*A Progress Report*

*on*

# Graphical User Interface based Framework for Machine Learning

*carried out as part of the course CS1634 Submitted by*


***Aakash Lakhangaonkar***

***189301168***

***VI-CSE***


*in partial fulfilment for the award of the degree*

*of*

**BACHELOR OF TECHNOLOGY**

In

**Computer Science & Engineering**


MANIPAL UNIVERSITY
JAIPUR


**Department of Computer Science & Engineering,**
**School of Computing and IT,**
**Manipal University Jaipur,**

***April 2021***

**ABSTRACT**

The GUI Based Application Framework has a lot of application in the real world. The final aim of the project is that the user can input custom dataset and to visualize the machine learning algorithms and to predict the values or to classify, and to check accuracy of the models saving us from tedious coding work.

As this works with custom dataset it has multiple real time applications in the field of medical diagnostics, research, media, etc.

***Keywords*:** machine learning algorithms, visualize the algorithms, predict the values, classify, check accuracy, medical diagnostics

# LIST OF FIGURES



# LIST OF TABLES

# TABLE OF CONTENTS

# Chapter-1: INTRODUCTION

## 1.1) Scope of the Work:

This is a GUI based application framework for machine learning with predictions and visualizations. It can be widely used in teaching, research and industrial applications in the field of healthcare, education, media, manufacturing industries, etc. It would contain built in tools for standard machine learning tasks. It can helpful when you want output from dataset and avoid the pain of writing all the code.

In this project we will use streamlit framework of python to build the GUI and front-end. And also use python as language and sklearn package to implement the machine learning algorithms. We will also use pyplot to visualize the algorithms.

## 1.2) Product Scenarios:

Following are some common scenarios:

- This can be used in various medical diagnostics such as predicting whether the cells are benign or malignant(Cancer).
- It can also be used specific to current times to predict the possibility of person getting covid positive.
- It can be used by banks to see which customers are churn customers, or to approve loans, etc.
- It can be used by researchers to cluster similar objects and understand their properties.
- It can be also used to see which algorithms is best suited for the given dataset.

# Chapter-2: Requirement Analysis

## 2.1) Functional Requirements:

- The user should be able to select the desired dataset.
- The user should be able to select which model to apply.
- The app should pre-process the data and visualize the data.
- It should also display the accuracy of the model.

## 2.2) Non-Functional Requirements:

- The site should be loaded in 10 sec.
- User should be able to upload dataset up to 100mb.
- Each request should be processed within 10sec.

## 2.3) Use Case:

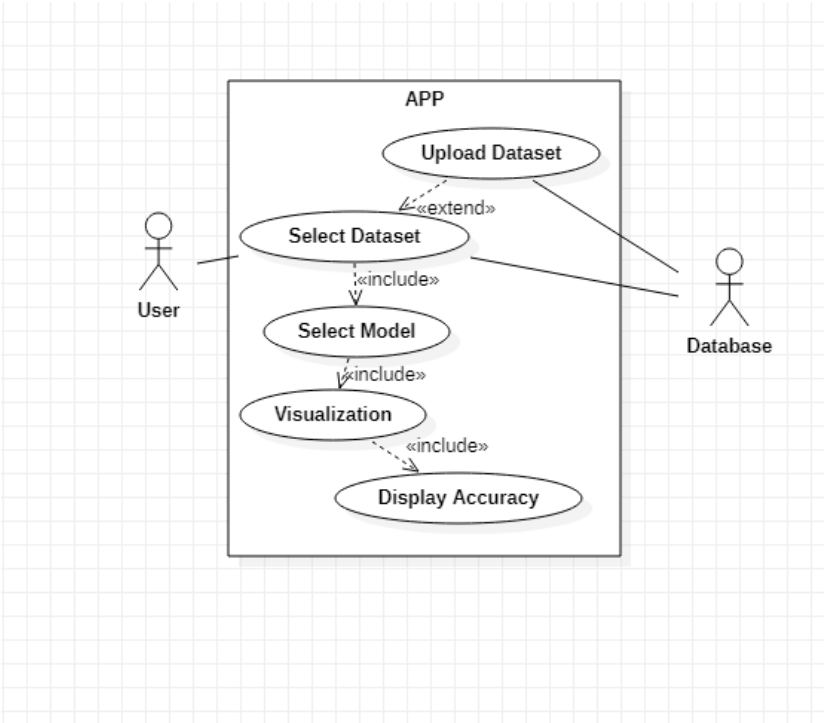



Figure-1 Use Case

## 2.4) Pre-Existing Tools:

| Existing Tools | Advantages | Limitations |
|---|---|---|
| Tkinter | It is good when the data is cleaned and prepared well.<br><br>Very simple syntax.<br>The canvas widget is also very easy and powerful. Tk is rock solid with few cross-platform idiosyncrasies. | It can only handle small datasets. Whenever a set is bigger than a few megabytes, an OutOfMemory error occurs.<br>Sometimes hard to debug in that Tkinter widgets at their core aren't python objects; tkinter provides a wrapper around the actual tk widgets which sometimes means you get weird error messages |
| WxPython | Data Preparation: Seamlessly integrated & optimized for building ML models.<br><br>Machine Learning: Design models using a visual workflow designer or automated modelling. | wxPython requires a separate download which can be a pain to manage when you deploy your app. wxPython is a bit buggy IMO, and there are definite cross-platform issues. |

Table-1

# Chapter-3: System Design

## 3.1) Design Goals:

Proposed work is to visualize all the machine learning algorithms. It can be divided into three main procedures as given below:

i. Select the dataset to use.
ii. Select the model that you want to apply.
iii. The app applies pre-processing on the dataset and divides it into training and testing data.
iv. Then the model is applied.
v. The visualization is shown and the accuracy of the model is calculated.

## 3.2) System Architecture:

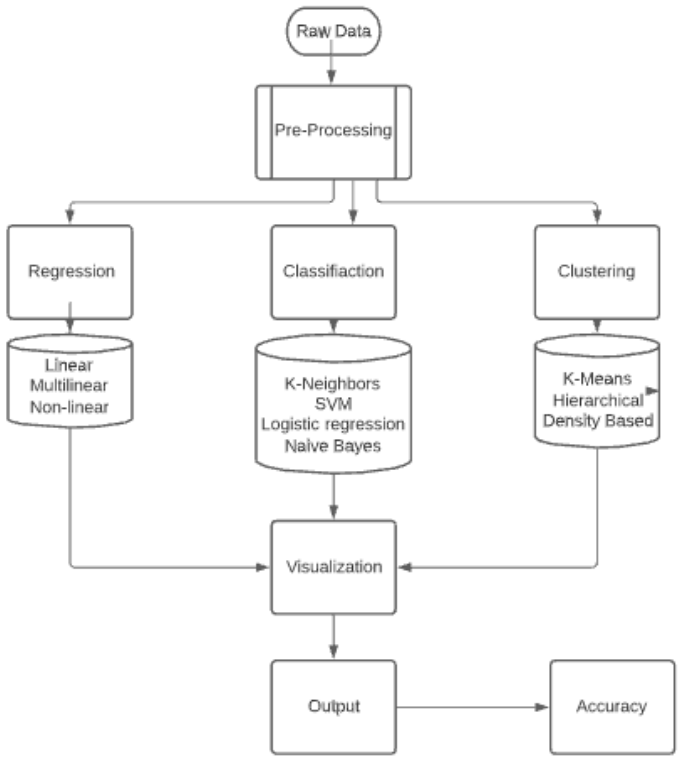



Figure-2 Block Diagram Of The Model

## 3.3) Detailed Design Methodologies:

**GUI:**

The GUI is designed with the help of streamlit framework of python. It helps in creating all the widgets in a simplified form and make the interface for the user simple to understand and easy to use.

**Pre-Processing:**

This Is very important before applying the model, as dataset have different types of values which cannot be used directly and requires some pre-processing.

**Applying Model:**

The required model is applied with the help of sklearn package in python.

**Visualization:**

This is achieved with matplotlib and pyplot from python to create graphs and represent pictographically all the information. Also seaborn was used for heatmap and confusion matrix.

**Accuracy:**

The accuracy is calculated by comparing the values the predicted values and testing values and displayed.

# Chapter-4: System Design

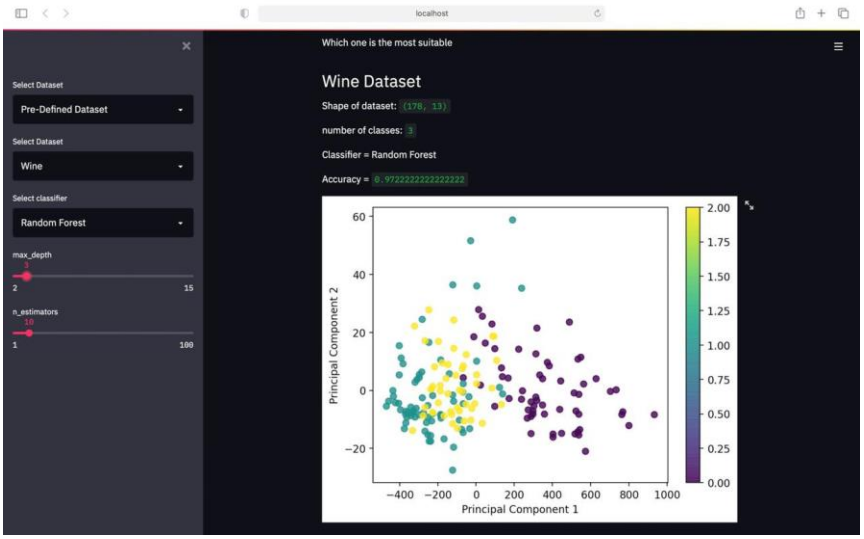
## 4.1) Development Environment: Pycharm

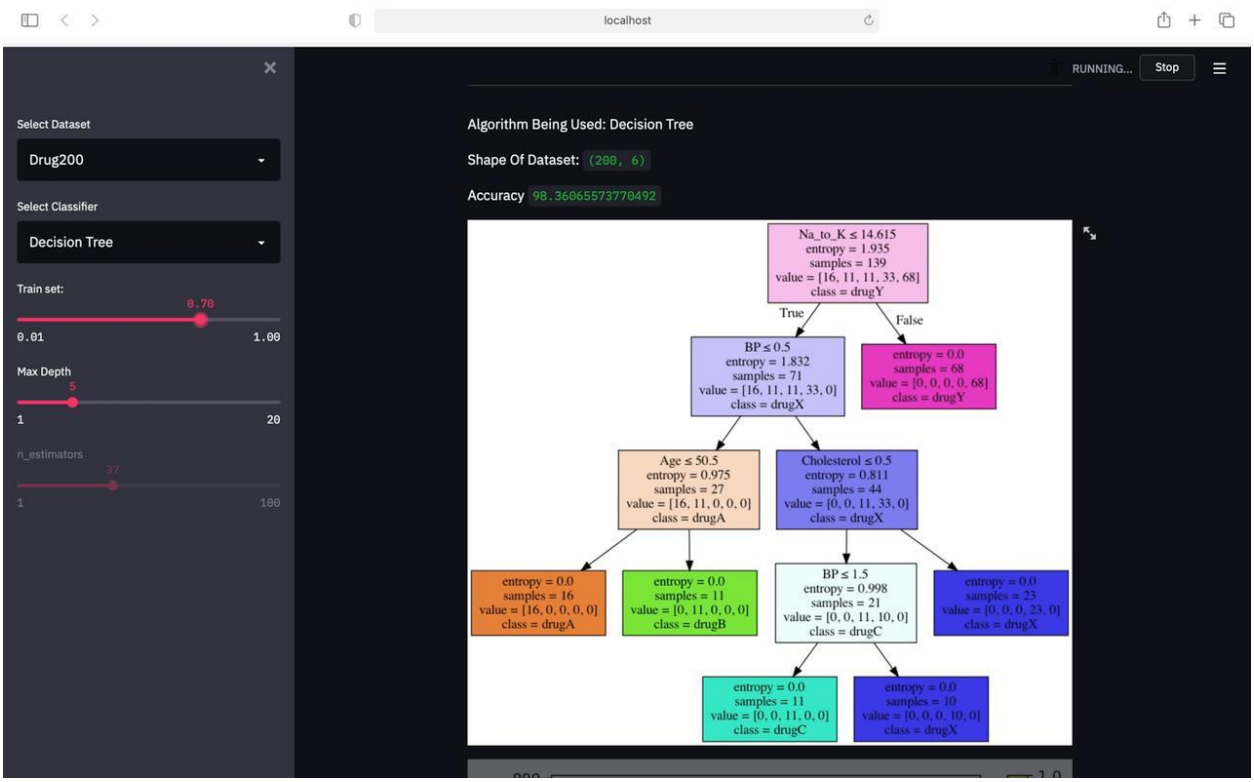
- Pycharm is an open source software available that mainly used to write python based codes and also some other plugins.
- It is easily available for operating systems like MAC, Windows, Linux and comes with various python packages and other frameworks.
- A range of python packages are available such as matplot.lib, pandas, numpy, etc.
- The main code file will generate a .py file.
- The IDE majorly consist 2 parts: Editor and Compiler where former is used for writing code and later is used for compiling code and running it on local host until deployed.
- The environment supports python, html, css.

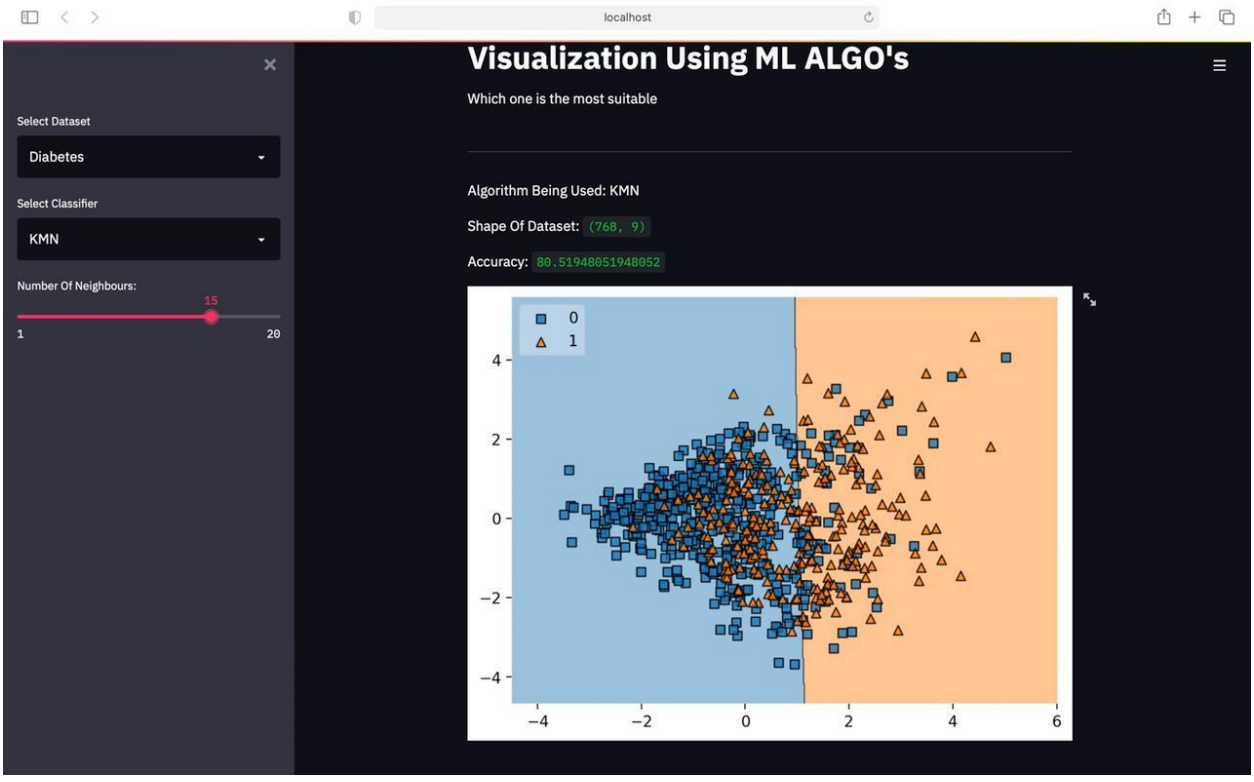
## 4.2) Work Done:

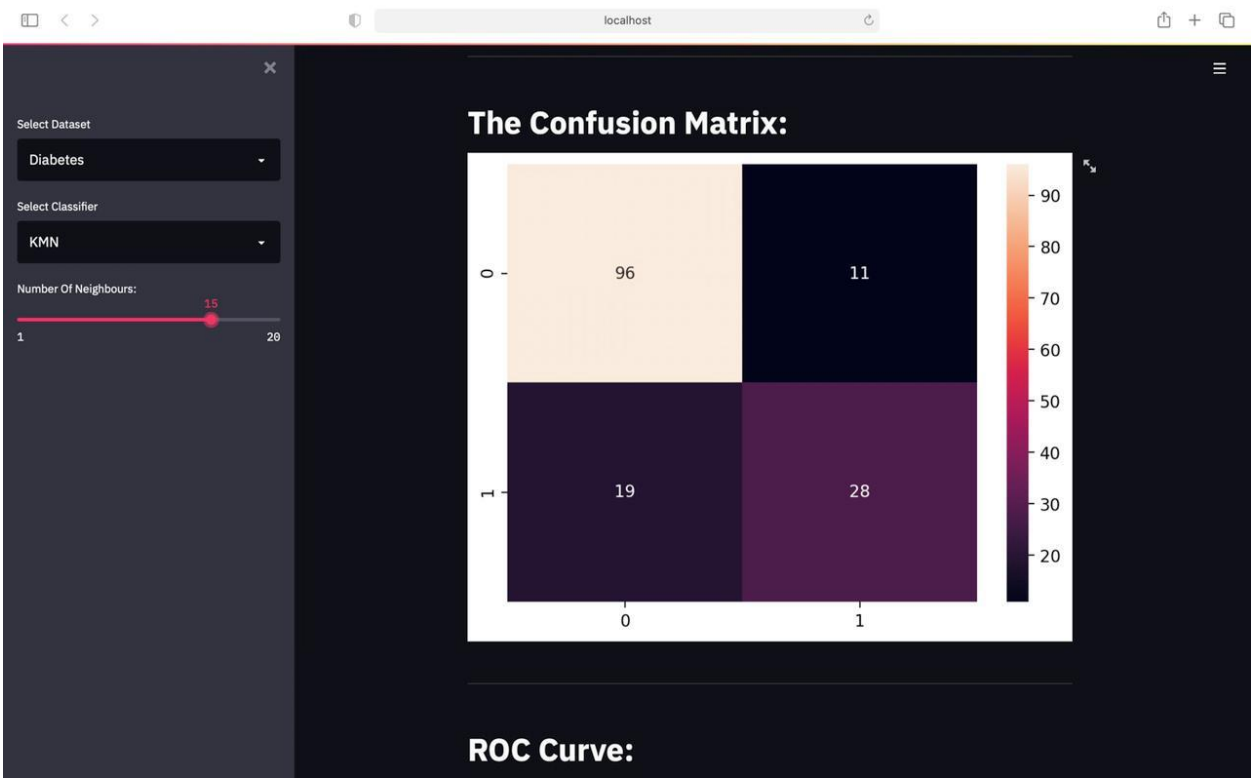
The aim was to create the GUI based framework for machine learning algorithms.

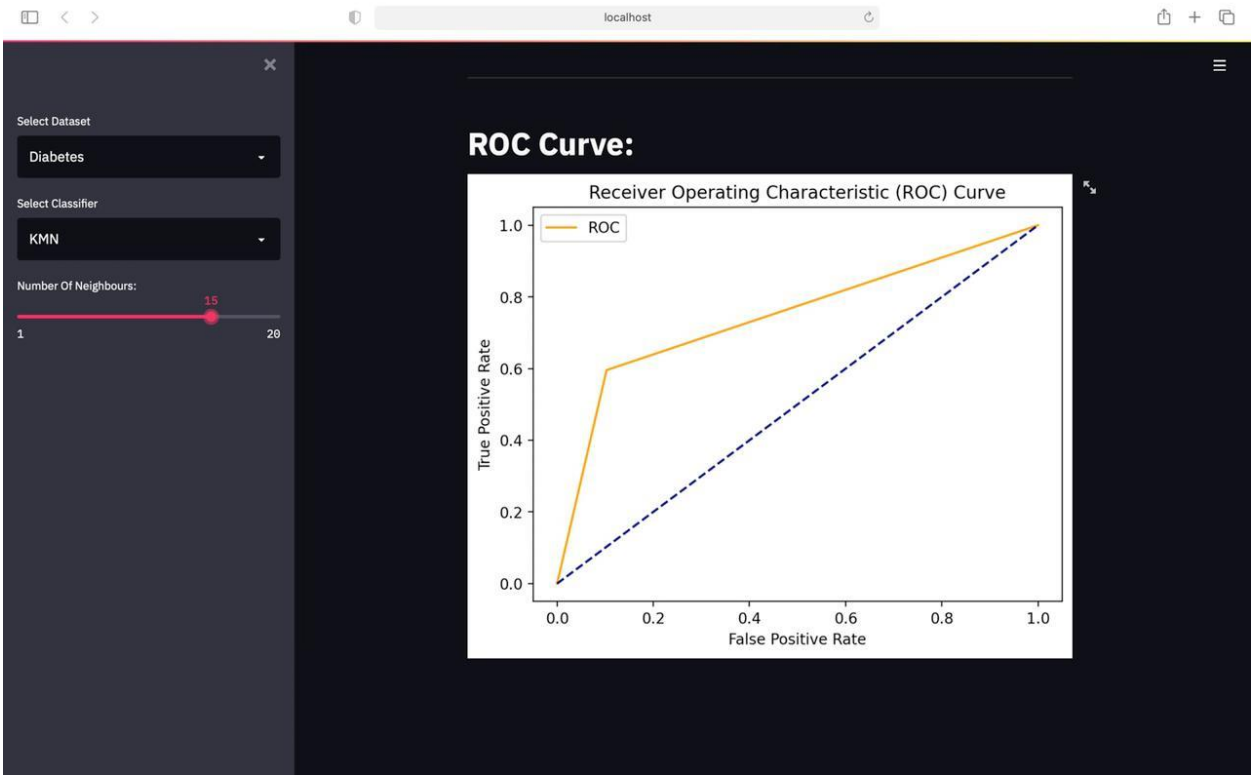
- I studied all the models that we are going to use and understood their functioning.
- Collected multiple datasets and tested each model to understand their behavior and test the accuracy of the models.
- Constructed a small level GUI and implemented the basic functionality.



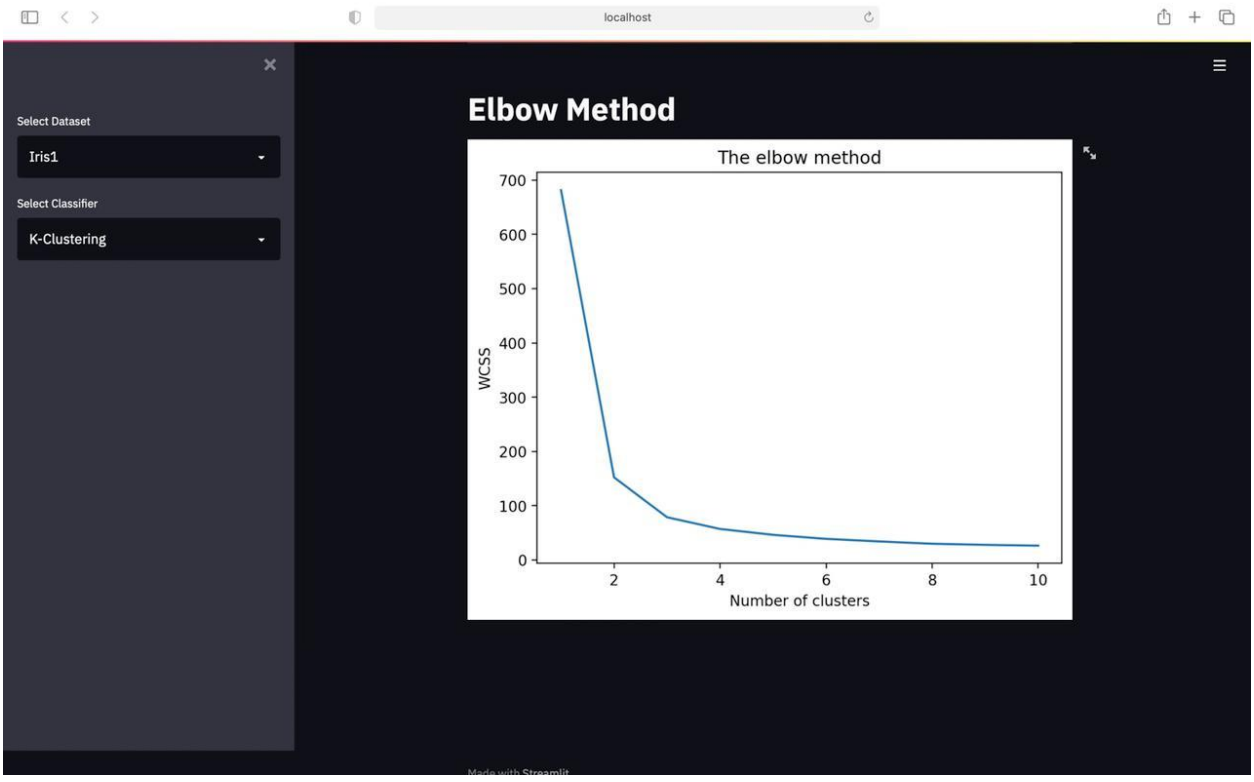

localhost
RUNNING...
Stop
Select Dataset
Drug200
Select Classifier
Decision Tree
Train set:
0.70
0.01
1.00
Max Depth
5
1
20
n_estimators
37
1
100
Algorithm Being Used: Decision Tree
Shape Of Dataset: (200, 6)
Accuracy 98.36065573770492
Na_to_K ≤ 14.615
entropy = 1.935
samples = 139
value = [16, 11, 11, 33, 68]
class = drugY
True
False
BP ≤ 0.5
entropy = 1.832
samples = 71
value = [16, 11, 11, 33, 0]
class = drugX
entropy = 0.0
samples = 68
value = [0, 0, 0, 0, 68]
class = drugY
Age ≤ 50.5
entropy = 0.975
samples = 27
value = [16, 11, 0, 0, 0]
class = drugA
Cholesterol ≤ 0.5
entropy = 0.811
samples = 44
value = [0, 0, 11, 33, 0]
class = drugX
entropy = 0.0
samples = 16
value = [16, 0, 0, 0, 0]
class = drugA
entropy = 0.0
samples = 11
value = [0, 11, 0, 0, 0]
class = drugB
BP ≤ 1.5
entropy = 0.998
samples = 21
value = [0, 0, 11, 10, 0]
class = drugC
entropy = 0.0
samples = 23
value = [0, 0, 0, 23, 0]
class = drugX
entropy = 0.0
samples = 11
value = [0, 0, 11, 0, 0]
class = drugC
entropy = 0.0
samples = 10
value = [0, 0, 0, 10, 0]
class = drugX

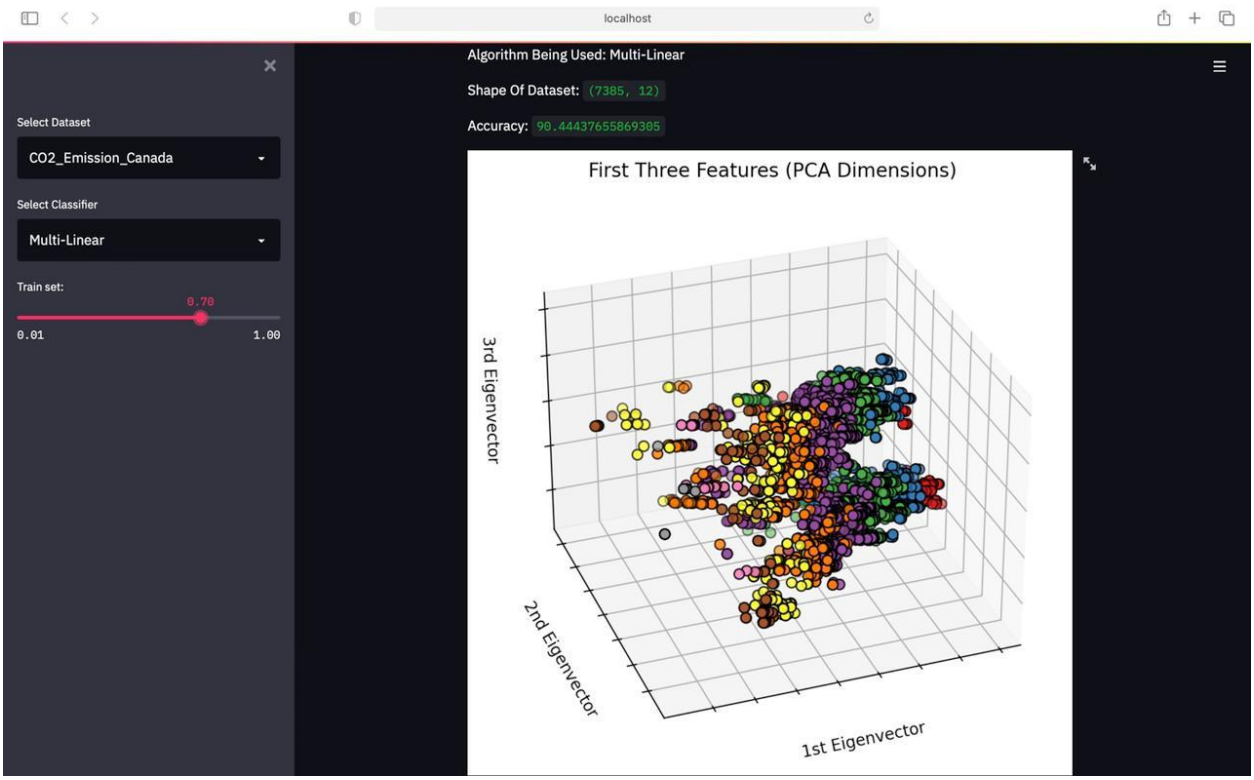

localhost
Visualization Using ML ALGO's
Which one is the most suitable
Select Dataset
Diabetes
Select Classifier
KMN
Number Of Neighbours:
15
1
20
Algorithm Being Used: KMN
Shape Of Dataset: (768, 9)
Accuracy: 80.51948051948052
0
1
4
2
0
−2
−4
−4
−2
0
2
4
6

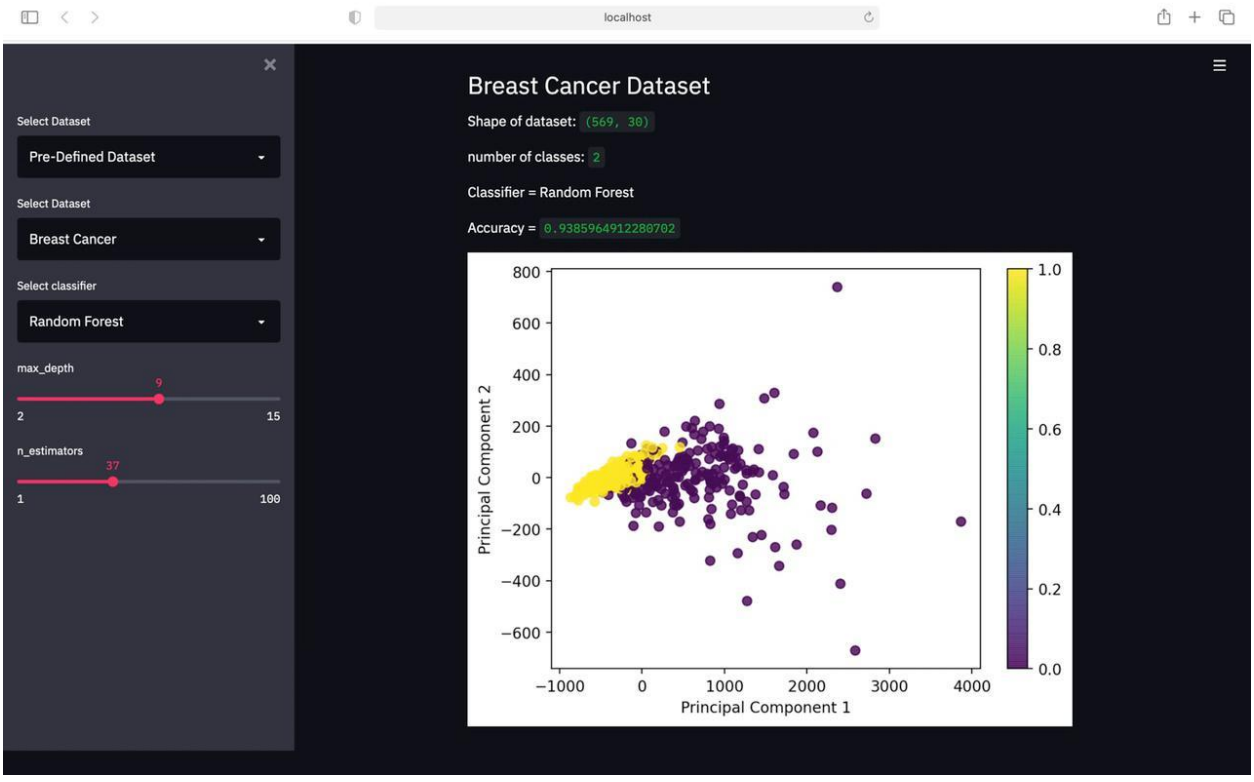

localhost
Select Dataset
Diabetes
Select Classifier
KMN
Number Of Neighbours:
15
1
20
The Confusion Matrix:
96
11
19
28
0
1
0
1
90
80
70
60
50
40
30
20
ROC Curve:

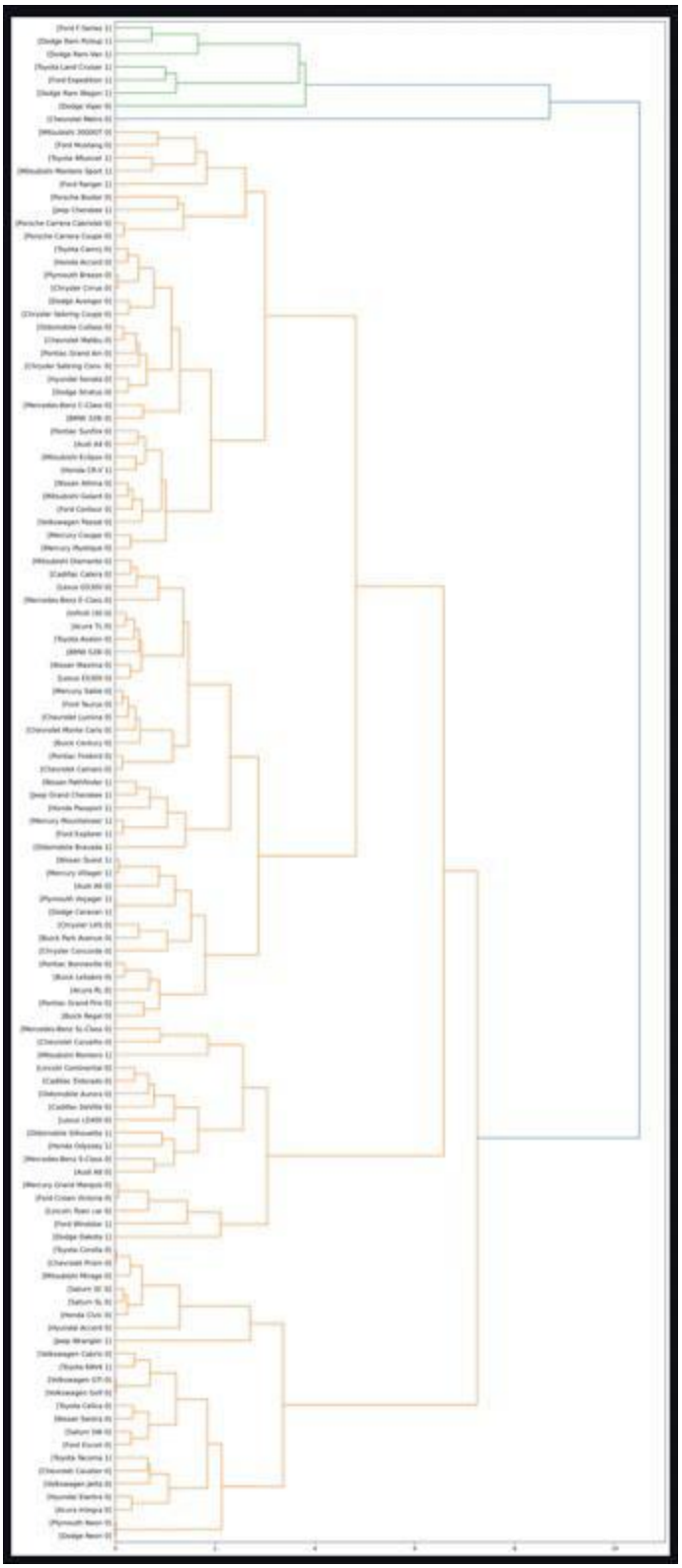

localhost
Select Dataset
Diabetes
Select Classifier
KMN
Number Of Neighbours:
15
1
20
ROC Curve:
Receiver Operating Characteristic (ROC) Curve
ROC
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0


localhost
Select Dataset
Iris1
Select Classifier
K-Clustering
Visualization Using ML ALGO's
Which one is the most suitable
Iris-setosa
Iris-versicolour
Iris-virginica
4.5
4.0
3.5
3.0
2.5
2.0
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0


localhost
Select Dataset
Iris1
Select Classifier
K-Clustering
Elbow Method
The elbow method
WCSS
Number of clusters
700
600
500
400
300
200
100
0
2
4
6
8
10
Made with Streamlit


localhost
Select Dataset
CO2_Emission_Canada
Select Classifier
Multi-Linear
Train set:
0.70
0.01
1.00
Algorithm Being Used: Multi-Linear
Shape Of Dataset: (7385, 12)
Accuracy: 90.44437655869305
First Three Features (PCA Dimensions)
3rd Eigenvector
2nd Eigenvector
1st Eigenvector


localhost
Select Dataset
Pre-Defined Dataset
Select Dataset
Breast Cancer
Select classifier
Random Forest
max_depth
9
2
15
n_estimators
37
1
100
Breast Cancer Dataset
Shape of dataset: (569, 30)
number of classes: 2
Classifier = Random Forest
Accuracy = 0.9385964912280702
Principal Component 2
Principal Component 1



These are few of the Screenshots of the GUI to give you an idea of the working.
In this:

- First you select a dataset on which the different types of classifiers available for that dataset are displayed.
- You then select a desired classifier upon which the GUI gives you a few parameters for the classifiers.
- After selecting those all the visualizations available for those classifiers are displayed, along with accuracy and a few other essential feature information.

## 4.3) Results and Discussion:

- This framework is easy to use.
- Adaptable to changes.
- It is quite handy and saves a lot of time of coding.
- The pictorial representation helps to understand the dataset and situation easily making it quite in useful in many areas.
- Hence, we conclude that required goals and objective of GUI based framework for Machine Learning Algorithms can be achieved.

# Chapter-5: Conclusion And Future Plans

## 5.1) Conclusion:

One can infer from the project that the proposed technical approach can be used to make this app and can be used in various areas. However, this cannot handle very large datasets (in GBs), which may be a limitation of this.

## 5.2) Future Scope:

- Create a better user interface.
- Provide Custom Feature Extraction
- Input for custom dataset.
- Reinforcement machine learning models
- A Prediction Feature

# References


- [1]Machine Learning With Python by IBM https://www.coursera.org/learn/machine-learning-with-python/home/welcome
- [2]machinelearningmastery.com
- [3]https://www.geeksforgeeks.org/
- [4]https://streamlit.io/
- [5]https://youtu.be/Klqn--Mu2pE
- [6]https://www.kaggle.com/