Relevance of Personal Information in Credit Card Fraud Detection: An Ablation Study

Jason Lebov
Computer Science
Emory University
Atlanta, GA
jason.lebov@emory.edu

Aarush Bedi
Computer Science & QSS
Emory University
Atlanta, GA
aarush.bedi@emory.edu

ABSTRACT

This ablation study investigates the significance of cardholder personal information *(PI)* in credit card fraud detection through binary classification analysis. Utilizing the Kaggle datasets 'Credit Card Fraud Prediction', comprising 22 distinct features related to specific transactions and cardholder PI, extensive data cleaning and preprocessing were performed. This preprocessed data was then broken up into several subsets with varying degrees of PI and were then fed into two different machine learning models, namely Decision Tree and XGBoost, to evaluate the importance of PI. While prior research in this domain has mainly focused on model evaluation and performance, this study prioritizes the impact of the input data on model efficacy. Understanding the role of PI in credit card fraud classification is crucial in today's society, particularly as people continue to be concerned with the amount of their personal information that is easily accessible and utilized by large corporations. Given that the current methodology for detecting credit card fraud requires data on specific transactions, the merchant involved, as well as personal cardholder details, this study aims to highlight the relevance of cardholder personal information, offering insight into the potential development of more privacy-conscious fraud detection algorithms. The results of this study show that personal cardholder information *(PI)* has insignificant relevance in terms of fraud classification, and therefore, could be considered for removal in future detection implementations.

Keywords: Ablation Study, Binary Classification. Credit Card Transactions, Personal Information, Fraud, Decision Tree, XGBoost

I. INTRODUCTION

Credit card fraud is a persistent and continuously growing global problem that has impacted people worldwide. With projected financial losses reaching around $397.4 billion over the next decade, with $165.1 billion attributed to the United States alone[1], this is an important area that has demanded extensive attention and innovative solutions. Historically, research has been conducted primarily focusing on detection methodologies, specifically the efficacy of numerous models such as Linear Regression, K-Nearest Neighbors, or Hierarchical Behavior-Knowledge Space Models.[2] However, the reliance on specific data for effective model creation has raised concerns, particularly regarding the need for personal cardholder information.

Recently, the over-sharing of people's personal data with large entities, such as financial institutions, has become a major concern, especially regarding the lack of transparency from these

[1] Egan, John. "Credit Card Fraud Statistics." *Bankrate*, www.bankrate.com/finance/credit-cards/credit-card-fraud-statistics/. Accessed 3 May 2024.

[2] Nandi, Asoke K, et al. "Credit Card Fraud Detection Using a Hierarchical Behavior-Knowledge Space Model." *National Library of Medicine*, 20 Jan. 2020, www.ncbi.nlm.nih.gov/pmc/articles/PMC8775357/.

larger corporations. According to a study conducted by the Pew Research Center in 2023, 81% of US adults are concerned with how companies use the data they collect from them, with 67% having little to no understanding of how this data is actually used[3].

This ablation study - a study involving the removal of individual features from modeling and measuring the resulting changes in output - aims to address these concerns by investigating the relevance of a cardholder's personal information in accurately detecting fraudulent transactions while maintaining comparable accuracy levels of models trained on traditional data inputs. Specifically, this study looks to answer key questions: Is a cardholder's personal information needed to accurately determine if a transaction is fraudulent? Which factors are the most influential?

Overall, this study looks to shed light on the intersection between accurate fraud detection and greater data privacy concerns.

II. RELATED WORK

Prior work regarding credit card fraud detection has been conducted, however, most of these studies focus on the effectiveness and performance of different machine learning models rather than looking at the data itself. For example, a 2023 literature review looked into over 181 different articles from 2019 to 2021 regarding fraud detection, each evaluating different machine learning and deep learning techniques for this task[4]. This review concluded that given most ML techniques are effective in detecting this type of fraudulent behavior, a deeper analysis should be conducted to determine other ways to evolve current practices. A different study also looked into fraud detection focusing specifically on logistic regression, random forest, and XGBoost as their models of choice[5]. Similarly, their results showed comparable effectiveness across all models, however, they determined that the use of random forest in combination with a hybrid oversampling technique called SMOTE and an undersampling technique called Tomek Links, performed better than the other models.

Given both preexisting studies, along with many others into credit card fraud detection, it is evident that current research has mainly focused on model performance and comparisons across models. Therefore, as addressed by both papers mentioned above, there is a need for further research and analysis into specific aspects of this classification problem. Given this, our study aims to address the data that the models are being trained and tested on, rather than solely the model implementation.

This idea served as the basis for our ablation study where we focused on only a few machine learning models while creating numerous datasets to be used for both training and evaluation to find data-specific insights. Our work hopes to shed light on the data aspect of fraud detection by looking into the relevance of personal cardholder information in classifying fraudulent transactions.

III. DESCRIPTION OF DATA

The dataset used in this study was sourced from 'Kaggle.com' and is called the 'Credit Card Fraud Prediction' dataset. It consists of 22 features and exactly 555,719 samples with no missing or null values. The features provide relevant information on credit card transactions, merchant information, and personal cardholder information. Moreover, it includes a binary classification variable called

[3] McClain, Colleen. "How Americans View Data Privacy." Pew Research Center, Pew Research Center, 18 Oct. 2023, www.pewresearch.org/internet/2023/10/18/how-americans-view-data-privacy/.

[4] Btoush, Eyad Abdel Latif Marazqah, et al. "A Systematic Review of Literature on Credit Card Cyber Fraud Detection Using Machine and Deep Learning." *National Library of Medicine*, 17 Apr. 2023, www.ncbi.nlm.nih.gov/pmc/articles/PMC10280638/.

[5] Shakya, Ronish. *Application of Machine Learning Techniques in Credit Card ...*, digitalscholarship.unlv.edu/cgi/viewcontent.cgi?article=4457&context=thesesdissertations. Accessed 28 Apr. 2024.

'*is_fraud'* that indicates whether a specific transaction is fraudulent or not. This variable served as the target feature for this analysis.

For the scope of this ablation study, the features that were classified as cardholder personal information *(PI)* are as follows: Year Born, Name (first & last), Address (lat & long), Gender, and Job. These features were isolated accordingly as we constructed seven different subsets of the original data to better understand how each PI feature impacts model efficacy and performance, and to determine if PI is needed for accurate detection *[See Fig 5]*.

IV. METHODOLOGY & SYSTEM DESIGN

Given the goal of this study was to identify data-related patterns, specifically the importance of cardholder person information, the majority of our overall approach and system design was based on the preprocessing and manipulation of the original dataset. These steps resulted in the creation of seven subsets of the original data, all with ranging levels of PI. These datasets were then fed into our two chosen models, Decision Tree and XGBoost, which were hyperparameter-tuned using GridSearchCV on several predefined parameter grids. Our study utilized pre-existing packages and functions from Pandas, Numpy, and Scikit-Learn to facilitate our analysis.

A. Data Exploration

Before beginning the preprocessing step of our study, we began with a basic exploration of the dataset to better understand its contents. This step was crucial as we planned our preprocessing steps to ensure we had the data in the correct format for modeling and evaluation.

From this step, we determined that we needed to perform several preprocessing techniques on our data to convert all values to numeric. Additionally, we found many features that we deemed to be irrelevant that were simply dropped from our dataset as well as features we felt could be extrapolated to obtain more information from. Lastly, we determined that our target feature was extremely unbalanced with 553,574 samples labeled as 'not fraud' *(0)* and 2,145 samples labeled as 'fraud' *(1)*, as seen in *Fig 1*. This meant that less than 1% of our samples were positively labeled, giving us a very small subset of values for our model to be trained and tested on.

All of these steps, along with their specifics, will be further expanded upon throughout the remainder of the paper.

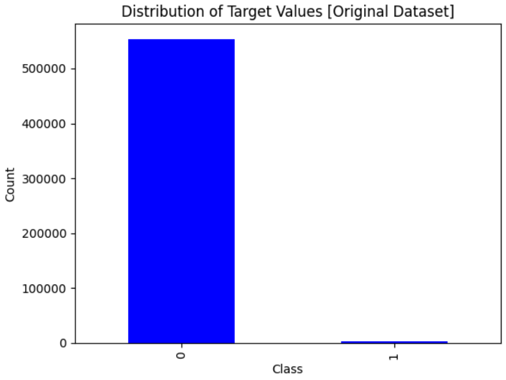


Fig 1. Target Feature Class Imbalance Visualized

B. Data Pre-Processing

After performing basic data exploration we determined our optimal preprocessing plan which included feature cleaning, feature extraction, feature selection, dataset creation, and lastly feature scaling.

***B.i)* Feature Cleaning:** The feature cleaning aspect of preprocessing mainly consisted of dropping any features we determined to be redundant or irrelevant to our study. These features were, 'street', 'city', 'state', 'zip', 'unix_time', 'trans_num', and 'cc_num'. The reason 'street', 'city', 'state', and 'zip' were dropped was because these values were for the address of the cardholder, but the dataset also had this same address in longitude and latitude which was better for our study given it helped reduce dimensionality and was already numeric. 'Unix_time', which was the time of a credit card transaction, was also dropped because it was the same as another feature named 'trans_date_trans_time'. Lastly, we dropped 'trans_num' and 'cc_num' because they were unique transaction and card identifiers, so every sample had a unique value and therefore it added no value to the study. After dropping all of these columns the data was left with 15 features.

B.ii) Feature Extraction: For feature extraction, several steps were performed on the original data to transform the data into a workable form. Initially, multiple features were extracted from existing features in the dataset. For example, the feature 'year born' was derived from the date of birth *('dob')* attribute. Additionally, features were extracted from the 'trans_date_trans_time' column including 'Day', 'Month', 'Year', and 'Time of Day'. 'Time of Day' categorized transactions into Morning *(00:00-09:59)*, Afternoon *(10:00-16:59)*, and Evening *(17:00-23:59)*, providing a more insightful representation than continuous numerical timestamps. Furthermore, we used One-Hot Encoder and Label Encoder to convert all categorical variables in the dataset to numeric. One-Hot Encoder was used for features with lower cardinality, while Label Encoder was used for features with higher cardinality to keep the dimensionality of the dataset within a reasonable range. The final step in feature extraction involved vectorizing any remaining categorical variables that could not be encoded using One-Hot Encoder or Label Encoder. Specifically, this applied to the cardholder name attributes which were in the form of 'first and 'last', separately. To accomplish this, a HuggingFace package named FastText was used on both first and last names respectively, converting these values into numeric representations based on their root of origin. This step aimed to preserve any semantic similarities among the names, which could potentially have an impact on the model's predictive power.

B.iii) Feature Selection: For this step of preprocessing, a Pearson Correlation Matrix was used, as seen in *Fig 2*, to determine any features that were highly correlated to each other or any features that were highly uncorrelated to the target features. Any features that broke the determined Gamma *(feature-to-feature)* or Delta *(feature-to-target)* thresholds accordingly, were dropped. This step resulted in the dropping of two features, 'merch_lat' and 'merch_long', which are both variables that gave the address of the merchant involved in a given transaction.

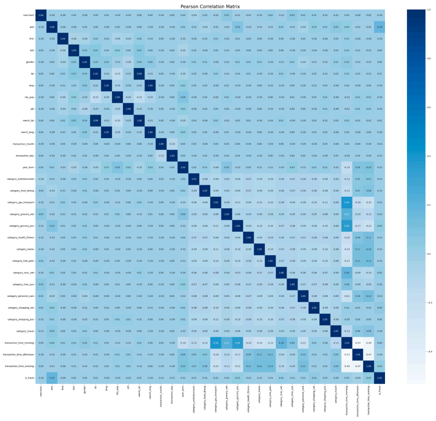


Fig 2. Full Dataset Pearson Correlation Matrix

B.iv) Dataset Creation: After fully preprocessing the data using the above steps, we were left with what we call our 'Base Dataset'. This data consisted of fully engineered attributes relating to credit card transactions and had information regarding the transaction, merchant, and cardholder details. Using this information, seven datasets were created from this 'Base Dataset' that would work as the main part of our ablation study and modeling phase. Given there are seven features in the data that are considered PI - Year Born, First and Last Name, Address (lat & long), Gender, and Job - each was broken up into five different datasets. Each of these datasets comprised all Non-PI attributes, plus one of the above PI features. This resulted in the seven distinct datasets that were used throughout the remainder of the study: Full Dataset w/ PI, Full No PI Dataset, No PI Dataset w/ Year Born, No PI Dataset w/ Name (first & last), No PI Dataset w/ Address (lat & long), No PI Dataset w/ Gender, and No PI Dataset w/ Job. Each of these datasets had the same number of samples but differed in the number of features.

Lastly, as mentioned above, the dataset used in this study suffered from a severe class imbalance problem which had to be addressed. Given this, an oversampling technique called SMOTE was employed on the training datasets, for each of the seven sub-datasets. This technique generates artificial samples of the minority class, creating a uniform distribution across all classes. As seen in *Fig 3*, after the application of SMOTE, all datasets (even though only two are shown) have an even distribution of both positive and negative cases.

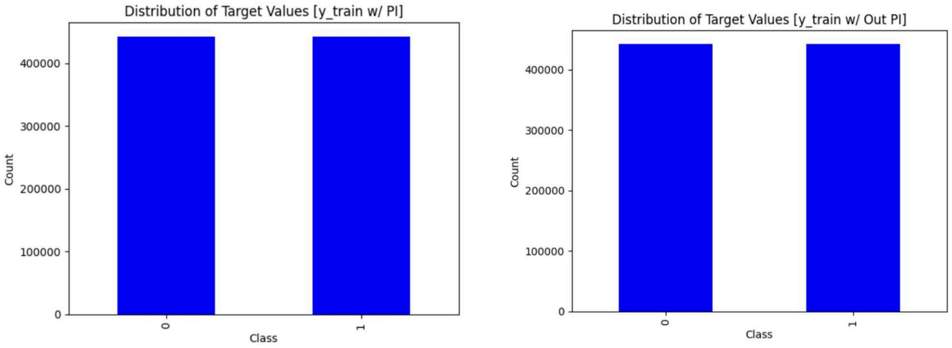


Fig 3. Full PI and No PI Datasets after SMOTE Application

B.v) Feature Scaling: Finally, the training datasets for all seven datasets created above were scaled using Min-Max Scaling. Given all values in these datasets are numeric post-preprocessing, the values were scaled to a range of 0-1. This step was crucial as it ensured that all values in the dataset were uniform and normalized regardless of magnitude or units.

C. Modeling

After the data was fully preprocessed, the modeling phase of the study began. This section consisted of three main parts - model selection, hyperparameter tuning, and model creation - for both of our selected models, across all seven datasets.

C.i) Model Selection: For model selection, we first created a simple rubric as to what we needed to prioritize when it came to choosing our optimal models. The main requirements we felt were necessary were the following - we needed the model to be fast at training due to the large size of the dataset (~800,000 samples post-SMOTE) and the model needed to be able to handle class imbalance. This criteria led to us choosing Decision Trees and XGBoost.

Decision Tree models work by recursively selecting the feature that best splits the data into more homogeneous subsets, using different metrics such as information gain and Gini impurity. After determining all the splits, the final result is a tree-like model that leads a sample of data to its predicted class based on the model's/tree's split values.

XGBoost, on the other hand, is a gradient-boosting framework that sequentially adds weak trees to the ensemble of models, with each new tree predicting the residuals or errors of the previous trees. This approach uses a more regularized model formalization to help with overfitting. The final model is an aggregation of all these weaker trees together, creating a powerful ensemble model. Additionally, XGBoost is strong at identifying differences in feature importance which is valuable for this ablation study.

C.ii) Hyperparameter Tuning: For both of these chosen models, hyperparameter tuning was performed using GridSearchCV from scikit-learn. This was done on the 'Base Dataset' (containing all information) for both the Decision Trees and XGBoost models. We decided to use the optimal hyperparameters that we got from GridSearchCV on the 'Base Dataset' for all the different implemented models (for each of the seven datasets). This would act as the control, or basis, given every model would then be built using the same parameters. This was done to ensure that the only major difference between each model, for Decision Tree and XGBoost respectively, would be the dataset that they were trained on - allowing us to isolate the effect of the datasets/features.

We used the following Parameter Grid for the Decision Tree Models

- Max Depth: a range between 5 - 100, with step size 5 | Optimal = 75
- Min Samples Split: [2, 10, 50, 100, 500] | Optimal = 2
- Min Samples Leaf: [1, 5, 10, 50, 100] | Optimal = 1
- Max Features: ['sqrt', 'log2'] | Optimal = 'sqrt'

We used the following Parameter Grid for the XGBoost Models

- Max Depth: [3, 4, 5] | Optimal = 5
- Learning Rate: [0.01, 0.1, 0.2] | Optimal = 0.2
- N Estimators: [100, 200, 300] | Optimal = 300
- Subsamples: [0.7, 0.8, 0.9] | Optimal = 0.9

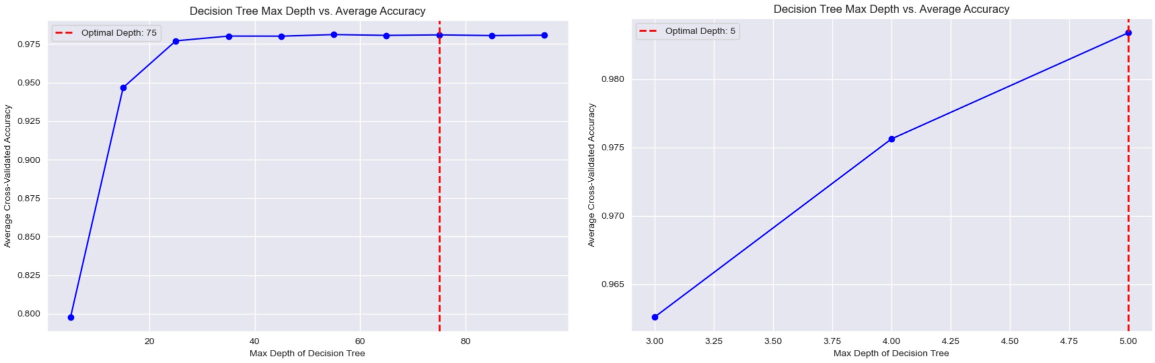


Fig 4. Max Depth vs Average Accuracy from GridSearchCV for Decision Tree & XGBoost

C.iii) Model Creation: After finding the optimal hyperparameters for both of the models, we needed to create the models to perform the analysis. We utilized the DecisionTreeClassifier class from Scikit-learn and the XGBClassifier class from the XGBoost package to build out these models in Python. Given the main goal of this ablation study is to understand and compare the differences in the effects of personal information *(PI)*, we used the seven different datasets created during preprocessing to create seven different models for both Decision Tree and XGBoost (14 models in total). For example, for the dataset No PI w/ Gender, both a Decision Tree and XGboost model were trained and tested solely on this dataset, and similarly for the rest of the datasets.

The reason we chose to do this was because we wanted to understand the different effects each personal feature would have on the model's ability to detect fraud. By creating different models with the only major differing factor being the data used, we could isolate the effect of each feature and understand if personal information is truly required or not.

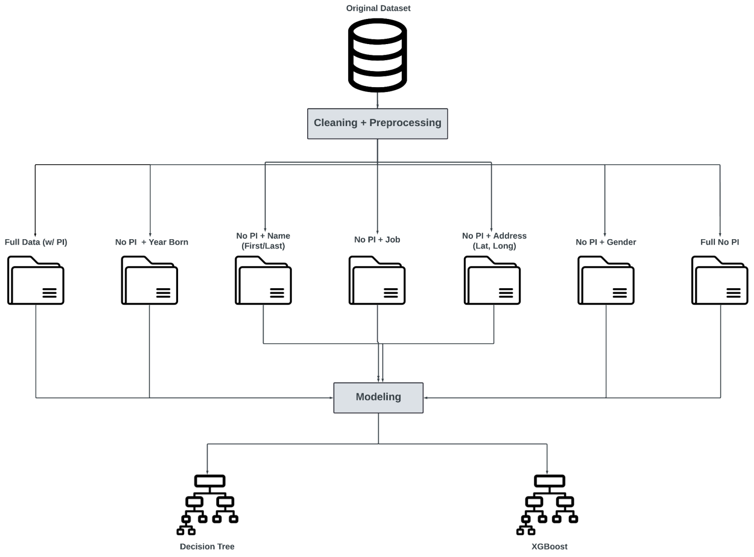


Fig 5. Overall System Design for Dataset Creation & Modeling

V. RESULTS

The work of this study resulted in interesting outcomes from both implemented models, offering valuable insight into our research questions.

A. Model Performance

- Evaluation Metrics -

To evaluate the performance of our models we decided to focus on metrics that are robust to class imbalance, given our dataset suffered from this. These metrics included: AUC and AUPRC. AUC, or Area Under the Curve, refers to the area under the ROC curve and measures the performance of a binary classification model. This value is also useful for plotting the ROC curve and understanding how a model performs - the higher the AUC, the better the performance. Similarly, AUPRC, or the Area Under the Precision-Recall Curve, is also a metric used to understand the effectiveness of classification models but is particularly useful in situations where there is significant class imbalance.

Although we still looked at the model's accuracy scores, given the dataset is extremely imbalanced across all models and datasets, the results were relatively similar with values all around 99%. Therefore, providing little insight into our analysis.

A.i) Decision Trees: The seven implemented Decision Tree models, although having poor overall classification results, led to interesting observations as we looked into the differences between each model/dataset with varying levels of PI. The AUC values for all models were in the range of 0.85 ~ 0.89, indicating a good discriminative ability between target classes. The highest observed AUC was for the model trained on the 'No PI w/ Year Born' dataset, which had a score of 0.89418. whereas the lowest AUC score was the model trained on the ‘No PI w/ Jobs’ dataset with a score of 0.85928. The remaining values for all models can be seen more clearly in *Table 1* and *Fig 6*.

In terms of AUPRC, we saw that Decision Trees in general had low values for each of the different models/datasets. The highest observed value was 0.37881 which was seen with the ‘Full w/ PI model’ and the lowest value was 0.17552 from the ‘No PI w/ Lat and Long’ dataset.

Looking into our research questions, specifically whether personal information is required for classification, our results indicated that although there are minor discrepancies across different datasets, overall results were similar across the board. As seen in *Table 1* and *Fig 6*, the model trained on the dataset with all the information (Full w/ PI) outperformed the model based on the dataset without any personal information (No PI), and this is very evident in the AUPRC values (0.37881 > 0.29601). This would suggest that having personal information would be beneficial for the overall classification, however further analysis of feature importance added deeper insight into this observation. Additionally, it should also be noted that the hyperparameters used across all models were determined based on the Full w/ PI dataset which could be responsible for the current results.

	AUC	AUPRC	Accuracy	Time
Full w/ PI	0.86371	0.37881	0.99635	2.63583
No PI w/ Lat and Long	0.87651	0.17552	0.98924	2.40021
No PI w/ Year Born	0.89418	0.29943	0.99416	1.66227
No PI w/ Gender	0.86216	0.24155	0.99327	1.67402
No PI w/ Jobs	0.85928	0.21269	0.99219	1.97852
No PI w/ First and Last name	0.87629	0.20508	0.99114	2.04177
No PI	0.85948	0.29601	0.99491	1.48723

Table 1. Model Evaluation for Decision Tree

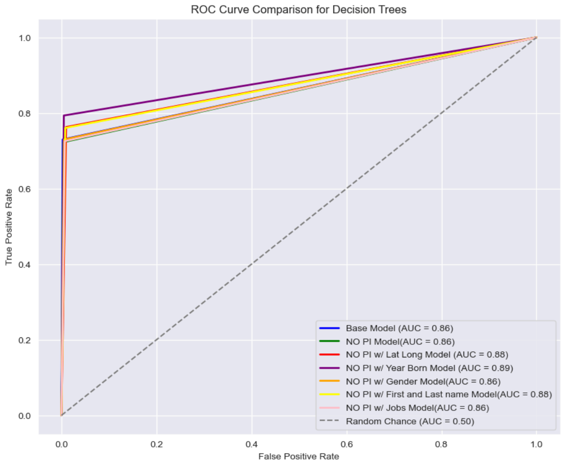


Fig 6. ROC curves for Decision Trees Models

Furthermore, to better understand how the model classified these samples, we can look at the confusion matrix for how the model made its predictions, as seen in *Fig 7*. We can see that the class imbalance between positive and negative classes is so extreme that the correctly identified ‘no fraud’ cases are over 300x the number of correctly identified ‘yes fraud’ classes. This shows that the model was overly sensitive to ‘no fraud’ samples in general. In terms of overall accuracy, it is evident the Full w/ PI model does get more samples correctly classified compared to the No PI dataset model, but only marginally.

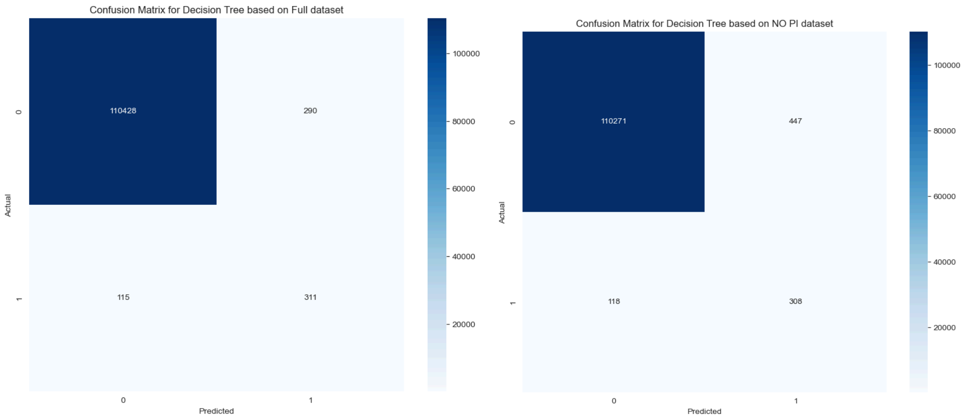
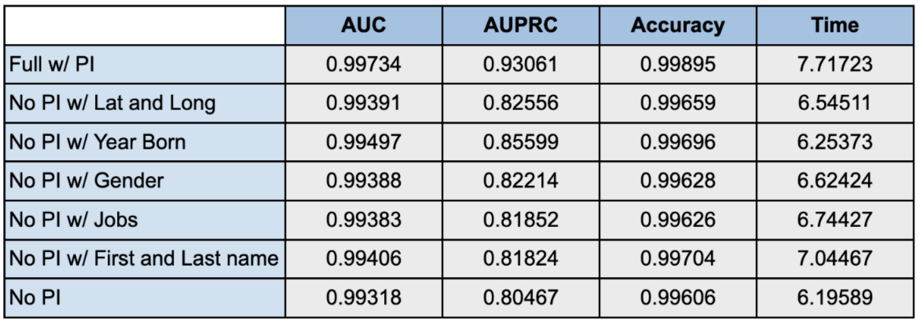


Fig 7. Confusion matrix for Decision Tree Models - Full Dataset vs No PI Dataset

A.ii) XGBoost Models: Similar results were seen for the implemented XGBoost models, except the AUPRC values were much higher. We can see that the AUC values for all models are above 0.99, indicating that this model has almost a perfect discriminative ability. This was expected as XGBoost is a culmination of multiple weaker trees, leading to a more precise overall model. The highest AUC value was 0.99734, which was observed with the 'Full w/ PI' dataset, whereas the lowest value was 0.99318, which was the model built on the 'No PI' dataset. Although the Full PI model did outperform the rest, it is important to note that anything above 0.99 is considered extremely good - which is seen across every model. The visualized ROC curve for all XGBoost models can be seen in *Fig 8*, although, given all values are > 0.9, it is hard to discern which model performs the best. For AUPRC, all models have a very large range of values, 0.80 ~ 0.93. This is rather interesting as six models are in the range of 0.8 ~ 0.85 and the only model having a value above 0.9 is the Full w/ PI model. Again, it should be mentioned that all XGBoost models were also hyperparameter-tuned on the Full w/ PI dataset, so these results may be slightly biased and not fully optimal. Additionally, it is important to reiterate that even though there is a varying range of values, all are relatively high, indicating that XGBoost overall performs well on this classification task.



	AUC	AUPRC	Accuracy	Time
Full w/ PI	0.99734	0.93061	0.99895	7.71723
No PI w/ Lat and Long	0.99391	0.82556	0.99659	6.54511
No PI w/ Year Born	0.99497	0.85599	0.99696	6.25373
No PI w/ Gender	0.99388	0.82214	0.99628	6.62424
No PI w/ Jobs	0.99383	0.81852	0.99626	6.74427
No PI w/ First and Last name	0.99406	0.81824	0.99704	7.04467
No PI	0.99318	0.80467	0.99606	6.19589

Table 2. Model Evaluation for XGBoost

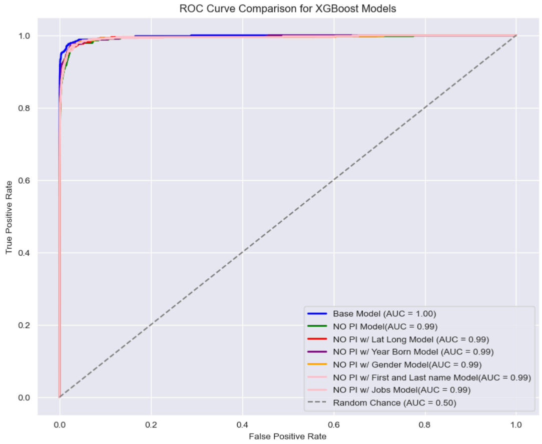


Fig 8. ROC curves for XGBoost models

Looking at the confusion matrices for the XGBoost models (*Fig 9)*, we again see the high-class imbalance, however, we also see that the correctly identified values for each class are higher than those of the Decision Tree Models. For example, the correctly predicted ‘is fraud’ samples on the No PI dataset was 365 for XGBoost whereas it was 308 for Decision Trees - an 18% increase in the number of correct predictions. All of these results indicate that XGBoost is a better overall model than Decision Tree, especially when it comes to handling the class imbalance.

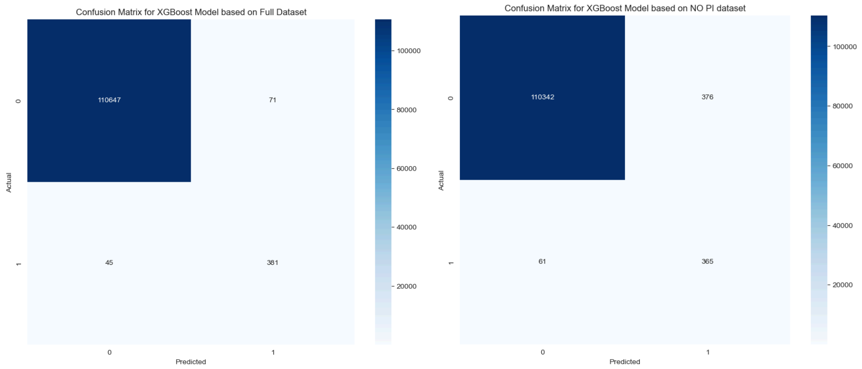


Fig 9. Confusion Matrix for XGBoost Models - Full Dataset vs No PI dataset

B. Feature Importance

Although our above results indicated that the Full PI dataset had marginally higher accuracy and efficacy when it comes to classifying fraudulent transactions, the ablation part of this study was utilized to do a deeper analysis of how each PI feature impacted the model's overall accuracy. This allowed us to conclude whether personal information is truly required for detecting credit card fraud.

Until now, as stated above, we have seen that in general the base models with Full PI, across both Decision Trees and XGBoost have performed better compared to any of the other models. However, we have also seen that for the models with varying levels of PI, there are little to no identifiable differences in results.

Taking a closer look into the feature importance scores for all datasets used in this study gave us a better idea of the role PI actually plays in classifying fraud. Below, in *Fig 10* and *Fig 11*, we have the feature importance graphs for the Full PI and No PI models for Decision Trees and XGBoost. All non-personal features have been colored blue, whereas all personal features have been colored red.

At first glance, there is one overly obvious observation - the ‘amt’ feature (indicating the dollar amount for each transaction) is by far the most important feature across all models and datasets.

This is further corroborated by looking at the other feature importance graphs for all five other datasets with varying degrees of PI (*See Appendix I)*.

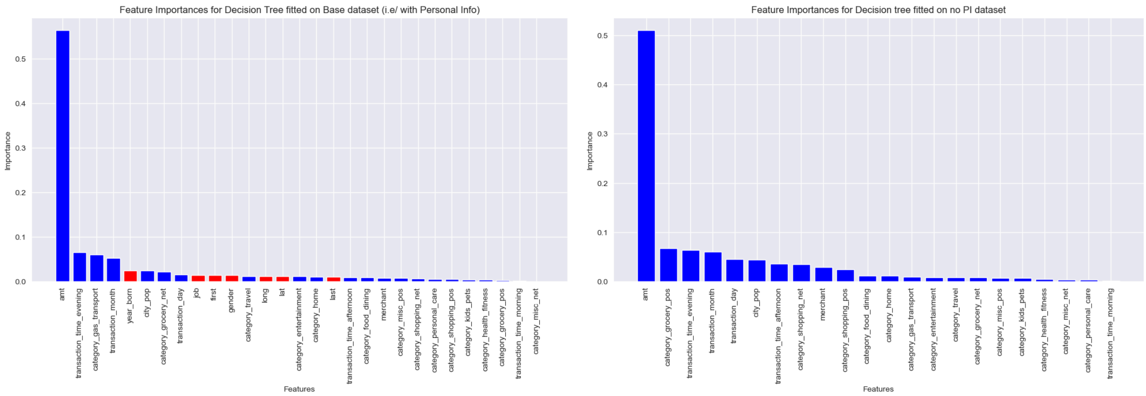


Fig 10. Feature Importance Graph for Decision Tree: Base Dataset vs No PI Dataset

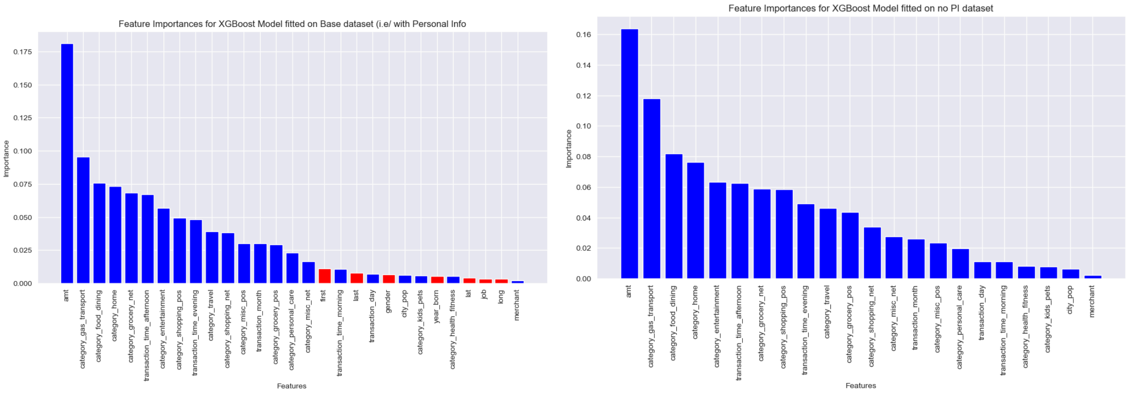


Fig 11. Feature Importance Graph for XGBoost: Base Dataset vs No PI Dataset

After analyzing the feature importance graphs, it is abundantly clear that due to the immense importance of the 'amt' feature, most of the other features have an insignificant relevance for this classification task. Additionally, all personal information features seem to have an even lower importance compared to other non-PI features. This indicates that in general, these features do not have much impact on a model's predictive power.

VI. DISCUSSION

Is a cardholder's personal information needed to accurately determine if a transaction is fraudulent?

No - although our models resulted in different efficacy levels, overall we can see from the feature importance analysis that PI does not play a pivotal role in fraud classification. Furthermore, especially for XGBoost models, all models and datasets regardless of PI levels had relatively high accuracy, and therefore, it can be removed.

Which factors are the most influential in classification?

After analyzing the feature importance graphs for all datasets and models, it is evident that the most influential factor in classifying fraudulent transactions is the amount of a given transaction ('amt'). Furthermore, our analysis shows that none of the PI features from the data play a significant role in the model's ability to accurately detect fraud, and therefore, do not appear to be relevant to the task.

Overall, our study results indicate that a cardholder's personal information is not needed for the classification task of identifying credit card fraud. Although our models results showed higher efficacy for models trained and tested on the Full w/ PI dataset, this difference can be attributed to the fact that all implemented models used the optimal hyperparameters generated from running GridSearchCV on this dataset. For future analysis, we would consider finding optimal hyperparameters for each model and dataset to remove this confounder. Additionally, although the Decision Tree models have relatively low metrics, this can be expected as Decision Tree models are extremely simple and may have had a difficult time learning the data properly, especially given the class imbalance. However, XGBoost performed much better which is expected given it is a more complex model.

Given these findings, it is evident that for future credit card fraud classification methodologies, the requirement for data containing personal cardholder information is not fully necessary - thus alleviating ongoing privacy concerns of people throughout society regarding their personal data being used by large corporations.

VII. CONTRIBUTIONS

Both members of this group played an equal role in the production of this final project. Both worked together closely to brainstorm ideas, plan, and execute the different aspects of this study. For specifics...

Jason Lebov - Jason worked on the preprocessing phase of this study along with helping outline the general approach to the modeling work. He also wrote the first half of the paper, specifically regarding the parts of the study he worked on, as well as editing and revising all work for submission.

Aarush Bedi - Aarush primarily worked on the modeling phase of this project, but also helped with the general approach to the preprocessing steps and overall plan. He also worked to obtain and organize all results shown in this study. For the report, he worked on the second half of the paper, specifically the parts he worked on when conducting the study.

VIII. CODE/DATASET

For access to the code used for this study please access our GitHub link here.
The Dataset was too large to put on our GitHub so for access to the original Kaggle dataset look here.

REFERENCES

[1] Egan, John. “Credit Card Fraud Statistics.” *Bankrate*, www.bankrate.com/finance/credit-cards/credit-card-fraud-statistics/. Accessed 3 May 2024.

[2] Nandi, Asoke K, et al. “Credit Card Fraud Detection Using a Hierarchical Behavior-Knowledge Space Model.” *National Library of Medicine*, 20 Jan. 2020, www.ncbi.nlm.nih.gov/pmc/articles/PMC8775357/.

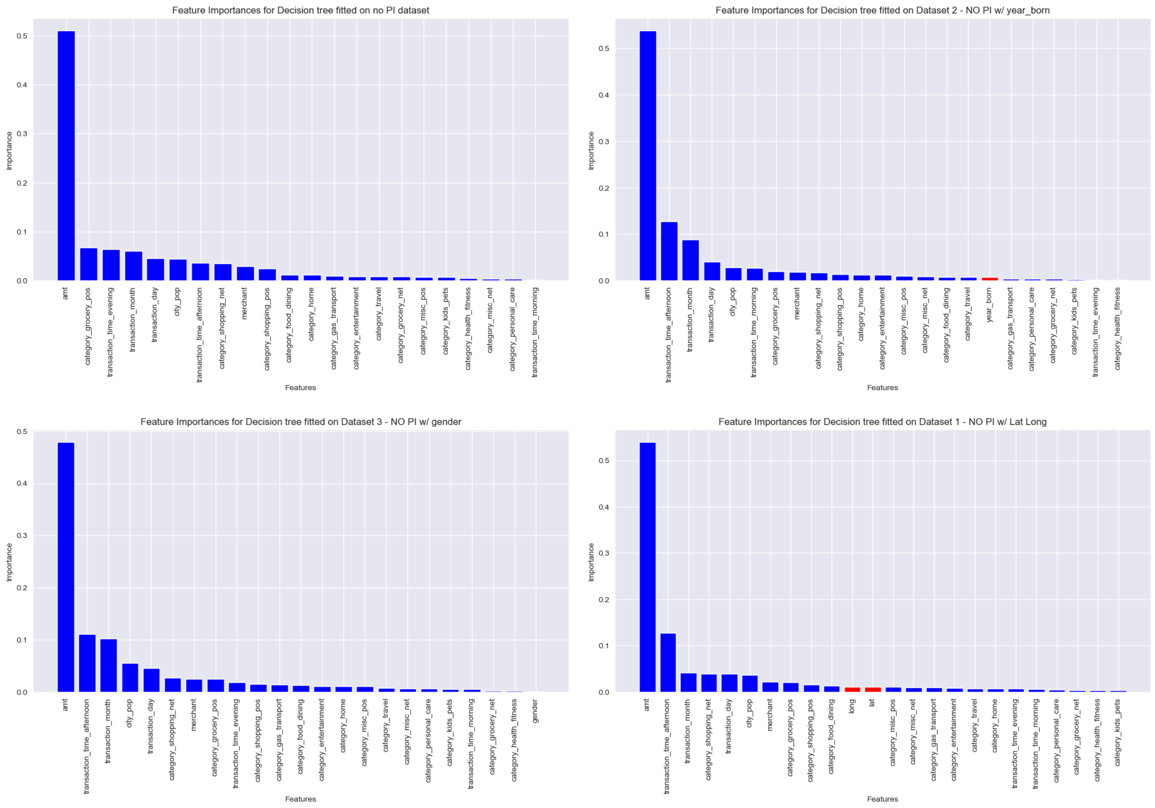
[3] McClain, Colleen. “How Americans View Data Privacy.” Pew Research Center, Pew Research Center, 18 Oct. 2023, www.pewresearch.org/internet/2023/10/18/how-americans-view-data-privacy/.

[4] Btoush, Eyad Abdel Latif Marazqah, et al. “A Systematic Review of Literature on Credit Card Cyber Fraud Detection Using Machine and Deep Learning.” *National Library of Medicine*, 17 Apr. 2023, www.ncbi.nlm.nih.gov/pmc/articles/PMC10280638/.

[5] Shakya, Ronish. *Application of Machine Learning Techniques in Credit Card …*, digitalscholarship.unlv.edu/cgi/viewcontent.cgi?article=4457&context=thesesdissertations. Accessed 28 Apr. 2024.

APPENDIX

Appendix I - Feature Importance Graphs

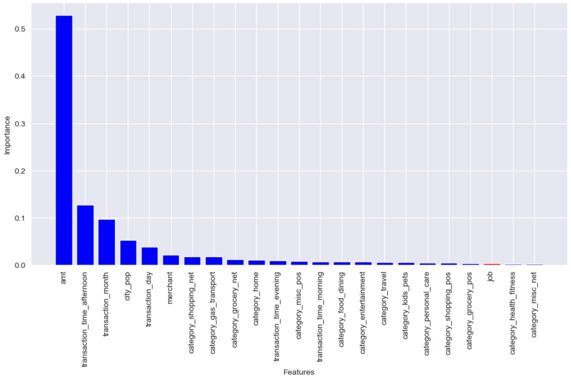


Feature Importances for Decision tree fitted on no PI dataset

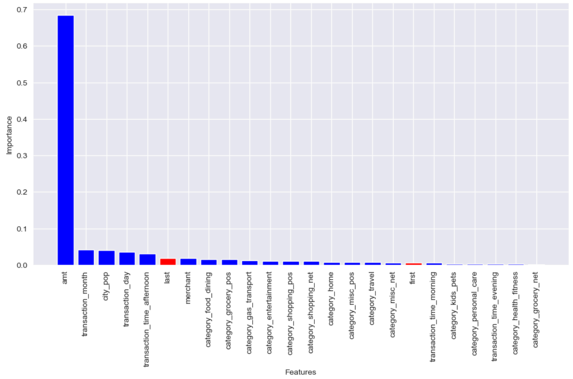
Feature Importances for Decision tree fitted on Dataset 2 - NO PI w/ year_born

Feature Importances for Decision tree fitted on Dataset 3 - NO PI w/ gender

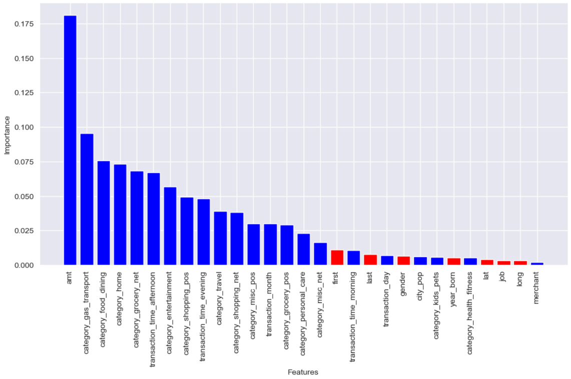
Feature Importances for Decision tree fitted on Dataset 1 - NO PI w/ Lat Long



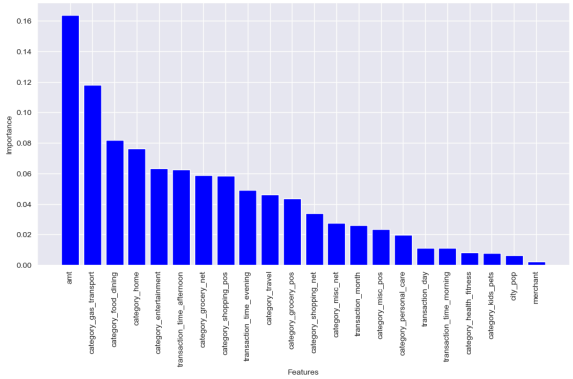
Feature Importances for Decision tree fitted on Dataset 4 - NO PI w/ jobs



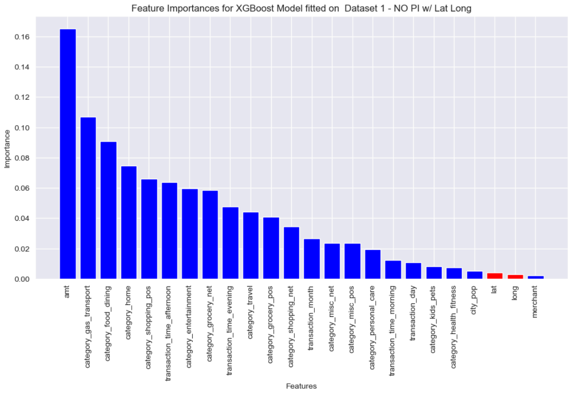
Feature Importances for Decision tree fitted on Dataset 5 - NO PI w/ First and Last name



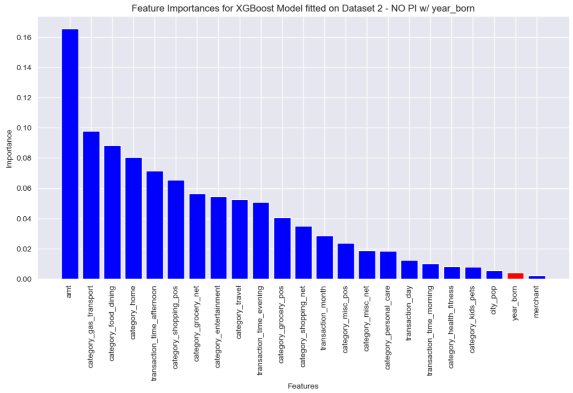
Feature Importances for XGBoost Model fitted on Base dataset (i.e/ with Personal Info



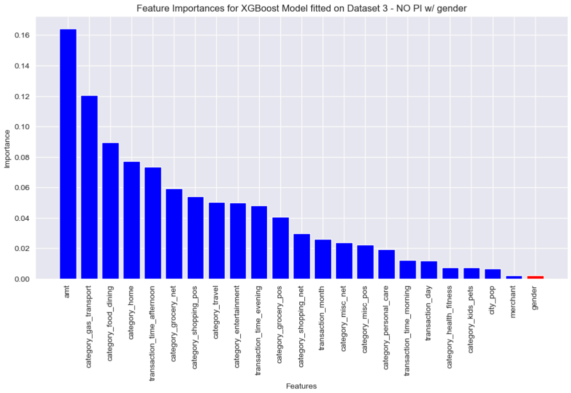
Feature Importances for XGBoost Model fitted on no PI dataset



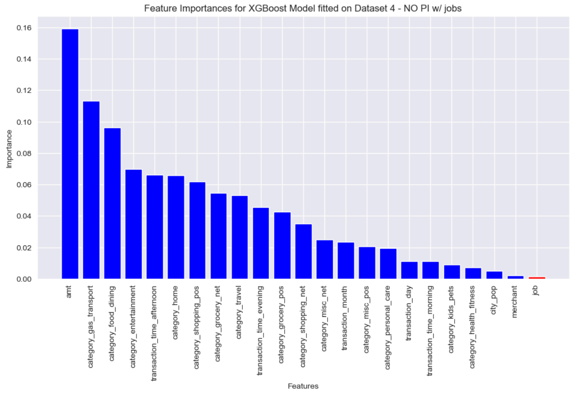
Feature Importances for XGBoost Model fitted on Dataset 1 - NO PI w/ Lat Long



Feature Importances for XGBoost Model fitted on Dataset 2 - NO PI w/ year_born



Feature Importances for XGBoost Model fitted on Dataset 3 - NO PI w/ gender



Feature Importances for XGBoost Model fitted on Dataset 4 - NO PI w/ jobs