EDA Tool

A Project Report

Compiled By :

Anitej Srivastava

Ref no: VT20211863

3rd Year Student

BTech – Computer Science and Engineering (Core)

School of Computer Science and Engineering

Under the guidance of

Mr. Sudipto Trivedi

Tata Steel, Jamshedpur

Introduction

The project is an all purpose **exploratory data analysis** tool. It is made as an interactive application where users can input their datasets & with a few clicks, perform EDA using various tools, visualise custom features & get predictions on the features and targets of their choice. A business has to usually employ people to analyse the crucial data that they collect in order to make more informed business plans, target the right demographic etc. The analysis requires numerous lines of code to carry out the different processes.

This application has been designed keeping in mind that the user would not have to know the technicalities of any of the processes mentioned above, rather they would only need to have knowledge & understanding of their own data as well as basic statistics to understand the end result.

Development Method

The entirety of the application has been developed using the Python stack. The UI/UX of the application has been built using PySimpleGUI which is written in python. A backend is not needed as the application can run locally on the machine once the software is downloaded if the appropriate libraries have been installed in the system.

Various Python libraries that are essential to Data Science have been used for the development. Some of the most critical ones are:

PySimpleGUI : PySimpleGUI wraps the entirety of Tkinter, which comes with Python. PySimpleGUI has wrapped most of PySide2, but only a small portion of wxPython. When you install PySimpleGUI, you get the Tkinter variant by default.

Pandas: Pandas is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series. It has been used to read and manipulate the datasets provided by the used depending on the functionalities.

Numpy: NumPy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays. It has been used to manipulate arrays, especially for performing machine learning operations such as modelling & prediction.

Matplotlib : Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, PySimpleGUI. It has been used in the application to provide visualisation using various graphs.

Sweetviz : Sweetviz is an open-source Python library that generates beautiful, high-density visualizations to kickstart EDA (Exploratory Data Analysis) with just two lines of code. Output is a fully self-contained HTML application.

Autoviz : Autoviz is an open-source python library that mainly works on visualizing the relationship of the data, it can find the most impactful features and plot creative visualization in just one line of code. Autoviz is incredibly fast and highly useful. For this application, autoviz can be interacted with in the command line itself.

Pandas-profiling : **Pandas profiling** is an open source Python module with which we can quickly do an exploratory data analysis with just a few lines of code. In short, what **pandas profiling** does is save us all the work of visualizing and understanding the distribution of each variable. Output is a fully self-contained HTML application.

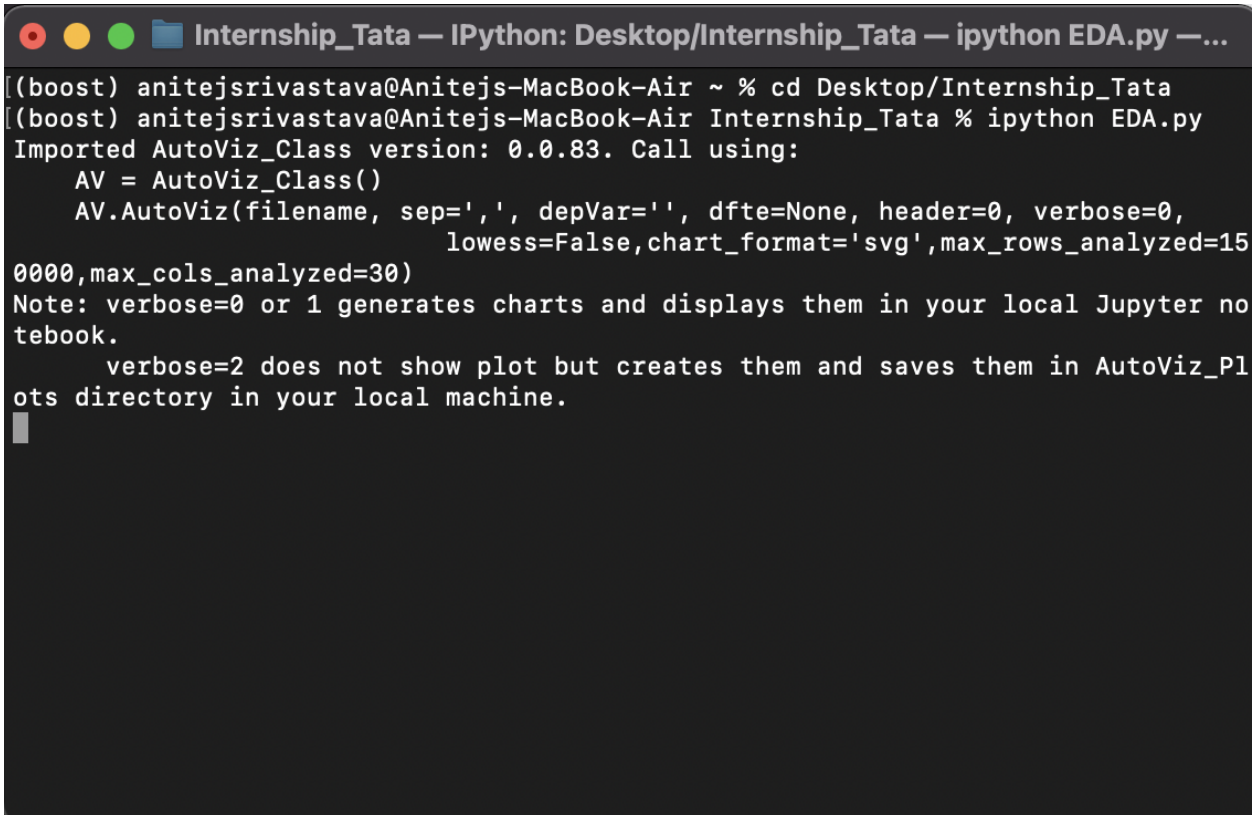
scikit-learn : **Scikit-learn** (also known as **sklearn**) is a machine learning library for the Python programming language. It features various classification, regression and clustering algorithms. Sklearn has been used in this application to provide data preprocessing functionalities like Label Encoding & One Hot Encoding. It has

also been used in the project to model & predict data using Regression and Classification on the input dataset.

Project Setup

To kickstart the project on a local machine, one would need to follow the following steps:

1) Clone the entire project using the terminal inside a directory with the command: ***git clone https://github.com/codeup729/EDA_Tool.git***
2) Install the required libraries (Python 3.8 & above should be installed)
3) cd to that directory
4) In the terminal, enter the command ***ipython EDA.py***



The result should be the following:

Application Architecture

The application consists 3 main functionalities:

1) Exploratory Data Analysis using various tools
2) Visualization of custom features of the input dataset
3) Modelling the Dataset with users' choice of features using Regression or Classification
4) Predicting the users' choice of target variable

Data Entry

The dataset can be entered with a click of the “Browse” button & the users can also see the entered data by clicking “Show Data”. The users can also re-enter the data if they mistakenly entered the wrong without reopening the application. This is all done using PySimpleGUI’s windows. The only constraint is that the input datasets have to be in the **.csv file format.**

Performing Exploratory Data Analysis

The users can select their choice of EDA tool from the drop down provided. When the “Perform EDA” button is clicked, the users’ default web browser will be automatically fired up & an HTML page containing the EDA Report would be displayed. The HTML files would be saved in the local directory & the appropriate file would be replaced when “Perform EDA” is clicked with a different dataset.

This has been achieved using the libraries **sweetviz, autoviz & pandas-profiling**. The **os & webbrowser** libraries have been used to display the files in the users’ web browser.

Visualisation

When the users click on “Visualisation options”, they get to chose the features they want to visualize and can also customize which features go on the X-axis & Y-axis. One feature can be chosen from the dropdown for X-axis while **multiple features can be chosen for Y-axis** by scrolling and clicking. The chosen features display a black background as an indication.

Users can choose from **scatter & line plots** to visualize their custom features. When the “Show” button is clicked, a PySimpleGUI window opens up to display the appropriate graphs with legends & labels. For visualisation the **data is first normalised using Min-Max Scaling** for a greater clarity on the relationship

between variables in the dataset. The visualisation has been achieved using the **matplotlib** library and Min-Max Scaling has been done algorithmically.

Machine Learning Functionalities

The users can access this by clicking on “Click here” below “To Fit a Model” Label. These functionalities require the users to have a knowledge of their dataset, what their goal is & basic analytics. The users can choose to model with **Regression or Classification**. If the target variable (prediction variable) is continuous in nature, Regression should be chosen & if the target variable is categorical in nature, Classification should be chosen.

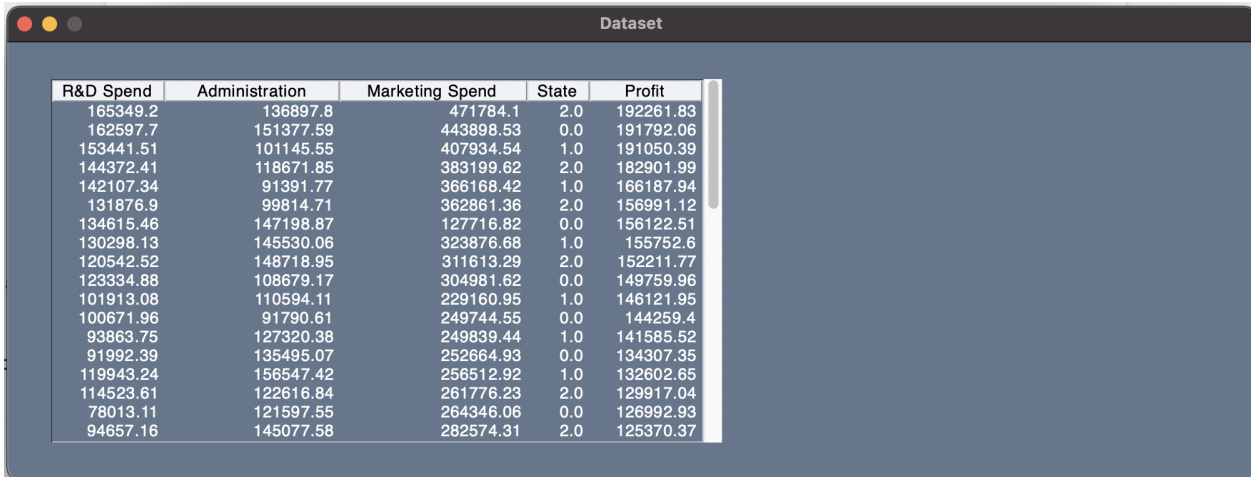
The users can choose the features (independent variables) of the dataset on which they want the model to be trained & also the target variable (dependent variable).

Once the user clicks “Fit”, the model trains itself accordingly and the accuracy of the model fit is also displayed. Once “To Predict” is clicked, the application asks the users to enter the values of the chosen features for which they want to predict the target variable. On clicked “Predict”, the predicted value for the target variable is displayed. For a model with a relationship close to linear, the accuracy is > 95%..

The **scikit-learn** library has been used to perform data preprocessing tasks such **Label Encoding** non-numerical columns & to train ML models using Regression and Classification on the dataset.

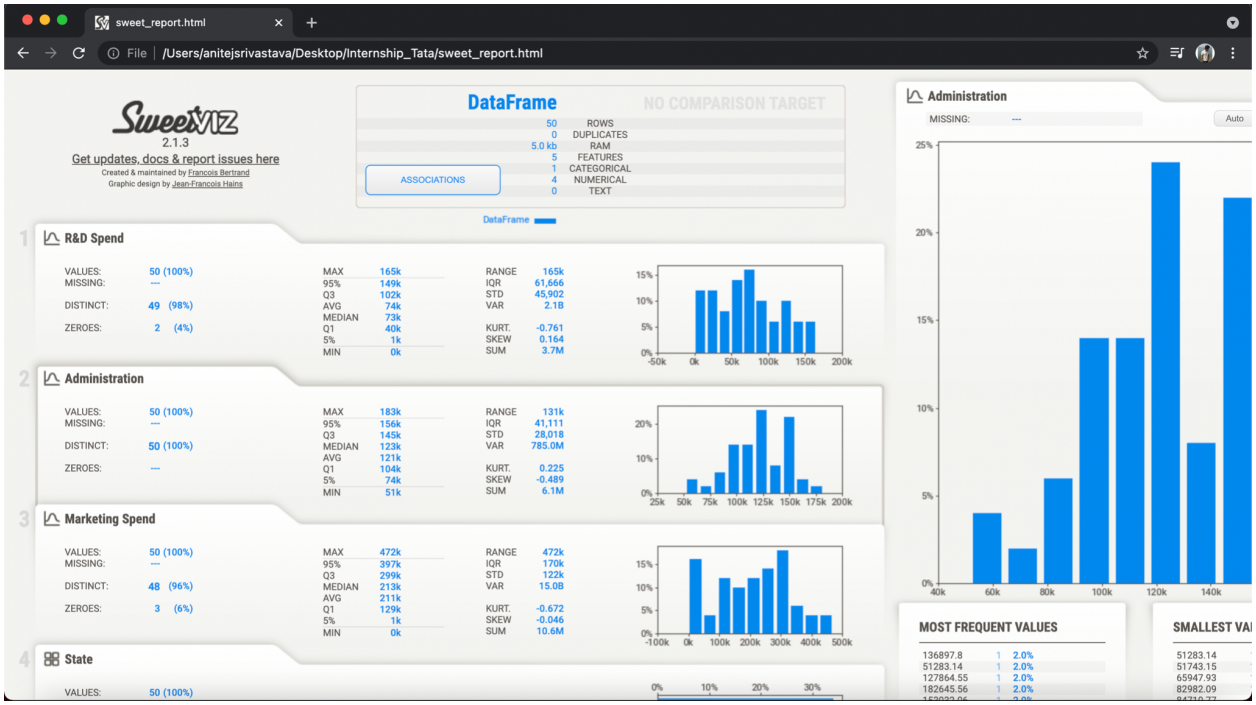
Screenshots

Show Data

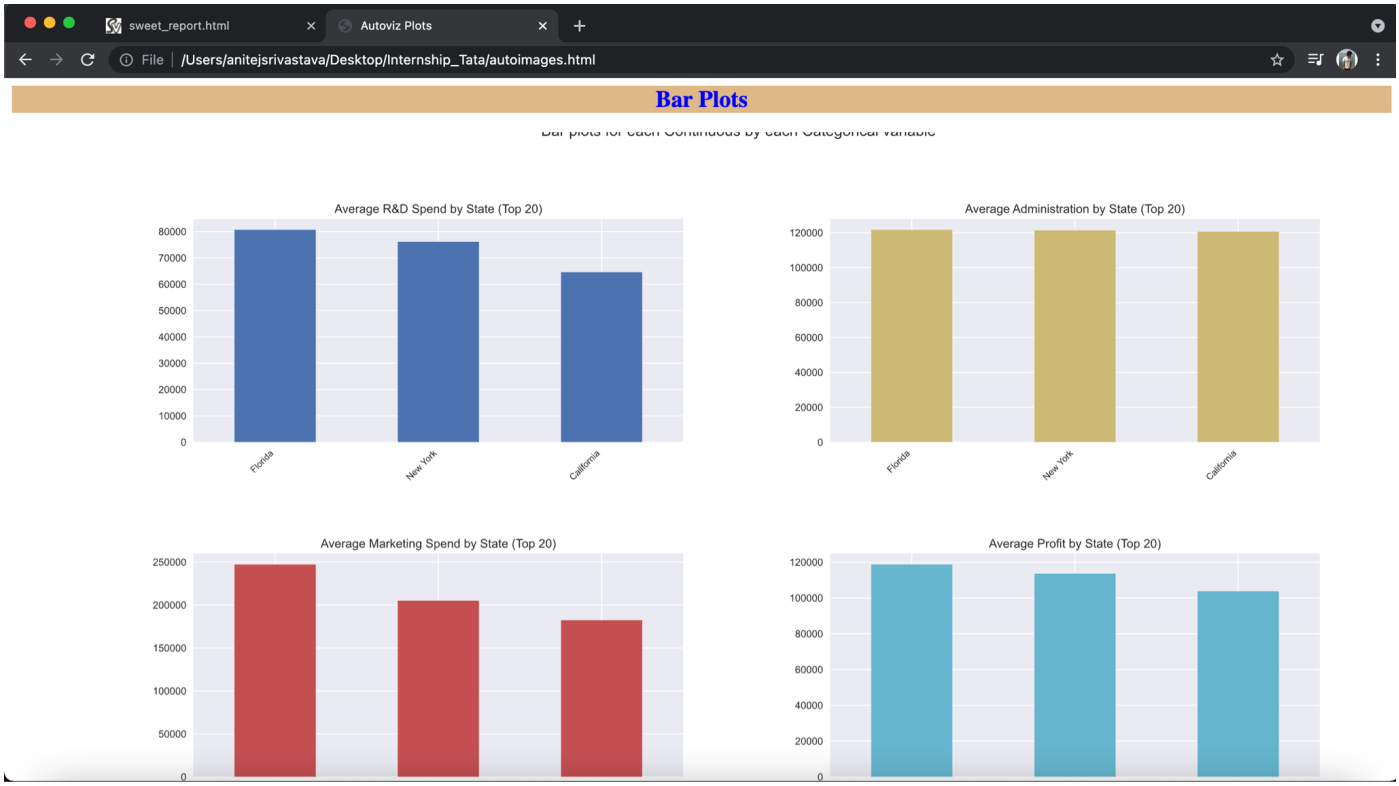


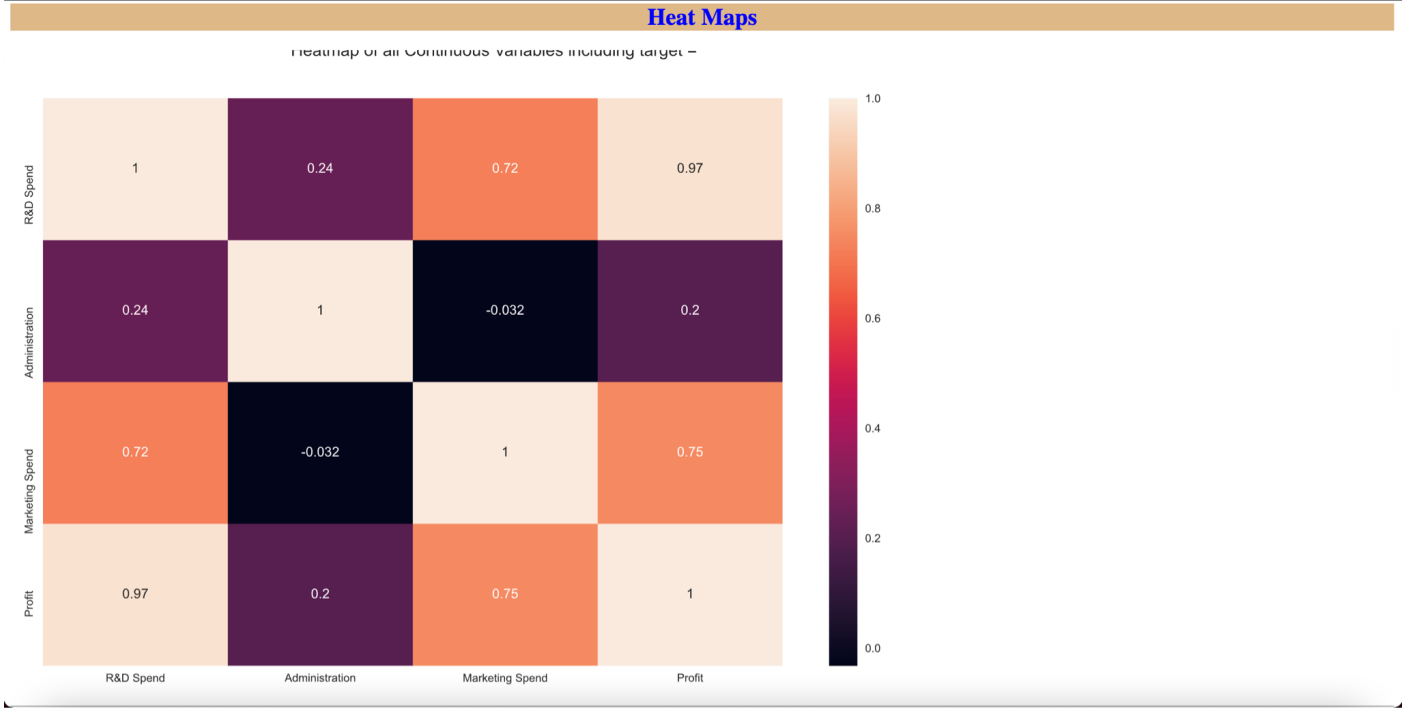
R&D Spend	Administration	Marketing Spend	State	Profit
165349.2	136897.8	471784.1	2.0	192261.83
162597.7	151377.59	443898.53	0.0	191792.06
153441.51	101145.55	407934.54	1.0	191050.39
144372.41	118671.85	383199.62	2.0	182901.99
142107.34	91391.77	366168.42	1.0	166187.94
131876.9	99814.71	362861.36	2.0	156991.12
134615.46	147198.87	127716.82	0.0	156122.51
130298.13	145530.06	323876.68	1.0	155752.6
120542.52	148718.95	311613.29	2.0	152211.77
123334.88	108679.17	304981.62	0.0	149759.96
101913.08	110594.11	229160.95	1.0	146121.95
100671.96	91790.61	249744.55	0.0	144259.4
93863.75	127320.38	249839.44	1.0	141585.52
91992.39	135495.07	252664.93	0.0	134307.35
119943.24	156547.42	256512.92	1.0	132602.65
114523.61	122616.84	261776.23	2.0	129917.04
78013.11	121597.55	264346.06	0.0	126992.93
94657.16	145077.58	282574.31	2.0	125370.37

Sweetviz

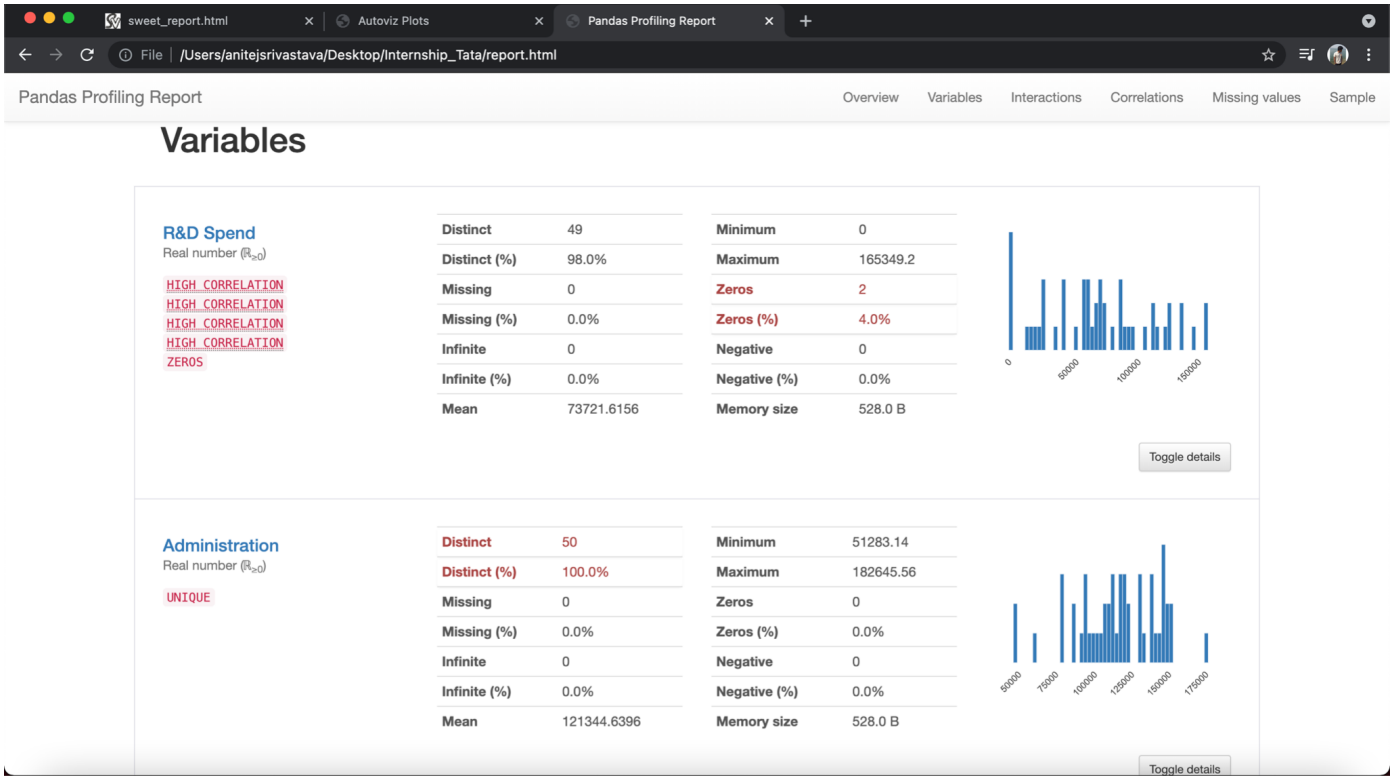


Autoviz

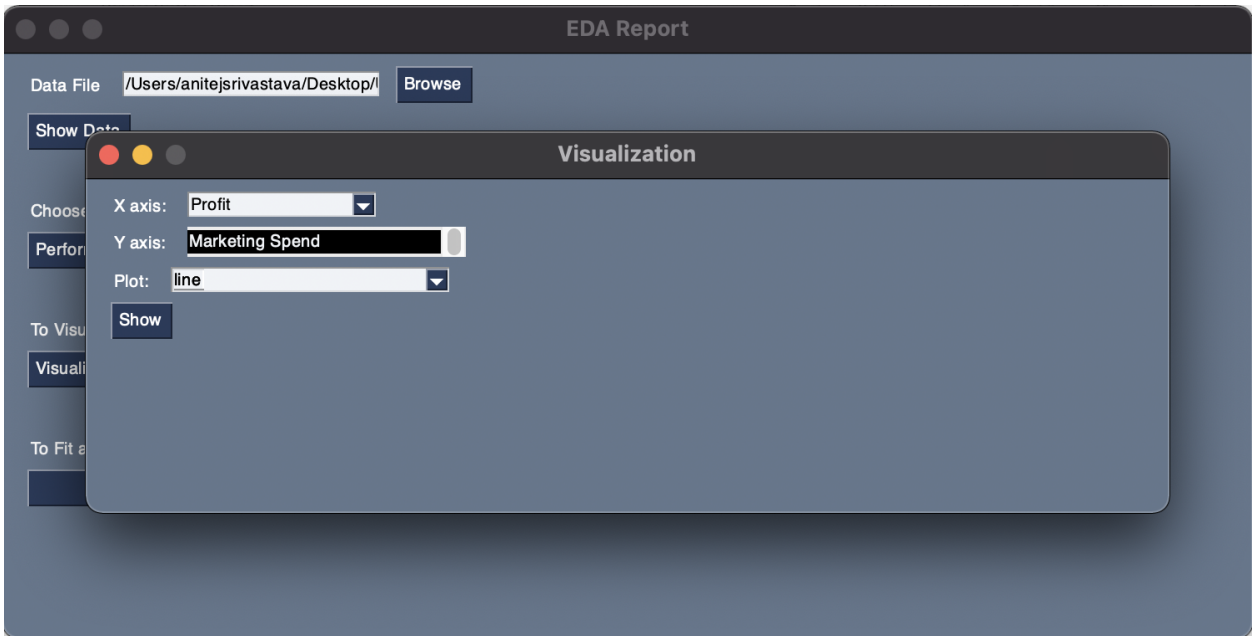


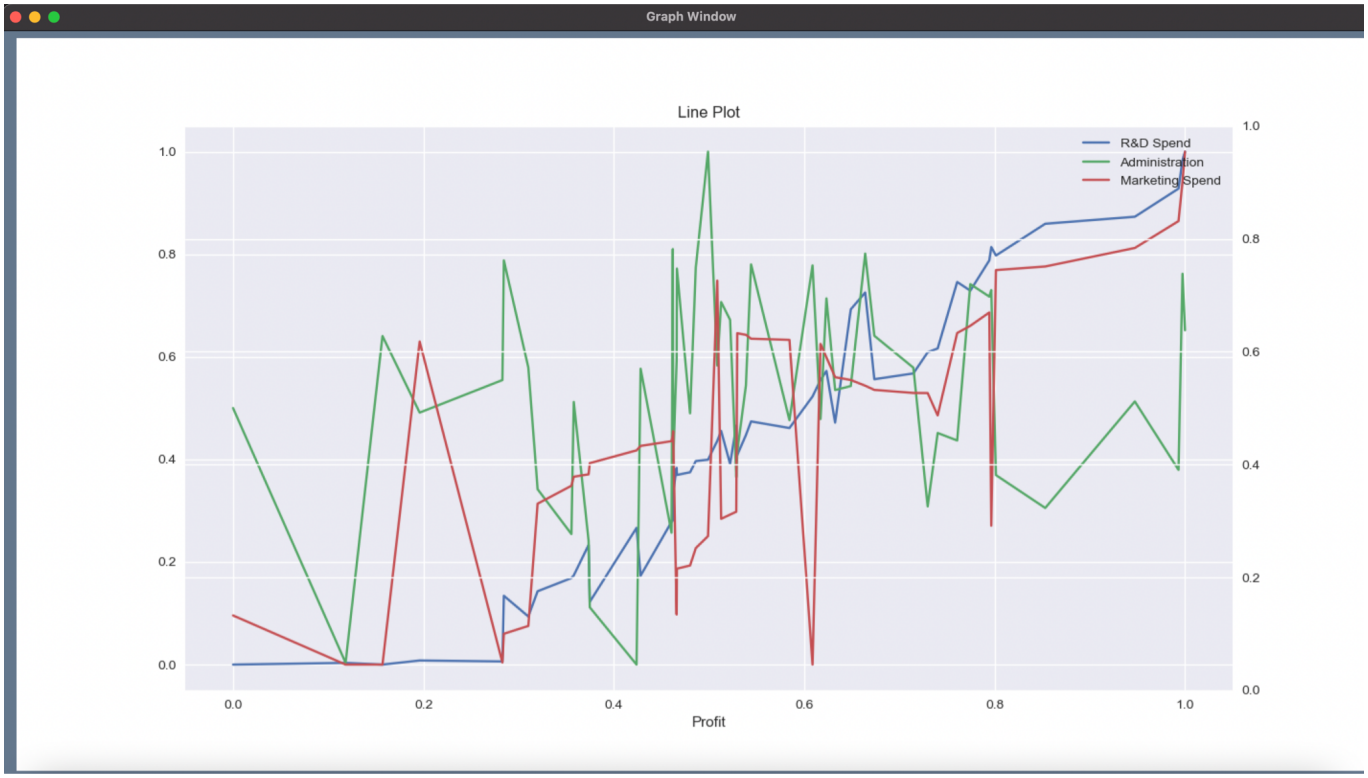


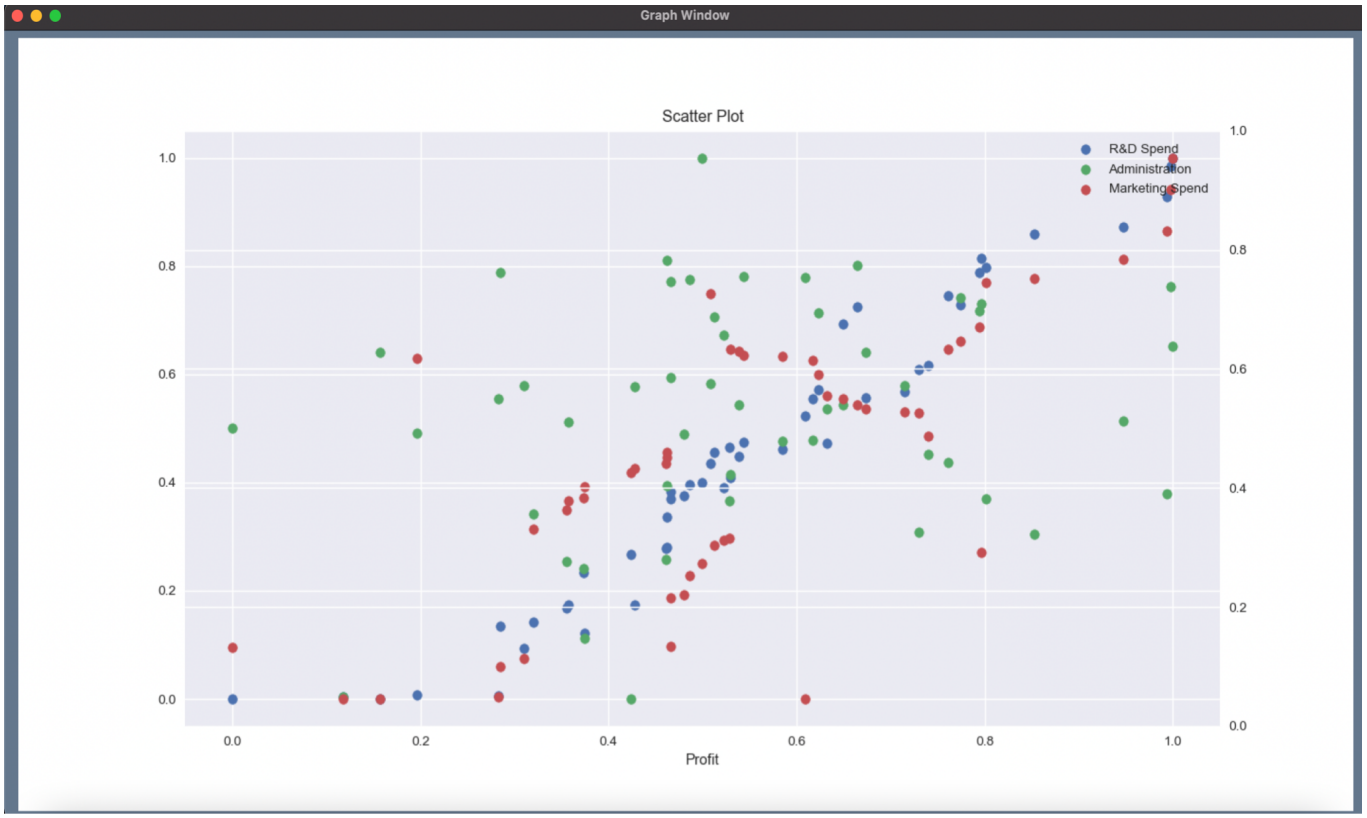
Pandas-profiling



Visualisation




Graph Window
Line Plot
R&D Spend
Administration
Marketing Spend
Profit
0.0
0.2
0.4
0.6
0.8
1.0


Graph Window
Scatter Plot
R&D Spend
Administration
Marketing Spend
Profit
0.0
0.2
0.4
0.6
0.8
1.0

Machine learning functionalities

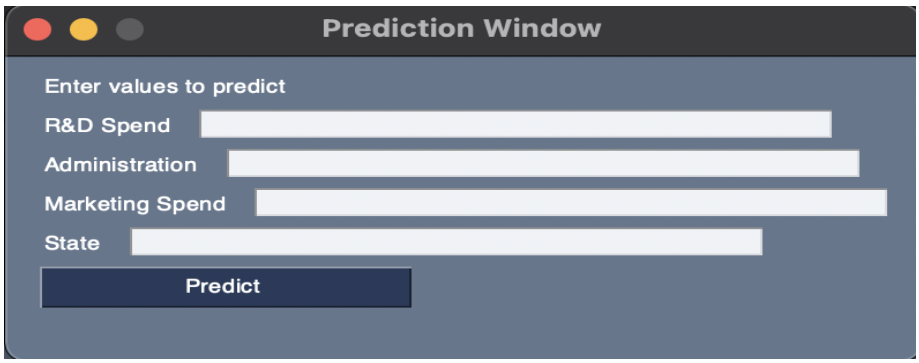
Example:

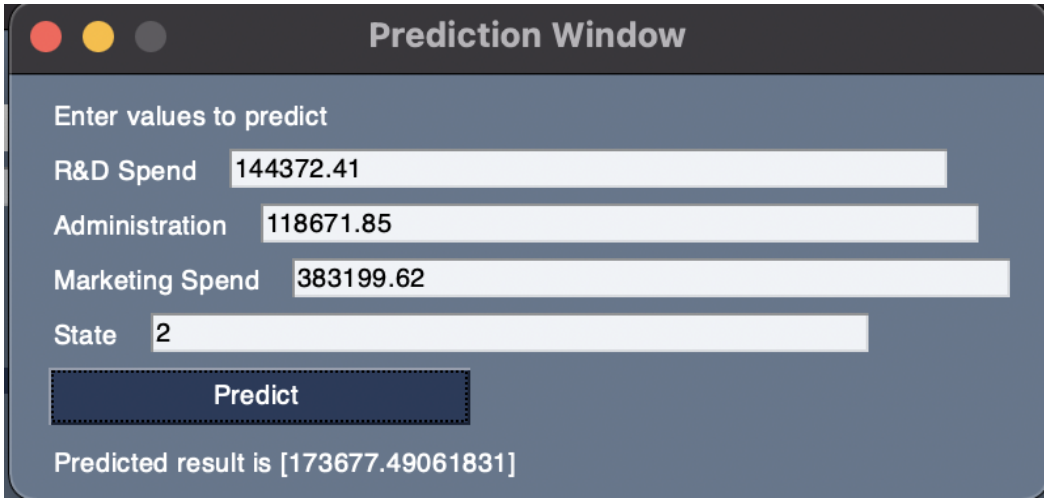
Model chosen: Regression

***Features selected*:** R&D spend, Administration, State

Target variable: Profit

Prediction





Conclusion

This project was a great learning opportunity for me. I came across various technologies and frameworks that are of great importance in the field of Data Science. I learnt about how real world problems are dealt with by trying to model solutions for the problems I faced. I gained knowledge on how to design UI using PySimpleGUI, learnt about various EDA tools & their significance in industries. I learnt about normalization techniques needed for clearly visualising the data & also learnt how to use pandas & numpy more effectively. The most important thing for me was learning how to implement machine learning on real time data sets which generally have various anomalies. To conclude, through this project I gained a lot of knowledge regarding Data Science & its use in the industry with the help of my mentor, Mr. Sudipto Trivedi.

References

https://pysimplegui.readthedocs.io/en/latest/

https://towardsdatascience.com/sweetviz-automated-eda-in-python-a97e4cabacde

https://towardsdatascience.com/exploratory-data-analysis-with-pandas-profiling-de3aae2ddff3

https://scikit-learn.org/stable/supervised_learning.html#supervised-learning

https://towardsdatascience.com/integrating-pyplot-and-pysimplegui-b68be606b960

https://towardsdatascience.com/building-data-science-gui-apps-with-pysimplegui-179db54a9a15

https://towardsdatascience.com/categorical-encoding-using-label-encoding-and-one-hot-encoder-911ef77fb5bd

https://towardsdatascience.com/autoviz-automatically-visualize-any-dataset-75876a4eede4

https://towardsdatascience.com/using-conda-on-an-m1-mac-b2df5608a141

https://towardsdatascience.com/install-xgboost-and-lightgbm-on-apple-m1-macs-cb75180a2dda

GitHub Link

https://github.com/codeup729/EDA_Tool