WiDS ‘22 - ‘23 Final Documentation

<Project UID - Name>
<Mentors>

Team Member Name	Roll Number	Email-Id
Aadarsh Dwivedi	19D180001	19d180001@iitb.ac.in

Introduction to Problem Statement

In order to prevent customers from being charged for products they did not buy, credit card firms must be able to identify fraudulent credit card transactions. Building a machine learning system in Python with the ability to distinguish between legitimate and fraudulent credit card transactions is the goal.

Existing Resources

Implementing a machine learning system to identify credit card fraud using a dataset of user-generated credit card transactions is the aim of this research. This dataset contains transactions from two days in September 2013, with a total of 284,315 transactions, 492 of which were fraudulent. As a result, the dataset is very imbalanced, with frauds making up barely 0.17% of all transactions. This will be covered in more detail when we talk about the ideal model and sample plan. The dataset comprises one goal variable and 30 input characteristics, 28 of which have been made anonymous. Only two of the 30 input features in the imported dataset below, Time and Amount, have labels. Due to this, we cannot do EDA on the majority of the features. The histograms of each of the characteristics below are displayed after we have constructed the feature variable (X) and the target variable (y). We can observe that the feature values that aren't labelled have undergone a PCA transformation. Amount and Time (between transactions) are not, with the latter showing a bimodal distribution (with two modes: one mode around 50K seconds, or 13.89 hours and the other around 150k seconds, or 41.67 hours). Let's next examine the distribution of class type types. It appears that the dataset we have has a huge non-fraudulent skew. The dataset we have appears to be significantly skewed toward legitimate transactions (284,315) when compared to fraudulent ones (492). The dataset includes credit card transactions performed by European cardholders in September 2013. We have 492 frauds out of 284,807 transactions in our dataset of transactions that took place over the course of two days. Only numeric input variables are present. The major components derived with PCA are features V1, V2,..., V28. The only features that have not been changed with PCA are "Time" and "Amount." The seconds that passed between each transaction and the dataset's initial transaction are listed in the feature "Time." The transaction amount is represented by the feature "Amount," which may be utilized for example-dependent, cost-sensitive learning. The response variable, feature "Class," has a value of 1 in cases of fraud and 0 in all other cases.

Proposed Solution

First the dataset is prepared and summarized by printing the first five rows of the dataset. Then some information is extracted from the dataset and the distribution of the fraudulent and legitimate transactions is derived. Then the two list variables are created each storing the information about the legitimate and the fraudulent transactions. The data attributes like count, mean, quartiles, etc. are then obtained from the variables. Then the under sampling is done where the two datasets of equal numbers of fraudulent and legitimate transactions are chosen. Then the data is split into the learning and the testing data and 20 to 80 percent ratio is maintained. Then the data is standardized and then the model is trained. There are several algorithms that can be used but due to the high efficiency, for this project, I have used random forest classifier and decision tree model as these give good accuracy. Many other algorithms have also been tried in the project but these have been the most efffective.

Methodology & Progress (Mention the work done week-wise)

In the week 1 I was basically learning about the credit cards and how they work and why does fraud occur in these supposedly robust and efficient systems. I also spent time in revising the basics of working in jupyter and ran some dummy algorithms.

In the second week I reviewed the various packages like numpy, matplotlib and pandas and did online courses on the same to brush up my concepts. I also was doing AI so I also ran some algorithms from that front like neural networks.

In the third week I worked on data analysis, data visualization and under sampling and then started running the algorithms during the day on my desktop. I was finished with this by the end of the week and was satisfied with my effort.

In the fourth week I fine-tuned my code and searched online for improved versions and incorporated these in my code. I also prepared the report and was done and finished with the project by the end of the week.

Results

https://github.com/AadarshDwivedi12/Credit-card-fraud-detection

Thus the decision tree algorithm shows the accuracy of 90 percent with the test data and the random forest algorithm shows the accuracy of 90 percent with test data while they both show 100 percent accuracy with learning or training data and so these are the recommended algorithm for this project.

Learning Value

Thus the learning value chain is I learnt the algorithms and the various packages and then also learnt the data visualization, standardization and data seperation and training and testing.

Tech-stack Used

Jupyter and anaconda with compilation help from pycharm.

Suggestions for others

NA

Contribution by each Team Member

NA

References and Citations

https://jupyter.org/try-jupyter/retro/consoles/?path=console-1-2980471a-a8e8-4d08-8e78-f356a4eff7d7
https://github.com/AadarshDwivedi12/Credit-card-fraud-detection
https://jupyter.org/try