Model Deployment

Deployment of an ML-model simply means the integration of the model into an existing production environment which can take in an input and return an output that can be used in making practical business decisions.

To deploy a model there are various web-framework available which depends on in which language the model has been developed.

Deployment of Python code:

Deployment of python ML-model can be done by using few modules.

1. **Pickle**.
   Pickle is a python library which is used to serialize and de serialize objects in Python. Any object in Python can be pickled so that it can be saved on disk.

2. **Flask**.
   Flask is a micro web application framework written in Python. It is designed to make getting started quick and easy, with the ability to scale up to complex applications.

Instruction for deployment:

1. Environment Setup

We need to install the necessary packages for the deployment. These packages are

- Flask
- Pickle
- Pandas
- Numpy
- Scipy
- Sklearn
- Seaborn
- Matplotlib
- Lightgbm
- Imblearn

To install the packages we can use pip command.

pip install flask, pip install pickle,…..

2. Directory Control

Put model.py, train.csv, app.py, static folder, templates folder in same directory.

3. Model Development

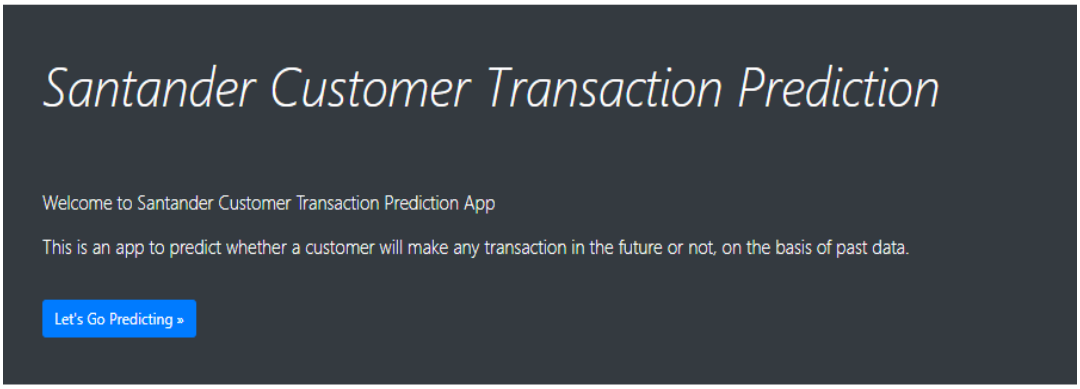
Develop
Santander_Prediction_model.model,
scaler.model,
columns_to_drop.list
By running the **model.py.**

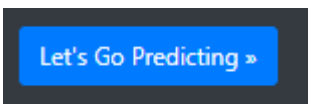
- **To run** Open the DOS command shell & direct to the directory containing **model.py**
- Run the **model.py** by using command
  >> python **model.py**

4. Deployment using Flask

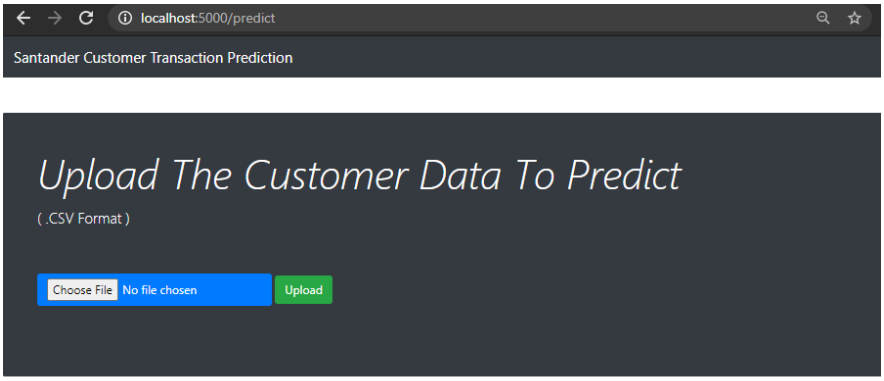
- Open the DOS command shell & direct to the directory containing app.py.
- Run the app.py by using command
  >> python **app.py**

  This will deploy the model on the local server : "http://127.0.0.1:5000/" or "http://localhost:5000/".
- Open "http://127.0.0.1:5000/" or "http://localhost:5000/" in the browser. This will open the homepage.



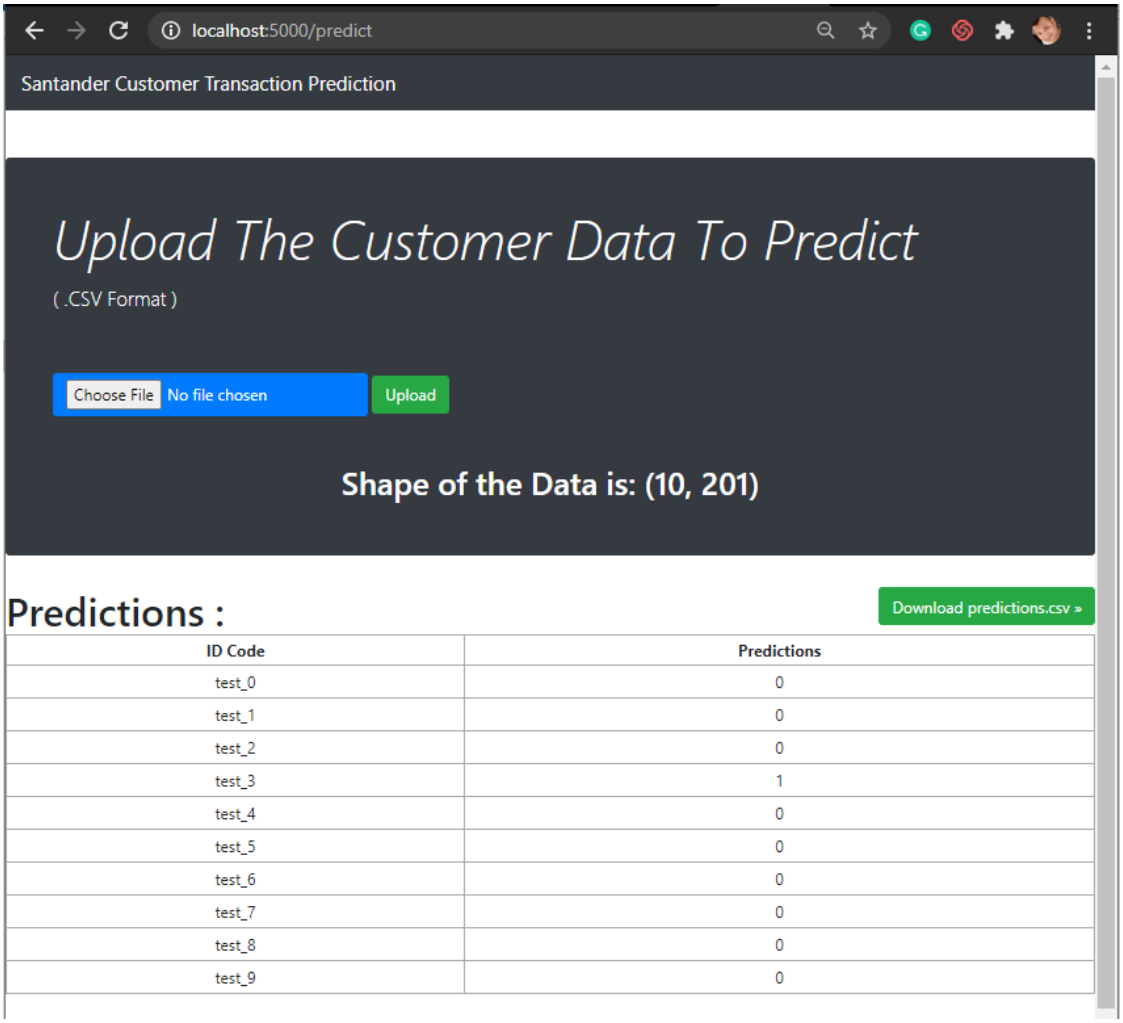
After that click on  button.

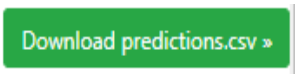
- This will direct you the predict page “http://localhost:5000/predict”



- Click on **Choose file** & select the .csv file for predictions & click **Upload**.

- After upload the shape of the data & predictions will be displayed on the screen.



- Click on  to download the predictions result in .csv format.

**********END**********

Deployment of R code:

Deployment of R ML-model can be done by using RShiny.

Shiny is an R package that makes it easy to build interactive web apps straight from R. You can host standalone apps on a webpage or embed them in R Markdown documents or build dashboards. You can also extend your Shiny apps with CSS themes, htmlwidgets, and JavaScript actions.

Instruction for deployment:

1. **Directory Control**

   Put Santander_Customer_Transaction_Prediction.R, app.R, train.csv files in the same directory.

2. **Model Development**

   Develop
   final_model.rds,
   column_toDrop_list.rds
   By running the **Santander_Customer_Transaction_Prediction.R**

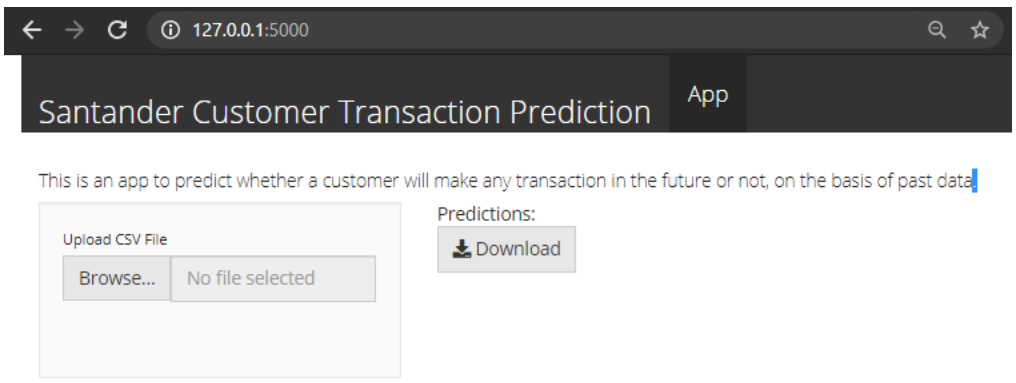
   - To run Open the DOS command shell & direct to the directory containing **Santander_Customer_Transaction_Prediction**.R.
   - Run the **Santander_Customer_Transaction_Prediction.R** by using command
     >> Rscript **Santander_Customer_Transaction_Prediction.R**

3. **Deployment**
   - Open the DOS command shell & direct to the directory containing app.R.
   - Run the app.R by using command

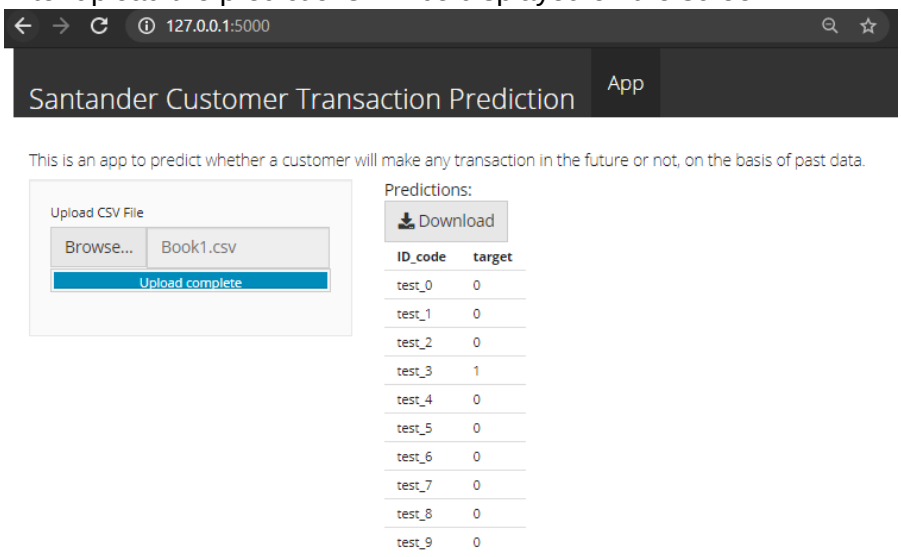
     >> Rscript **app.R**

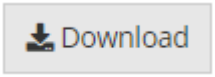
     This will deploy the model on the local server : "http://127.0.0.1:5000/" or "http://localhost:5000/".

   - Open "http://127.0.0.1:5000/" or "http://localhost:5000/" in the browser. This will open the web application.



- Click on **Browse** & select the .csv file to be uploaded for predictions.
- After upload the predictions will be displayed on the screen.



- Click on  to download the predictions result in .csv format

**********END**********