimensional space(N — the number of features) that distinctly classifies the data points.

In the SVM algorithm, we are looking to maximize the margin between the data points and the hyperplane. The loss function that helps maximize the margin is hinge loss.

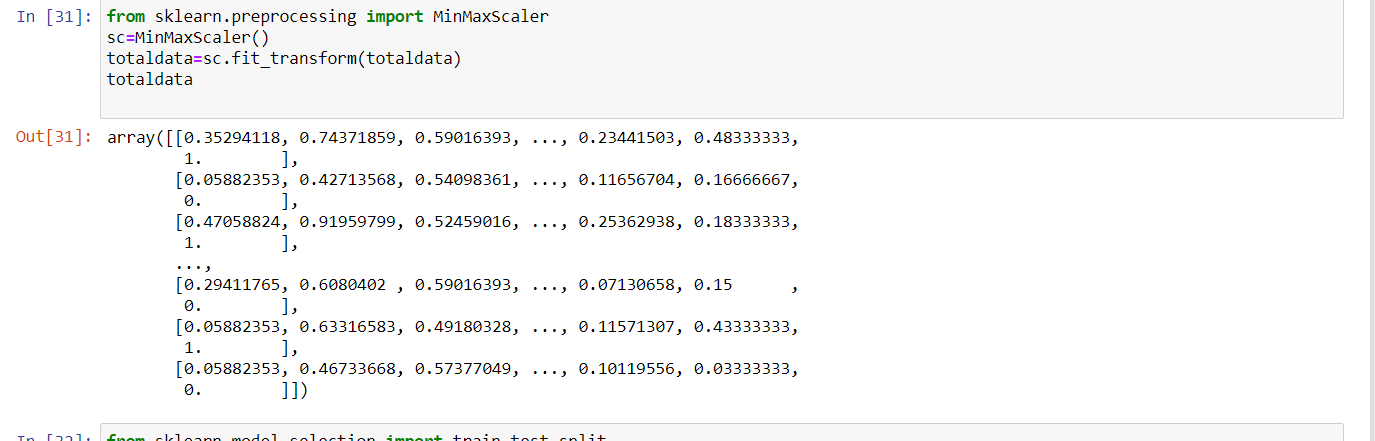
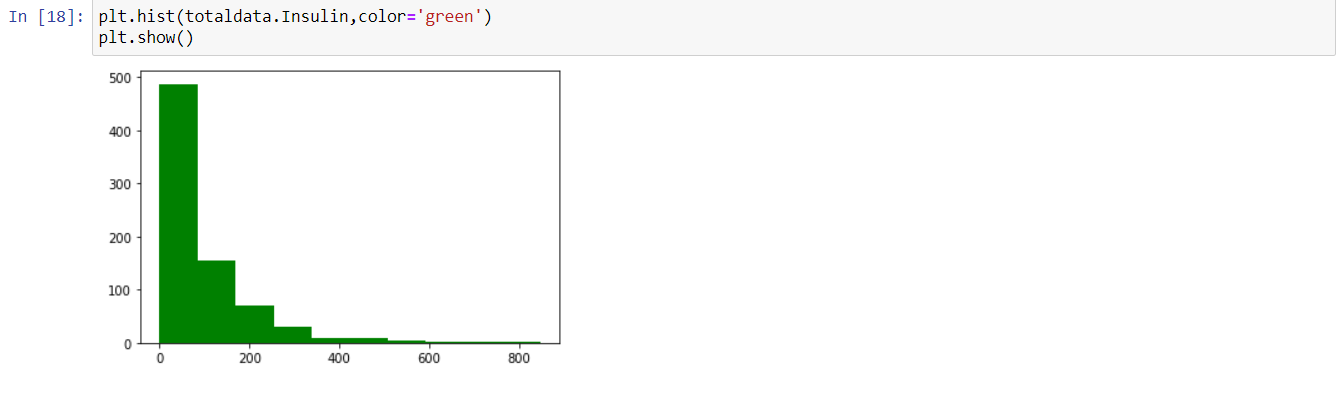
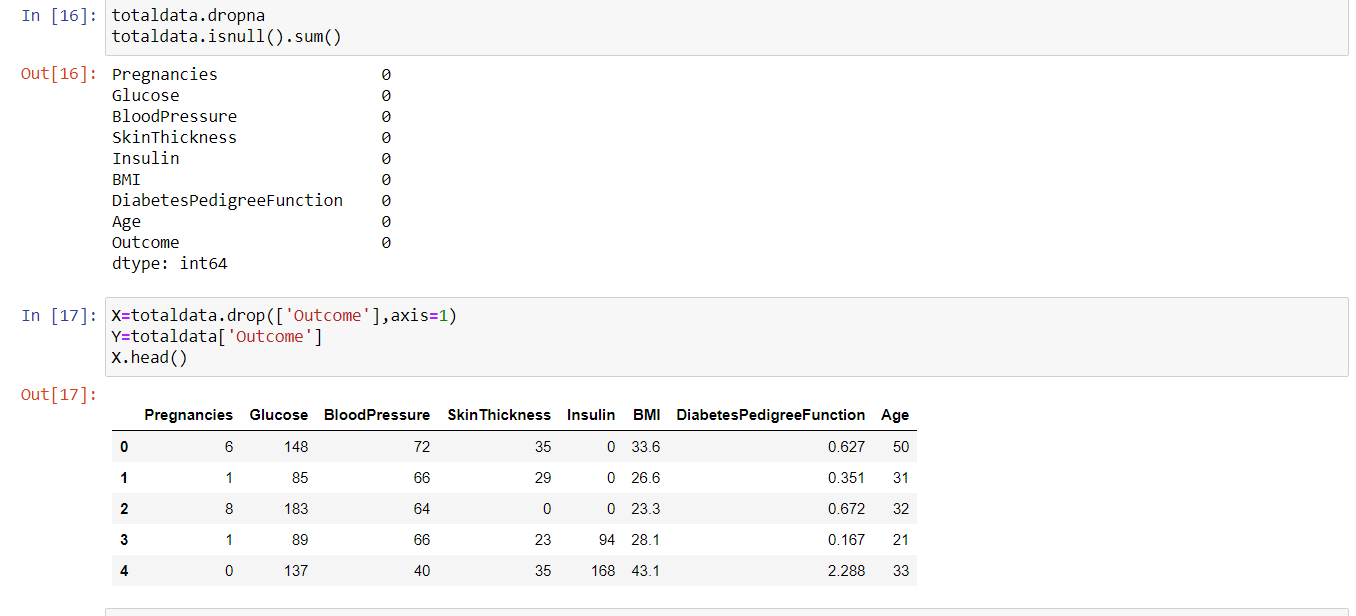
Dataset:-diabetes

Jupyter file link:- https://drive.google.com/file/d/1mypB66mOWUsj8nrIzBhft55byFKoZlPF/view?usp=sharing

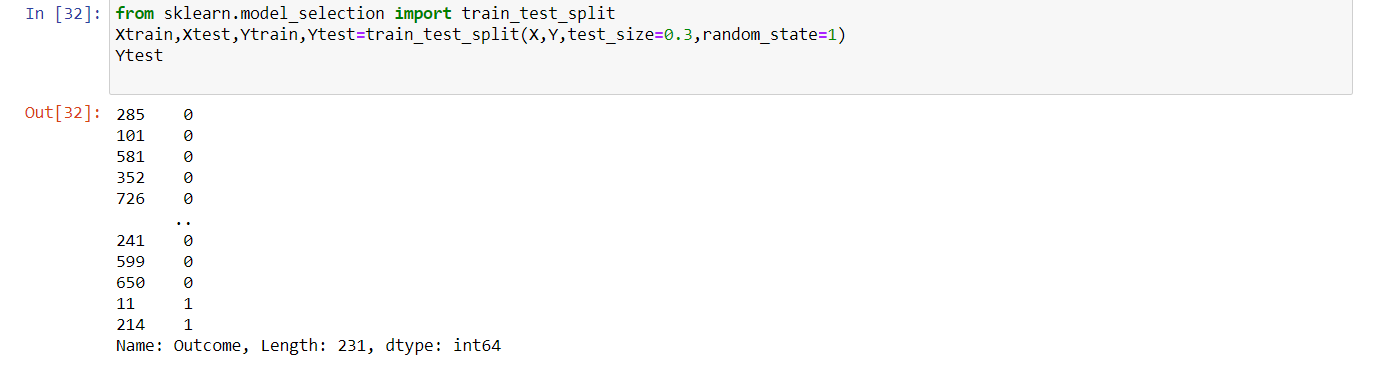
1.LOAD THE DATASET



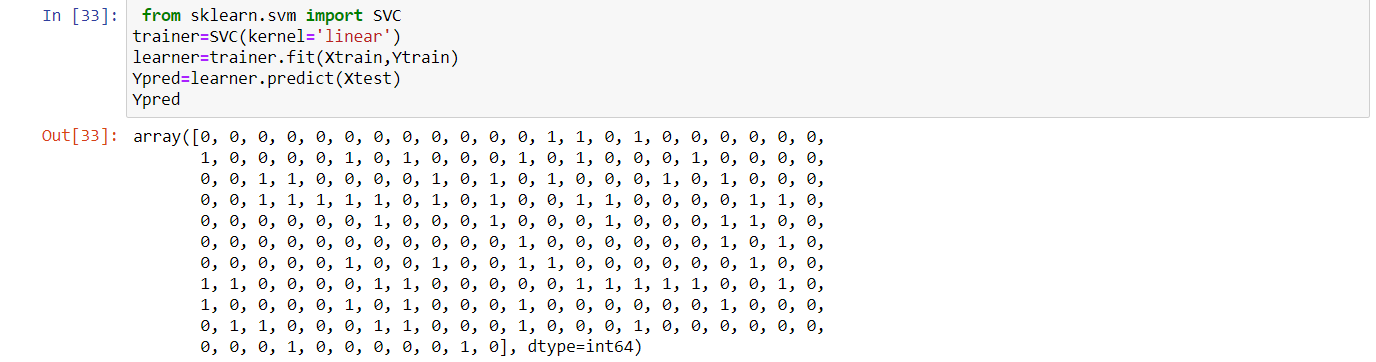
2.DATA PREPROCESSING



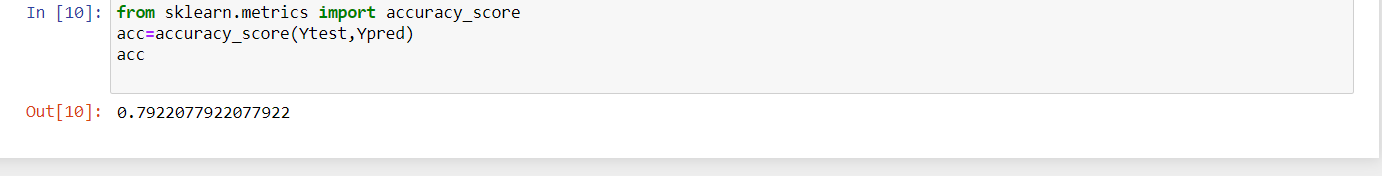
3.SPLITTING THE DATA



4.FITTING THE SVM ALGORITHM



5.ACCURACY OF THE MODEL



**6.KNN ALGORITHM**

The KNN algorithm assumes that similar things exist in close proximity. In other words, similar things are near to each other.

The k-nearest neighbors (KNN) algorithm is a simple, easy-to-implement supervised machine learning algorithm that can be used to solve both classification and regression problems.

**The KNN Algorithm**

1. Load the data
2. Initialize K to your chosen number of neighbors

3. For each example in the data

3.1 Calculate the distance between the query example and the current example from the data.

3.2 Add the distance and the index of the example to an ordered collection

4. Sort the ordered collection of distances and indices from smallest to largest (in ascending order) by the distances

5. Pick the first K entries from the sorted collection

6. Get the labels of the selected K entries

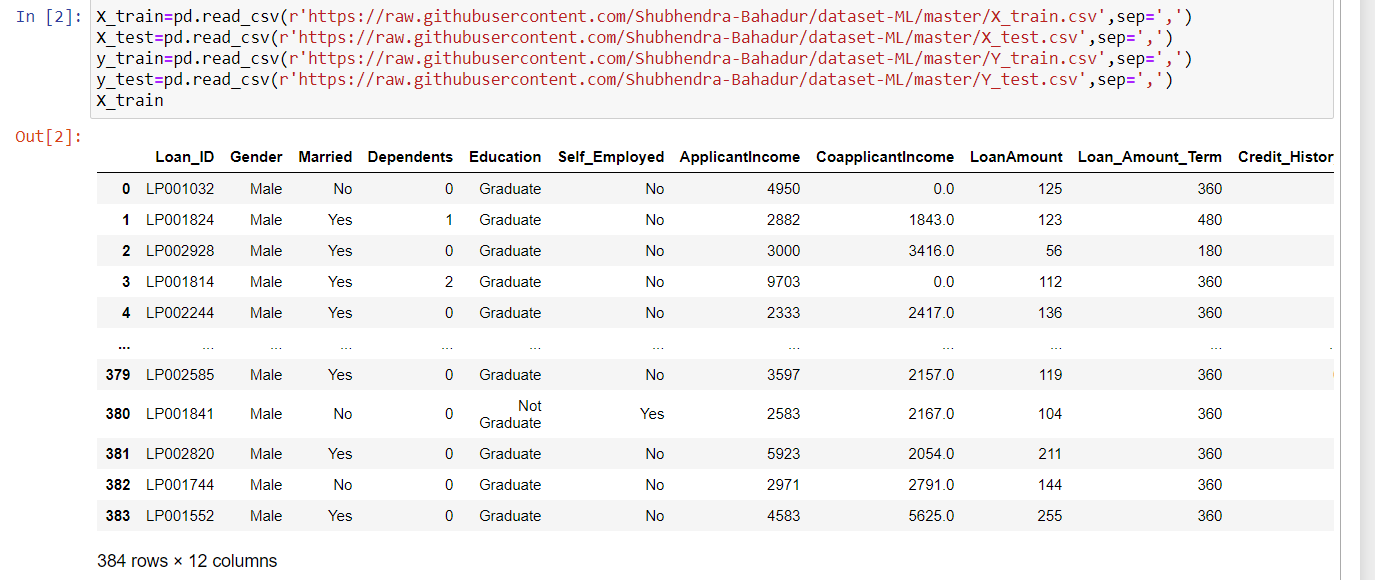
7. If regression, return the mean of the K labels

8. If classification, return the mode of the K labels

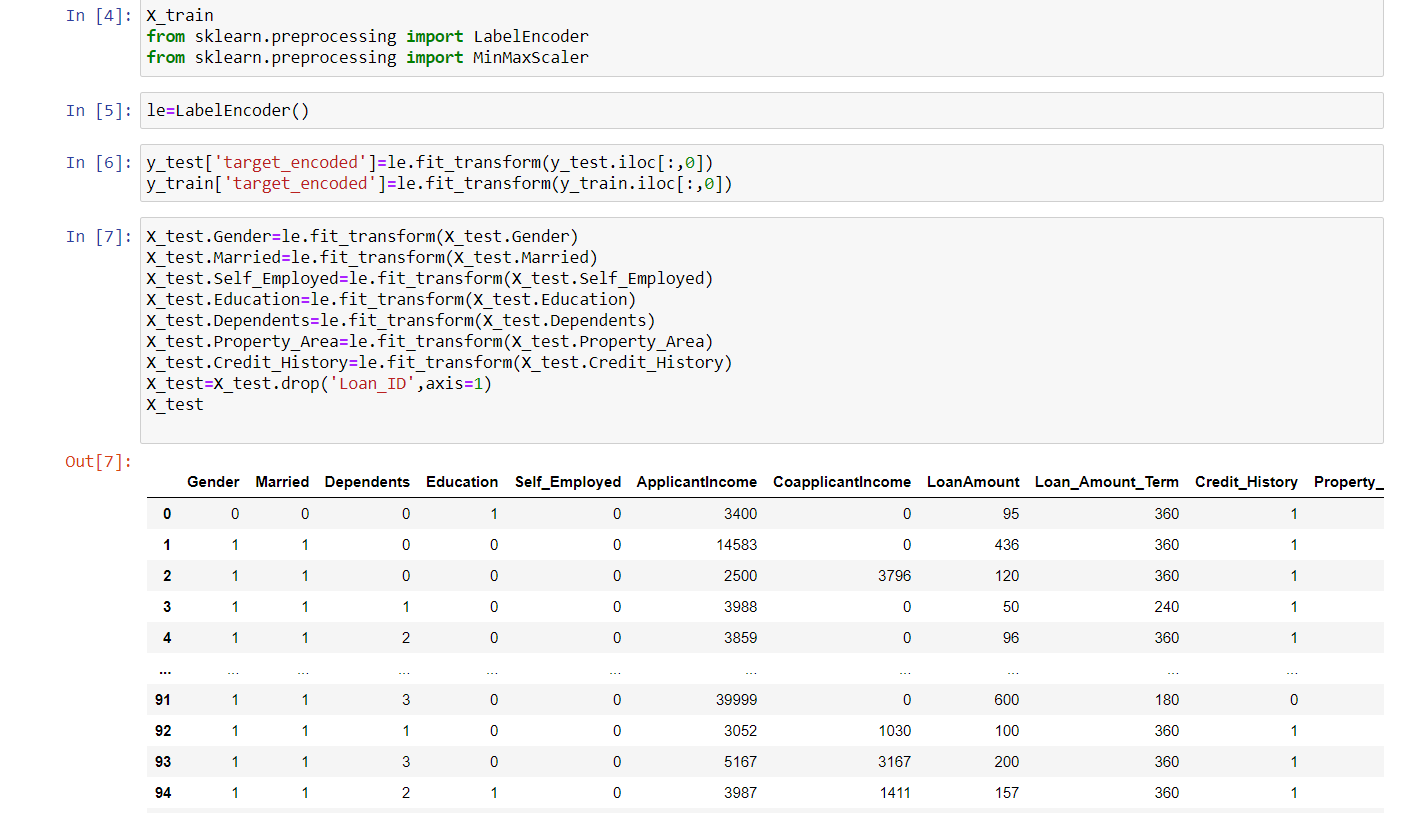
Dataset:-

Jupyter file link:- https://drive.google.com/file/d/1q9kLqEy3vbIuSbCgsQfxwqoONZYdqlnc/view?usp=sharing

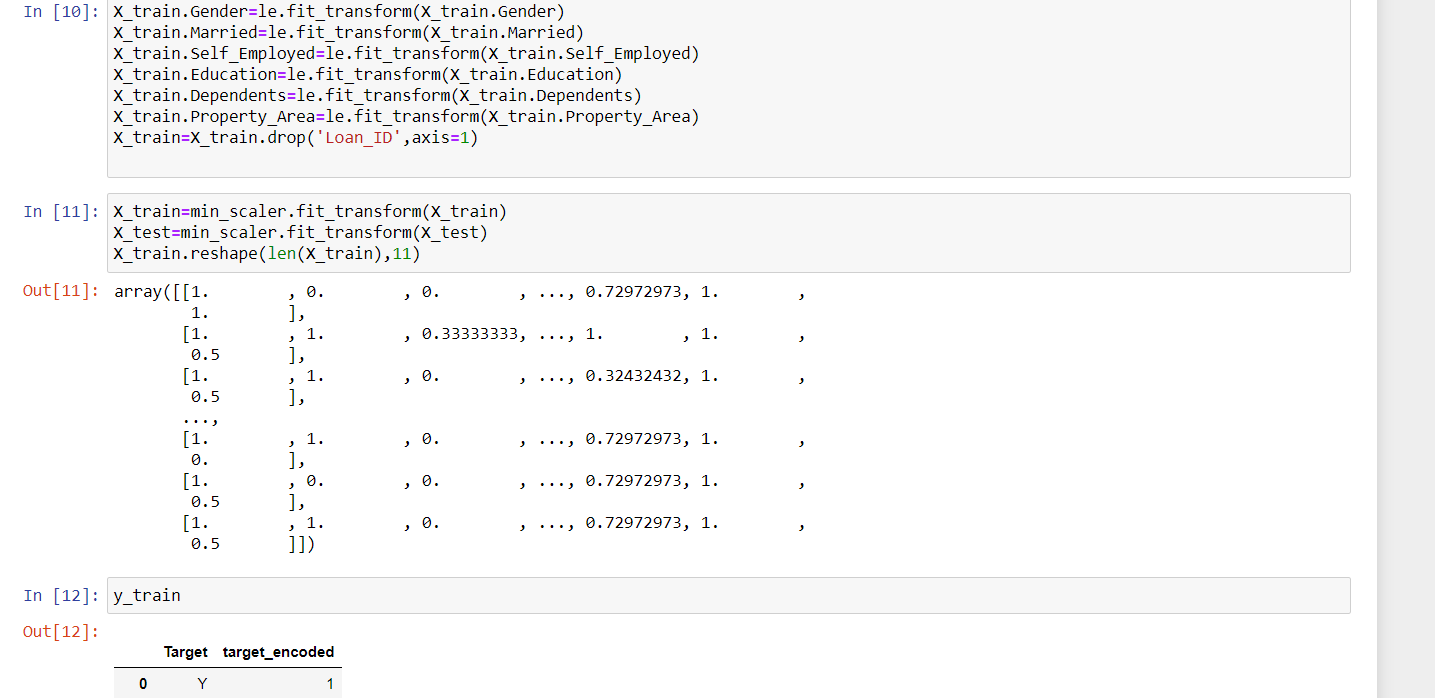
1.LOAD THE DATASET



2.DATA PREPROCESSING



3.SPLITTING THE DATA



4.FITTING THE KNN ALGORITHM



**References**

[1] "Intro to Machine Learning | Udacity." Intro to Machine Learning | Udacity. Accessed April 27, 2016. https://www.udacity.com/course/intro-to-machine-learning--ud120.

[2] "Elements of Statistical Learning: Data Mining, Inference, and Prediction. 2nd Edition. Datasets:Coronary Heart Disease Dataset." Elements of Statistical Learning: Data Mining, Inference, and Prediction. 2nd Edition. Accessed April 27, 2016. <http://statweb.stanford.edu/~tibs/ElemStatLearn/>.

[3] "No Free Lunch Theorems." No Free Lunch Theorems. Accessed April 27, 2016. <http://www.no-free-lunch.org/>.

[4] Hastie, Trevor, Robert Tibshirani, and J. H. Friedman. The Elements of Statistical Learning: Data Mining, Inference, and Prediction: With 200 Full-color Illustrations. New York: Springer, 2001.

[5] "Weka 3: Data Mining Software in Java." Weka 3. Accessed April 27, 2016. <http://www.cs.waikato.ac.nz/ml/weka/>.

[6] Bozhinova, Monika, Nikola Guid, and Damjan Strnad. Naivni Bayesov Klasifikator: Diplomsko Delo. Maribor: M. Bozhinova, 2015.

[7] Schölkopf, Bernhard, Christopher J. C. Burges, and Alexander J. Smola. Advances in Kernel Methods: Support Vector Learning. Cambridge, MA: MIT Press, 1999.

[8] Norving, Peter, and Stuart Russel. Artificial Intelligence: A Modern Approach. S.l.: Pearson Education Limited, 2013.

[9] Witten, I. H., and Eibe Frank. Data Mining: Practical Machine Learning Tools and Techniques. Amsterdam: Morgan Kaufman, 2005.