Credit Card Fraud Detection Using Anomaly Detection Techniques

Anubhav Sharma
Data Science, University of Colorado Boulder
anubhav.sharma@colorado.edu

ABSTRACT

In recent years due to COVID Pandemic and increasing growth of computing power the whole world has seen a wider acceptance of Artificial Intelligence and Machine Learning Technologies and this has not only led to good things but also bad things like increasing number of Credit Card Frauds. In US alone Credit Card Frauds have increased by more than Three Times from 2018 to 2022.

Credit Cards [1] are generally safer as compared to Debit Cards, Cash, and many other methods while being more convenient and this has led to a wide acceptance of Credit Card throughout the world, and with new AI technologies they are clear targets for perpetrators for Credit Card Frauds and even Identity Theft. Since perpetrators are using new ML powered techniques there is even greater need for Banking Companies to revamp their Credit Card Fraud Detection Techniques.

In this project we covered multiple Anomaly Detection Techniques, from Traditional ones like IQR to Deep Learning based Auto-Encoders with Focal Loss, and aimed to test and compare Nine such techniques on a Real World Credit Card Fraud Dataset. Our goal is to compare multiple popular techniques and provide a model which is able to detect Credit Card Fraud with a high Recall Value. For this we performed proper Data Preprocessing, EDA, and Evaluation 16 different Models or techniques to get various results, from 0.418 Recall in basic IQR Method to Auto-Encoder Model with Focal Loss reaching Recall of 1.0. That is the best Model of this project was able to flag every single Fraud Transaction in the Test Dataset.

1. INTRODUCTION

Anomaly Detection [3] is the task of identifying unusual instances that deviate significantly from the majority of data. Also called Outlier Detection we can use this task to identify different types of Outliers such as Global Outliers(Point Anomalies), Contextual Outliers (Conditional Anomalies), and Collective Outliers. We can use Anomaly Detection on various domains, to uncover these outliers, such as intrusion detection, rare disease detection, network anomaly detection, and financial fraud detection. In this project we will focus on one type of Financial Fraud Detection: Credit Card Fraud Detection.

Credit Card Frauds are harmful for both customers and companies, while some customers are able to take preventive measures most of them face issues like identity theft and credit card decline for crucial purchases made by the customer because of breach of Credit Card limits by the Perpetrators [2], and for companies these events show the vulnerability in their system and this may lead to decrease in confidence in the companies and overall financial systems. And due to increased overall computational power new novel methods are being developed to perform these crimes, this is leading to increasing number of such attacks, from 2018 to 2022 Credit Card Fraud has increased by multiple of 3x and even more in US alone and it is still rising [4].

With such rising trend Credit Card frauds are not something rare event for financial institutes but a real threat to the financial safety of both customers and institutions. To defend against it while financial institutes and governments have introduced stricter online banking by adding 2 Factor Authorization and Credit Card Chip Technology [5] due to which it is getting difficult to steal data customer key information from the Credit Card itself. But even then as we are turning into a much more prevalent online environment these methods are not fail-proof as Data Breaches have exposed hundreds of millions of Credit Card information till now and this number can only increase over time.

Due to this we need not only to physically prevent the Credit Card Fraud we also need to prevent them digitally and traditional outlier methods such as Inter Quantile Range, Standard Deviation Method, and Z Score Methods are not suitable for increasingly complex Credit Card Fraud in the Big Data age. We need better techniques and in this project, we will cover many of them, ranging from simple Logistic regression to Isolation Forest to Deep Learning Networks such as Auto-encoders.

We will apply these techniques on the Credit Card Fraud Dataset provided by Worldline and the Machine Learning Group – ULB [6] as this one of the public Credit Card Fraud Dataset which is not just simulated data and contains transactions made by credit cards in September 2013 by European cardholders. This dataset contains PCA components instead of real features for these transactions due to the anonymity concerns. In total dataset has 284,807 transactions which has only 492 frauds.

The techniques covered in this project are not only able to handle the new huge amounts of Data but also provide more effective methods which are far better than traditional methods for Outlier Detection/Anomaly Detection. We will apply them along with the traditional methods and evaluate the best based on evaluation metrics explained in further section in the report, in the end we will obtain a technique which is able to perform Credit Card Fraud Detection in best possible way among many of such techniques in this new Information Age.

2. Related Work

Outlier Detection or Anomaly Detection has been a topic attracting focus in multiple domains, especially finance, and this has led to the continuous research in this field and this had resulted in multiple established pre-existing techniques [6, 7]:

2.1 Modified Z-Score

Z Score thresholds are most commonly used Statistical Tool for Anomaly Detection and has been included in the toolkit for a long time. Modified Z-Score [8] is an improvement on it as it is based on Median which may help in increasing the robustness of the technique as it is not influenced by the outliers to the level of the mean.

Modified Z Score is given by:

$$0.6745(x_i - \tilde{x})/MAD$$

Where x_i is the single data value, x_tilde is the median of the dataset, and MAD is the Median Absolute Deviation of the Dataset.

We can also use MeanAD [9] and mean instead of MAD and Mean instead of x_tilde. Nevertheless, using this technique points having Modified Z Score of 3.5 or more are considered anomalous.

2.2 Interquartile range (IQR)

Another basic Statistical method which is also tried and tested like Z Score. Here data points within IQR range are considered normal and points beyond it as anomalous. IQR is generally 3rd Quantile – 1st Quantile however we can adjust it to use nth and mth tentile.

IQR being used with Box Plots is one of the most basic visual tool for Anomaly Detection and we will use it in the project too as a form of traditional Anomaly Detection techniques.

2.3 Density-Based Spatial Clustering of Applications with Noise (DBSCAN)

DBSCAN is one of the most famous Machine Learning techniques for Anomaly Detection. It came into light during KDD Conference 1996 [10] and is still widely used for Anomaly Detection.

It is a better clustering technique than K means as it allows us to form better clusters overall. In DBSCAN we use Core Points and Border Points as the basis of clustering rather than centroids in the K Means. We fine tune the Hyper Parameter of Border Points and radius, this helps us define the Core Points, for example if Border Points are 4 then a Core Point is a point which has at least 4 points around it within the given radius.

After defining the core points, we start from random and start making cluster. Every core point which falls under the boundary of the previous one gets assigned to the same cluster; we iterate it until we get only Non-Core points that is the points which has less than minimum points within its boundary. If we are left with more core points then we randomly initiate the clustering process once again and form other clusters.

In the end we attach the Non Core Points connected to the Core Points of each clusters to the respective clusters, this will leave us with the Non Core Points which will not fall within any cluster and they are our Anomalies.

Detailed Algorithm for DBSCAN and more details can be found in the paper "*A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise"* written by Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu (paper has been given as a reference in the point number 10.)

2.4 LOF (Local outlier factor)

Local Outlier Factor or LOF is another Unsupervised Outlier Detection algorithm. While having a similar logic to KNN and DBSCAN, LOF assigns a metric (LOF) to each datapoint, normal points would have a score somewhere between 1 and 1.5 while outliers have a higher score.

It is a widely used tool and has been adopted in the popular ML packages such as sklearn [16]. We will use the same in our project.

2.5 Isolation Forest/iForest

Isolation Forest or iForest as called by the authors [12] in the 2008 paper is an unsupervised Machine Learning technique for classification which is called as possibly the best technique for Anomaly Detection in recent years.

Isolation Forest is efficient in Big Data applications, and perform favourably as compared to the ORCA, a near-linear time complexity distance-based method and in term of efficiency and Area Under Curve or AUC it is far better than Random Forest and LOF we discussed above. It is able to achieve good results with near constant efficiency because of its approach, it does not focus on profiling the data points but focus only on separating the outliers.

Outliers are important observations in the areas where most of the time we have normal data points, they are not meant to be removed or changed just because they exist but should be analyzed further as they reflects clear Anomalies and this is also the reason why we are doing this Credit Card Fraud Detection using Anomaly Detection techniques, the few 100 outliers in front of hundreds of thousands of normal observations are object of interest. And Isolation Forest focuses on exactly this factor.

In iForest we use tree structure such as Random Forest to identify the outliers, we use the fact that by using random cut offs we can get the leaves containing outliers much earlier in the overall path. Due to this Anomalies become the instances which have shorted average path from the root. Each observation is assigned an Anomaly Score which is the average of the path from the root and smaller the Anomaly Score means more likely that we have an Anomaly in our hand.

Isolation Forest does not try to model the data points or try to find the representation, but it focuses on highlighting the outliers in efficient and effective manner making it one of the best technique for Anomaly Detection even now. We will use it along with other techniques in our project and compare it against other techniques.

2.6 Deep Neural Network

Credit Card Fraud Detection involves binary classification, and this opens the path for Logistic Regression. We can apply it after balancing the dataset or even better we can use a form of Deep Neural Network, namely Dense Neural Network containing Dense units and Relu Activations for hidden layers. It is a standard Neural Network which is able to learn the representations better than the Logistic Regression and we will use this in our project.

3. Proposed Work

The whole project is divided into following parts following Plan-Analyze-Construct-Execute (PACE) Framework:

1. Data Procurement and Preprocessing
2. Exploratory Data Analysis (EDA)
3. Application of Anomaly Detection Techniques
4. Evaluation
5. Conclusion

We needed a Credit Card Fraud Dataset for our project, and we can either try to procure the real dataset or use the stimulated dataset. While the real-life dataset containing the information of real-life transaction sounds best it is to be noted that it is not only very difficult to obtain it as such information is sensitive and companies are even legally required to safeguard them across the world. We can use simulated dataset, but it has its own fair share of problems as data is created synthetically.

In the project we have decided to hit a middle ground and use the 150.83 mb dataset [6] containing anonymized transactions made by credit cards in September 2013 by European cardholders. This data set is based on real life transactions and also solve the ethical concerns we had because not only the identification information has been removed PCA analysis have been already done to replace the original columns, leaving us with 28 PCA components, time, amount, and class columns.

The dataset has been collected and analyzed during a research collaboration of Worldline and the Machine Learning Group of ULB (Université Libre de Bruxelles) and contains 284,807 transactions out of which only 492 transactions are fraudulent. While in the current landscape with increasing number of fraud incidents we can definitely see higher proportion of the Credit Card Fraud detection but this will not impact much in our methodologies for this project.

The dataset is in form of .csv file and Anomaly Detection will be performed natively on a CUDA enabled RTX 4060 Laptop GPU which has 8 GB memory. We will perform all the necessary steps given belon in Python 3.11 on WSL2 and utilize Tensorflow for Deep Learning based Techniques and Scipy for the ML based techniques and other packages as required

3.1 Data Procurement and Preprocessing

This stage has been completed. We have obtained the Credit Card Fraud Dataset. Dataset has 31 Columns, namely Time, Amount, Class, and V1-28 columns which are result of PCA performed to anonymize the data.

We performed following Data Preprocessing steps on the data:

1. Checked NA Values
2. Checked Duplicated Values
3. Scaled the Amount and Time with RobustScaler from skit learn package.

But these three are not the end of Preprocessing Step. Since our transactions of interest, namely fraud transactions, are only representing 0.172% of the dataset we are in clear need of Balancing the Dataset. We have two options here, Under-Sampling the Normal Transaction and Oversampling the Fraud Transactions.

For this project we performed Oversampling as Under-Sampling the Normal Transactions will severely restrict all the models. We applied Borderline SMOTE [17] which works similarly to traditional SMOTE but with a few caveats. To overcome the shortcoming of SMOTE, it identifies two sets of points — Noise and Border. A point is called “Noise” if all its nearest neighbors belong to a different class (i.e. the majority). On the other hand, “Border” points are those that have a mix of majority and minority class points as their nearest neighbors.

When the sampling is done with basic SMOTE, only the Border points are used. Afterwards, when finding the nearest neighbors, the criteria of selecting only points belonging to the same class is relaxed to include points belonging to any class. This helps select points that are at risk of misclassification and new points closer to the boundary.

We utilized the Borderline SMOTE to create a Training Dataset where we have Oversampled Fraud Transactions leading to approximately 50-50 split between Normal and Fraud Transactions.

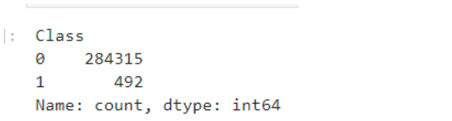
Due to this we will have three Datasets of concern, **Training Balanced Dataset, Training Un-Balanced Dataset**, and **Test Dataset** which is Unbalanced as it is necessary to test the various techniques over the real world transactions on which they will be applied and if we apply Borderline SMOTE to even Test Dataset it will not be a representative of Real World Transactions as Fraud

Transactions do not amount to 50% of all the Credit Card Transactions in Europe or World.

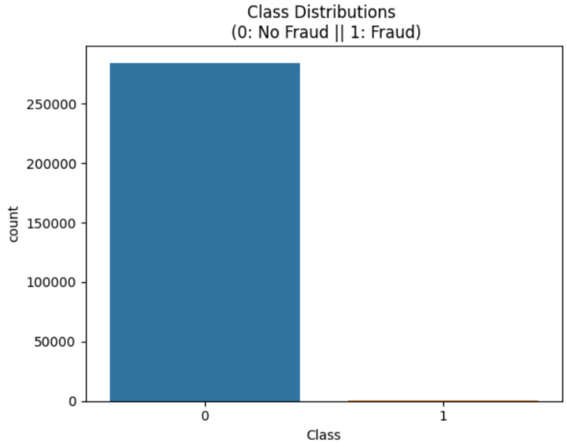
3.2 Exploratory Data Analysis (EDA)

Balancing will be done before Applying the Techniques, however at this stage we have finished with Exploratory Data Analysis or EDA. We will share key findings of EDA in this section.

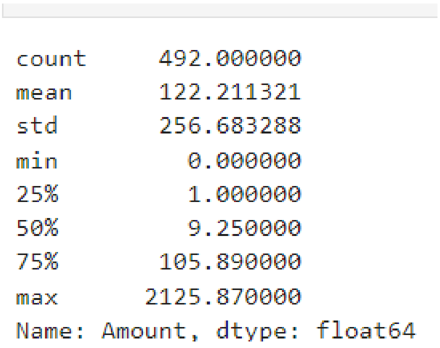
We have total 284,807 observations and total 31 columns with following class distribution:



```
Class
0    284315
1       492
Name: count, dtype: int64
```

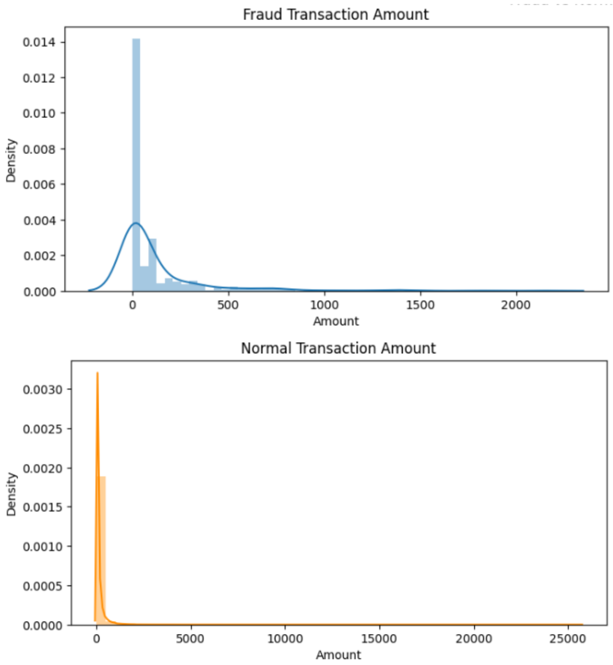


Now we will focus on Fraud transactions:

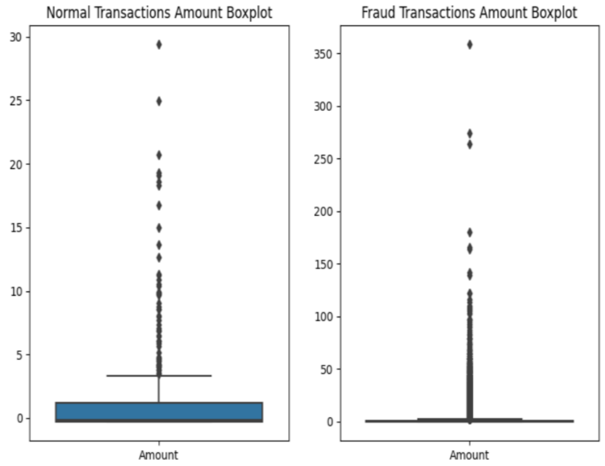


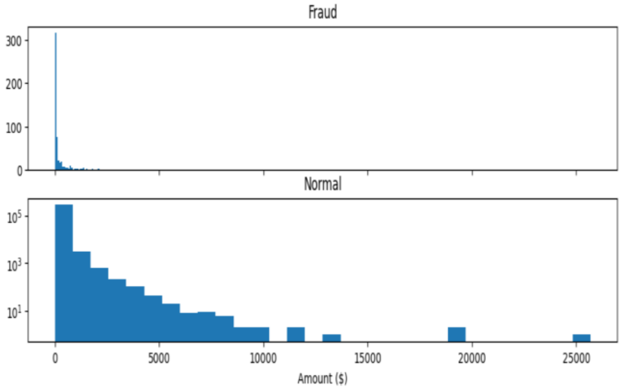
```
count     492.000000
mean      122.211321
std       256.683288
min         0.000000
25%         1.000000
50%         9.250000
75%       105.890000
max      2125.870000
Name: Amount, dtype: float64
```

Among 492 Fraud Transactions mean is 122.21 while the highest recorded Fraud Transaction amounts to 2125.87. In contrast, mean of the Normal Transactions is 88.29 which is much less than 122.21. However overall both Normal Transactions and Fraud Transactions have visibly similar Skewed Distributions with most transactions below 500:



Both distributions could have been even more similar if we had more sample points for fraud transactions.

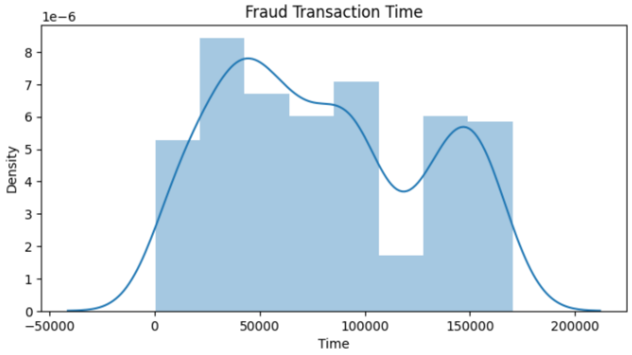


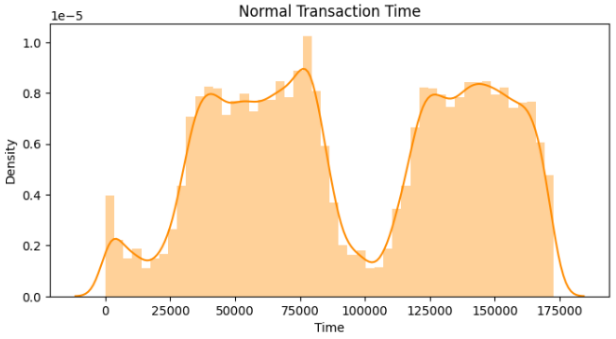


And from the additional graph above we can confirm that most number of transactions of both Fraud and Normal

Transactions are below 2000-3000 with more obvious outliers in Fraud transactions as we can see from the Boxplots.

After looking at the Amount Column differences we will now look the Distributions for the Time column which reflects the time difference between Normal and Fraud Transactions respectively.

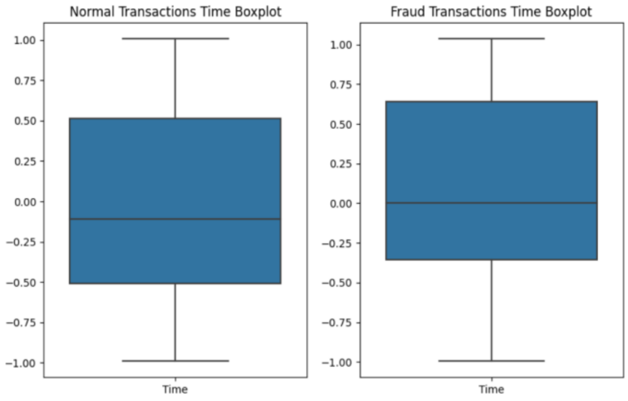




From the above graphs we can see that both are cyclic in nature with Normal Transactions having more prominent cyclical patterns. But still, there are no specific peaks in Fraud Transactions distribution as both graphs are peaking at same time, hence there is no evidence that Fraud Transactions happen more or less at some specific time interval.

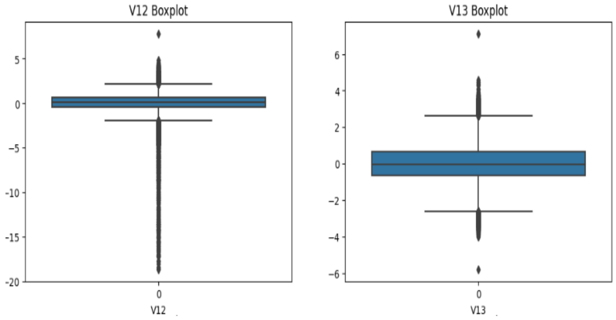
However, it is to be noted that this Data is for One month only, that too of the September 2013 in Europe. It is possible that we may observe concrete temporal patterns with more data.

Now we will observe the boxplot of Time for more details.

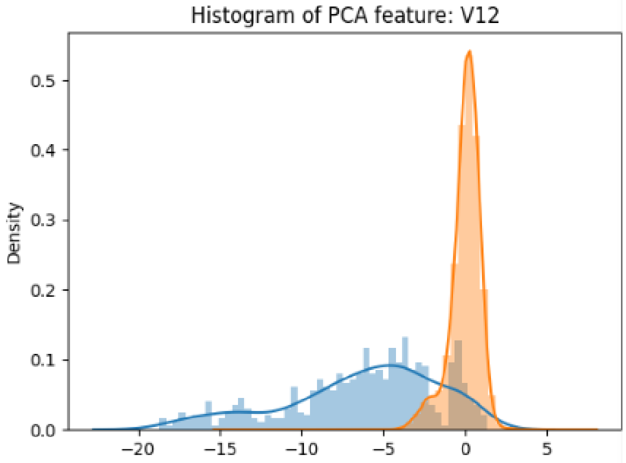


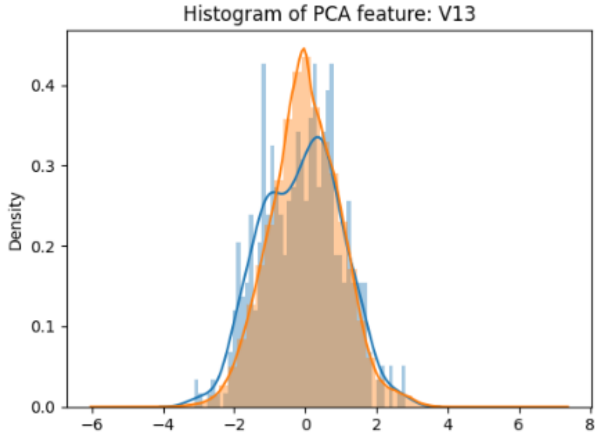
Boxplot above is corroborating with our previous conclusion, distribution of both Normal and Fraud Transactions are similar with no obvious Outliers in both of them.

Now we will focus on the PCA columns, all of them have been already preprocessed and are mysteries to us. But still we will try to analyze them in this EDA. For graphs we will focus on V12 and V13 selected randomly.



V12 appears to have more outliers than V13 and such findings are common across all the V1-V28 columns. Now we will see the distribution of V12 and V13 across Normal and Fraud Transactions.

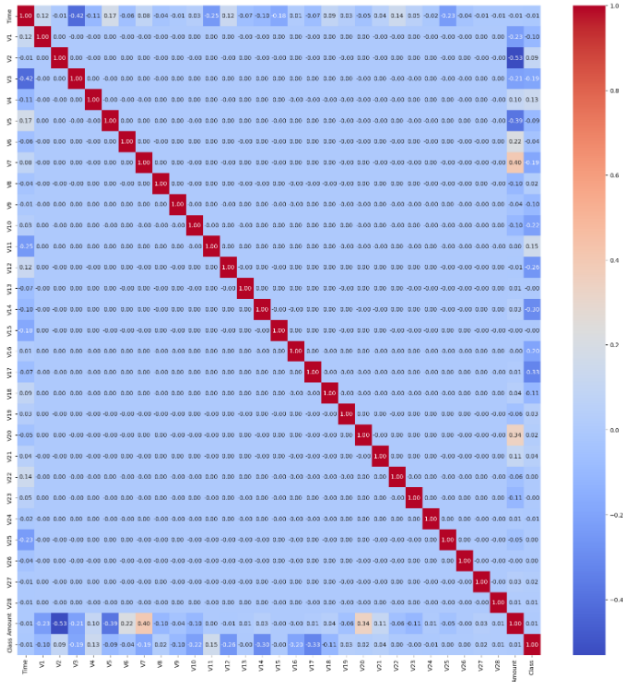




Orange Curve is for Normal Transactions and Blue Curve is for Fraud Transactions. While for V13 both are overlapping a lot for V12 both distributions are even visually distinguishable. And this can be seen across other

PCA columns too. While not all of them have very different distributions for Normal and Fraud Transactions most of them have such differences. This tells us that the Features were selected with proper care by the Machine Learning UBL group.

Now we will check for collinearity as many techniques like Logistic Regression require no collinearity to work properly. To do this we will use a heatmap as given below:



Due to the limited space we are not able to extend the graphic above properly. However still we can observe that there are no high collinearity across all the columns.

Highest Positive Correlation is 0.40 between Amount and V7. And Highest Negative Correlation is -0.53 between V2 and Amount Column. While we can check for multi-collinearity with Kurtosis and other measures too from the Heatmap above we can see that there are no high correlation between the columns in the dataset which can impact the Modeling process and possibly prevent the processing of a model.

With this we have finished our EDA. We covered key findings in this report and full walkthrough of the EDA can be found in the GitHub project page with same name which will be uploaded once project is completed and edited.

3.3 Anomaly Detection Techniques

After the EDA we will move to Step 3 where we applied Different Anomaly Detection Techniques and this part was the biggest part of the project. Following Anomaly Detection Techniques were covered:

1. IQR
2. Modified Z-Score
3. DBSCAN
4. LOF
5. Isolation Forest
6. Gaussian Mixture Model (GMM)
7. Logistic Regression
8. Dense Neural Network
9. Auto-Encoder

While 1-7 techniques above are from the related works we applied two new techniques for the Credit Card Fraud Detection, Gaussian Mixture Model and Auto-Encoders.

IQR, Modified Z-Score, DBSCAN, LOF, Logistic Regression, and Isolation Forest may require Balancing of the Data Set by either upscaling the Fraud Transactions or down sampling the Normal Transactions. However, we also applied these techniques on Training Unbalanced dataset as it is not always assured that the balancing will result in better results. For Deep Learning Techniques we fitted the DNN models to both Balanced and Unbalanced Datasets and trained the Auto-Encoders with all the Normal Transactions.

Gaussian Mixture Model [13] is a type of Unsupervised Clustering Algorithm which is considered far better than basic K-Means algorithm. Here we will model the data as a mixture of different gaussian distributions. Main difference than traditional K-Means is that it is able to cluster while producing probability distribution of each data point for different cluster, this feature allows it to handle complex cluster shapes overall. While it may not be better than Isolation Forest which is designed with the aim of finding Outliers, nevertheless we will apply it and check how much more simpler algorithm will work on the dataset.

We will also apply the Auto-Encoder Deep Learning Technique for the Credit Card Fraud Detection along with the **Focal Loss** which is known for better loss function for the outlier detection as it is able to work better than normal Binary Cross Entropy Loss function in case of Class Unbalance.

Auto-Encoder is a powerful deep learning methodology and it can be used to identify Outliers in Unsupervised learning. In Auto-Encoders we have three parts, Encoder Neural Network whose goal is to compress the input and produce the Bottleneck layer which contains the learned representation and finally Decoder Neural Network which de-compress the Bottleneck layer and give the output similar to the input, and in order to get this we use the Reconstruction Loss. Main goal here is to get the Bottleneck layer to learn the representations of the input data.

This feature of Auto Encoders can be used easily in Unsupervised Anomaly Detection, we will train the model on the Normal Data Points due to which model will be able to learn the representations of the Normal Data Points and in testing if we process Fraud Transaction it will give out a higher reconstruction loss as since it didn’t learn the representations of Outlier Points. Due to this we will be able to identify Outlier Points/Anomalies as they have higher reconstruction loss.

We will use the above Auto-Encoder and other possible Deep Learning Techniques with Focal Loss which is a special type of Loss technique created for the Object Detection Algorithms [14] but can be used for different kinds of Unbalanced Dataset, and since our chosen dataset is highly unbalanced instead of Balancing it for Deep learning techniques, we will use Focal Loss to mitigate this issue.

Focal Loss is based on the Binary Cross Entropy and in this project we will use the Alpha Based Variant which involves two hyperparameters α, Weighing Factor, and γ, Focusing Parameter:

$$Focal\ Loss = -\sum_{i=1}^{i=n} \alpha(i - pi)^{\gamma} Log(pi)$$

Main focus will be on Gamma or γ or Focusing Parameter as with Gamma >= 1 we are able to get a loss function which is able to focus more on difficult examples and will down-weight the easy cases. For instance, consider predicted probabilities to be 0.9 and 0.6. Considering γ = 2, the loss value calculated for 0.9 comes out to be 4.5e-4 and down-weighted by a factor of 100, for 0.6 to be 3.5e-2 down-weighted by a factor of 6.25.

Author of the Focal Loss papers claim that γ = 2 works best as per their experiments with γ = 0 being same as Binary Cross Entropy Loss. For this project we will utilize the γ = 2 and will test other inputs for both hyperparameters.

3.4 Application of Techniques

In this part we will cover the methodologies adopted to apply these Techniques.

We used three Datasets as explained in Section 3.1, namely Training Un-Balanced Dataset, Training Balanced Dataset, and Testing Dataset. We tested all the models specifically on the Testing Dataset as both Training Datasets represented conflicting information. For example, DNN trained on Balanced Dataset had a Recall of more than 0.80 and Un-Balanced Dataset had only Recall of 0.62 but both models on Testin Dataset performed similarly well as we can see from the Results Table in Section 4.2. Due to this we focused only on Testing Data for evaluation.

For training we applied both Training Balanced and Un-Balanced Datasets where it was applicable.

IQR Method and Modified Z Score are only traditional Anomaly Detection algorithms covered in this project. To apply them properly, we classified an observation as Fraud only that observation contains Outlier Data in 15 or more columns out of 30 columns. We will record the Outliers in all the columns depending on respective method and then flag the transactions as fraud if it has 15 or more columns as outliers.

DBSCAN was facing multiple Runtime issues due to RAM overflow, and even if we got it working it didn’t work well on both Training Models, but it gave good results when we applied it on the Test Dataset it performed remarkably well.

Local Outlier Factor also faced issues similar to DBSCAN as it was also not able to classify Frauds properly in both Training datasets. We considered to remove it altogether but decided to apply it on Test Dataset and get better results and included them to show that not all ML based techniques are better than Traditional techniques.

Isolation Forests were applied to both Training Datasets respectively and we got interesting results after evaluation the model with Test Dataset which are discussed separately in Section 4.3.

Gaussian Mixture Models was also trained on both Training Datasets, Training Un-Balanced and Training Balanced. And it also got results similar to Isolation Forest, both warranting special attention in Evaluation.

Logistic Regression Models was also trained with both Datasets. Both of these LR Classifiers were evaluated against Test Dataset.

We made Four **Dense Neural Networks** with similar Architectures. Two are based on Focal Loss, for both Training Datasets. And other Two are based on Binary Cross Entropy Loss Function. All of them were trained on Adam Optimizer with default Learning Rate of 0.001 and similar overall attributes. We evaluated them with Test Dataset

We made two **Auto-Encoders**, one for Focal Loss and other for Mean Squared Error as Reconstruction Loss. We trained both of them using all the Normal Transactions and tested them against Test Dataset representing Real World Credit Card Transactions.

In total we applied 16 different techniques or models and will evaluate their results in following section.

4. Evaluation

After preliminary application of Modeling and various Evaluation Techniques like Accuracy, Recall, F1-Beta Score, and AUC we found that Recall and F1-Beta Score serve the purpose of the project properly.

Accuracy has a clear problem that it give equal weight to the Normal and Fraud Transactions and AUC or Area Under Curve gave an inconsistent result, for some models it was working as intended but in models which were giving higher Recall it was giving us lower AUC.

Due to above issues with Accuracy and AUC we will specifically focus on Recall and F1-Beta Score, with primary focus on Recall.

4.1 Metrics

Recall is essentially a measure of how much a model is able predict correct Positives among all the true positives. It is calculated as the ratio between the number of Positive samples correctly classified as Positive to the total number of Positive samples. In case of unbalanced dataset Recall is much better metric as it focus on the Positives strictly and it is important that we are able to identify Credit Card Frauds properly.

$$Recall = \frac{True\ Positives}{True\ Positives + \ False\ Negatives}$$

F1-Beta Score or F-Beta Score [15] is a modified version of F1 Score which itself is a harmonic means of precision and recall. F1 Score summarize the information of precision and recall in a single metric, and with F1-Beta Score we are able to adjust it to give more weight to the precision and recall as required.

F1-Beta Score Formula:

$$\frac{((1 + beta^2) * Precision * Recall)}{beta^2 * Precision + Recall}$$

The essence of this formula is that Beta < 1 gives more weight to the Precision and Beta > 1 will give more weight to the Recall. For this project since we need more focus on Recall we will use Beta > 1.

For this project we will utilize Beta = 2, giving twice the weight to the Recall than Precision, with threshold >0.5 where needed.

4.2 Evaluation Process

After tidying the data we constructed three Datasets, namely Training Unbalanced, Training Balanced (Borderline SMOTE) and Testing Dataset. Training Unbalanced and Testing Dataset are part of raw scaled with Training Dataset having 80% of the samples with both Datasets having same ratio of Fraud Transactions. Training Balanced was created to test if the Balancing technique like Oversampling via Borderline SMOTE helps in performance of the model.

We ran the models or techniques natively on a RTX 4060 Laptop 8GB GPU and we analyzed the different model and techniques based on the Recall and F Beta Score metrics discussed in Section 4.1. For this evaluation we focused only on Test Dataset metrics as due to using Oversampling only common way to test the models was to evaluate them with the real world Test Dataset which represent small proportion of Fraud Transactions as we see in reality.

We also tuned the hyperparameters as required and in the end, we obtained results for different models which are discussed below.

4.3 Evaluation Result

While we covered Nine Different Anomaly Detection Techniques we trained some of the techniques to both Training Balanced Dataset and Training Un-Balanced Dataset leading to development of 16 Different models.

Results were stored in a Pandas Dataframe which has been sorted by the Recall, and is given below as a table:

Model/Technique	**Recall**	**F Beta Score**
Auto-Encoder Focal Loss	1.000	0.008
Auto-Encoder Mean Squared Error	0.990	0.008
Isolation Forest Unbalanced	0.898	0.072
Gaussian Mixture Model Unbalanced	0.898	0.020
DBSCAN	0.888	0.062
Logistic Regression SMOTE (Balanced)	0.867	0.321
DNN SMOTE Focal Loss	0.847	0.788
DNN SMOTE Binary Cross Entropy	0.796	0.788
DNN Unbalanced Binary Cross Entropy	0.786	0.785
DNN Unbalanced Focal Loss	0.776	0.788
Logistic Regression Unbalanced	0.561	0.604
IQR Method	0.418	0.381
Modified Z Score	0.388	0.377
Isolation Forest SMOTE	0.378	0.142
Local Outlier Factor	0.265	0.021
Gaussian Mixture Model SMOTE Balanced	0.010	0.013

Best performing model is Auto-Encoder with Focal Loss if we consider specifically Recall. And if we consider both Recall and F Beta Score then best model is Dense Neural Network with Balanced Data and Focal Loss.

While Auto-Encoders were able to flag almost all the Fraud Transactions properly, they were also flagging most of the transactions as Fraud too which will be impractical in real world application as it will only increase workload. In this regards DNN trained on Balanced Data with Focal Loss appears to be best technique.

There were other good models too like other Dense Neural Networks with Balanced Data which were trained in half the time of DNN SMOTE Focal Loss but still fives appropriate results.

And if we focus on the Traditional Techniques IQR Method works the best with a Recall of 0.418 and F Beta Score of 0.381. While these numbers are less than DNN, Logistic Regression, and Auto-Encoders they were not the weakest models by far. I

It was assumed that all the ML Based models or DL models will work better than Traditional Techniques but worst model in the project was Gaussian Mixture Model Trained on Balanced Dataset, which had a paltry recall of 0.010. Isolation Forest too with Balanced Dataset got a low result of 0.378. And we also have LOF with the 0.265 recall, highest result of it across all three Datasets.

All these results reflects three important findings:

1. More complex the model or technique leads to better results overall but this is not always true as LOF being more complex than IQR performs worse than it.
2. Balanced Datasets can help us to create models which can learn representations better than Unbalanced Datasets but this is not always true. Isolation Forest has a high recall of 0.898 in Unbalanced Datasets because it is able to identify outliers specifically. By balancing the dataset we essentially generalized those outliers leading to worse performance for iForest. Same happened to Gaussian Mixture Model which also had 0.898 Recall.
3. Even simple Dense Neural Networks are able to perform better than Logistic Regression. Implying that Deep Learning based Anomaly Detection techniques have a clear edge over ML based techniques but they will also take more time to train, which is an important factor of consideration too.

5. Discussion

Credit Card Fraud Detection Project is based on the PACE framework as mentioned in Section 3.1. This allows us to chart out properly detailed Milestones as follows:

Milestone	**Tasks**	**Status/Estimated Time**
Milestone 1- Planning	• Project Workflow is outlined. • Data obtained. • Anomaly Detection Techniques identified	Completed
Milestone 2- EDA	• Data is scrubbed, converted, and tidied up. • Perform comprehensive EDA analysis. • Update Project Slides and Report	Completed
Milestone 3 – Anomaly Detection	• Application of Nine Anomaly Detection Techniques listed in Section 3 • All techniques are tested and retested for accuracy while as required performing hyperparameter tuning. • Updating the Project Slides and Report	1-2 Days(In Progress)
Milestone 4- Evaluation and Conclusion	• Evaluation Metrics are gathered and subjected to a proper Evaluation Process. • Forming final Conclusion • Finalizing the Project Slides and Report • Finalizing the Presentation Video	1-2 Days

This is a tentative milestone chart and estimates can vary on the basis of time allocated to the project. The milestone and estimates may change as the Project progresses. As of writing this Project we are doing EDA step and after it is completed, we will move to next stage of applying various Anomaly Detection techniques to identify Fraud transactions or learn the representation in the dataset.

5.1 Potential Challenges and Backups

While all the Anomaly Detection techniques were chosen with careful consideration alongside the Dataset, we still can face following potential challenges:

1. Anomaly Detection Techniques chosen not compatible with the Dataset.
2. The dataset is not fit for the project or is lacking in other regards.
3. Unforeseen difficulties in procuring the Resources required for the project.

There can be many more Potential Challenges we could document but these three seems to be most relevant after careful consideration as they can influence the Scope and Methodologies of the project.

We have also have backup plans for them:

1. After EDA we will once again analyze the application of the anomaly detection techniques and check if they are applicable to the dataset, and if any change in techniques is required. We will update about this in future updates of this report.
2. This Credit Card Fraud Dataset is based on the real-world transactions and was selected after checking other alternatives. But still if we face any problems during the EDA process we will pick the Simulated Fraud Dataset from Kaggle or other reputed sites.
3. We will perform this project locally on a Laptop powered with CUDA Enabled RTX 4060 8 GB

Laptop GPU with TensorFlow 2.13/2.14 library and other Python libraries required for the task. We will run the project on the WSL framework for Windows 11 and we have three alternatives if it does not work due to some technical issues. First, we can run the TensorFlow 2.10 natively on Windows 11. Second, we can use RTX 3070 8 GB Desktop GPU with WSL on Windows 11 if RTX 4060 Laptop GPU is not available. And finally, we can run the whole project on the dedicated GPU Kaggle or Google Collab provides, P-100 GPU or other similar GPU.

5.2 Changes/Lessons

After working on the project till previous checkpoint we made following changes:

1. More clear overall direction with cleaner Executive Summary.
2. Indepth knowledge about the various Predictors of the Credit Card Fraud Detection Dataset and application of Preliminary Modeling gave us the glimpse of the limitations of the Environment in which project will be conducted.
3. Deciding on Recall and F-Beta Score as only metrics for Evaluation after confirming the limitations of Accuracy and AUC metric.

These changes were important as they guided the project in a better direction. Cleaner Abstract in form of Executive Summary helped in forming a clear vision for the overall project. Proper EDA helped in order to understand which type of Anomaly Detection Techniques will really apply, we even considered more techniques like T-SNE which failed to fit properly on the dataset. And finally, deciding on Recall and F-Beta Score provided a much clear and compact evaluation process which was to the point and allowed comparison across different types of techniques.

However, there were many things that didn’t work too, like T-SNE and other techniques. In Auto-Encoders Focal Loss, which was the best Model in this project, parameters we decided upon earlier didn’t work at all. And DBSCAN was overloading RAM in both local and Cloud Environment like Kaggle.

We could have created a better overall project with much better dataset, more Computing Power, and more techniques. But in the end it was a well implemented project as it was designed to run most of the Environments people can access nowadays and it has achieved the purpose it was designed for.

6. Conclusion

We performed a complete Knowledge Discovery in Database or KDD process in this Datamining Project. We moved from Data Procurement, to Preprocessing, to Modeling, to Evaluation, and finally to the current conclusion in a smooth flow.

By using KDD process and PACE Framework we were able to apply 16 different types of Anomaly Detection Techniques on Credit Card Fraud Detection and were able to show that due to increasing focus on AI technologies not only Credit Card Frauds are getting sophisticated but Fraud Detection is also moving from Traditional Techniques to high end state of the art techniques, and even easily available Hardware environments can run them.

We tested 16 models in Evaluation where Auto-Encoder with Focal Loss appears to be best Model if our goal is only to detect Credit Card Fraud Transactions as it was able to identify all 98 Fraud Transactions properly in Test Dataset with Threshold = 3. But it had its own limitations too, it had much weaker Precision leading to abysmal F-Beta Score of 0.008 when we even increased weight of Recall to 2.

If a middle ground is needed we suggest Dense Neural Network Trained on Training Balanced Dataset with Focal Loss as it has good recall of 0.867 and highest F beta Score of 0.788. Of course, it will lead to missing around 13% of the Fraud Transactions which may be a bigger blow to Banking Companies and Customers than 0.008 F-Beta Score.

If only Speed is concerned best model considering it was Gaussian Mixture Model Trained on Training Un-Balanced Dataset with Recall of 0.898 and F Beta Score of 0.072 as it was fastest model in training besides much faster Traditional techniques.

Still, in this project we also saw that more complex techniques are not always better with even IQR able to beat LOF and other ML Based Techniques.

Overall, we started this model to apply various Anomaly Detection Techniques on the Credit Card Fraud Dataset in a Local Environment or a Cloud based free Environment to show the power of new Anomaly Detection Techniques and their utilizations. All of the techniques we covered are usable in different scenarios but if main objective of a Banking Company is detect as much as Fraud Transactions it can even at the cost of fall in Precision to very low level Autoencoders can work well. But this will lead to too much work for the company thus we can even increase the Precision with different Threshold values and try different loss functions and architecture in Auto-Encoders.

There are many limitations in this project. We could have tried to get bigger real life dataset even at a cost by pooling resources with other peers. We could have explored even more Deep Learning Techniques and check if they work for Anomaly Detection. Our best working model, Auto-Encoder, was given a lot of time, but we could have explored better Architectures for Auto-Encoders, explored more Loss functions and different optimizers.

Due to Hardware Limitation, namely limited RAM and VRAM, we were not able to run more sophisticated Architectures in DNN and Auto-Encoders, with better resources we can Experiment different architectures and even more resource intensive techniques.

This project was applied to One month transactions in One year a decade ago. So the scale of project is much smaller than current real world transactions across the globe. DNN on Balanced Dataset with Focal Loss will work best in such setting but many of the techniques may require substantial changes, especially Auto-Encoders which is reaching Recall of 1.0 by flagging most transactions as Fraud.

7. References

[1] Types of Credit Card Frauds - https://www.bajajfinserv.in/insurance/types-of-credit-card-fraud-and-how-you-can-avoid-them

[2] Anomaly Detection Wiki- https://en.wikipedia.org/wiki/Anomaly_detection

[3] Victims of credit card fraud tell their stories - https://www.computerworld.com/article/2591492/victims-of-credit-card-fraud-tell-their-stories.html#:~:text=She%20had%20been%20writing%20checks,small%20purchase%20at%20a%20drugstore.

[4] Identity Theft and Credit Card Fraud Statistics for 2023- https://www.fool.com/the-ascent/research/identity-theft-credit-card-fraud-statistics/

[5] How Does the Chip in My Credit Card Work? https://www.fool.com/the-ascent/credit-cards/how-chip-works-in-credit-card/

[6] Credit Card Fraud Detection- https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud/data

[7] Anomaly & Fraud detection https://towardsdatascience.com/anomaly-fraud-detection-a-quick-overview-28641ec49ec1

https://towardsdatascience.com/anomaly-fraud-detection-a-quick-overview-28641ec49ec1

[8] V. Ceronmani Sharmila, K. K. R., S. R., S. D. and H. R., "Credit Card Fraud Detection Using Anomaly Techniques," 2019 1st International Conference on Innovations in Information and Communication Technology (ICIICT), Chennai, India, 2019, pp. 1-6, doi: 10.1109/ICIICT1.2019.8741421.https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8741421

[9] Hoaglin, David C.. “Volume 16: How to Detect and Handle Outliers.” (2013). https://www.semanticscholar.org/paper/Volume-16%3A-How-to-Detect-and-Handle-Outliers-Hoaglin/d524a172b49e25f888376d662ee364aa77d99e8a

[10] Modified z score- https://www.ibm.com/docs/en/cognos-analytics/11.1.0?topic=terms-modified-z-score

[11] https://www.dbs.ifi.lmu.de/Publikationen/Papers/KDD-96.final.frame.pdf

[12] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu “A Density-Based Algorithm for Discovering Clusters

in Large Spatial Databases with Noise” KDD-1996- https://cs.nju.edu.cn/zhouzh/zhouzh.files/publication/icdm08b.pdf

[13] Unsupervised Learning: Clustering using Gaussian Mixture Model (GMM) https://behesht.medium.com/unsupervised-learning-clustering-using-gaussian-mixture-model-gmm-c788b280932b#:~:text=Clustering%20is%20a%20fundamental%20task,a%20mixture%20of%20Gaussian%20distributions.

[14] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár “Focal Loss for Dense Object Detection” 2018 (v2)- https://arxiv.org/abs/1708.02002v2

[15] A Gentle Introduction to the Fbeta-Measure for Machine Learning- https://machinelearningmastery.com/fbeta-measure-for-machine-learning/

[16] LOF Scipy https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.LocalOutlierFactor.html

[17] Handling Imbalanced Data by Oversampling with SMOTE and its Variants https://medium.com/analytics-vidhya/handling-imbalanced-data-by-oversampling-with-smote-and-its-variants-23a4bf188eaf