**Final Project Report**

Overview of Problem

In this project, we focused on the issue of credit card fraud throughout the world. Credit card fraud has been a problem around the world ever since the start of credit cards in the 1950s and has become a more widespread issue as credit cards have become much more popular today. This kind of fraud occurs when unauthorized users use someone’s credit card information to make purchases or complete other transactions. There are two main kinds of credit card fraud – one where the person committing fraud has the card and uses it in person, and one where they do make a purchase online or over the phone (the card is not physically present). The card-not-present approach can make it much harder to detect fraud when it occurs, for obvious reasons.

When credit cards were relatively newer, fraud was rather easy to commit since transactions were processed manually. Additionally, there were not any security measures in place to prevent criminals from gaining access to other peoples’ cards. In the mid- 20th century, credit card fraud usually occurred mostly through physical theft. However, as credit cards became more popular, criminals found new ways to exploit weaknesses in the system, such as stealing credit card numbers online and creating counterfeit cards. However, in the 1980s and 1990s, credit card companies introduced new and improved security measures such as magnetic stripes and security codes to combat fraud. However, as time went on and technology advanced, criminals also found more techniques such as skimming, phishing, and hacking to steal credit card information. Credit card companies today use many different technologies, such as EMV chips, CVV codes, and biometrics to prevent fraud and protect consumers. Despite all of these extensive security efforts, credit card fraud still causes billions of dollars to be lost every year throughout the world. Overall, as fraudsters become more intelligent, consumers must be very careful with their information to help minimize the chances of falling victim to credit card fraud.

Why It’s an Issue

As I mentioned above, criminals are constantly adapting and learning how to use new technology to their advantage, which causes issues for companies trying to minimize credit card fraud. Once again, these criminals can often obtain credit card information through things like phishing scams, where they trick people into providing their card information by making something like a legitimate advertisement. This is an especially large issue within younger communities who do not have as much online experience and do not know how to be super careful on the Internet. Another huge issue that we often see in the news is data breaches of companies that store credit card information. This has happened to several massive corporations such as Target and PayPal, and it puts millions of customers at risk of having credit card fraud committed with their information. Oftentimes credit card information is then sold over the dark web after these breaches, making it much harder to track. Skimming devices are also a huge problem when it comes to credit card fraud and how to prevent it. These are devices that are able to read and steal information from credit card readers, thus allowing criminals to store it and use it at a later time. In fact, I have seen several news stories over the last month about different stores/restaurants using these devices when people purchase from them in order to steal their credit card information and use it for future purchases. With all of these issues occurring, many people are also increasingly worried about how many payments are shifting primarily to mobile payments and the security surrounding them.

As of today, many credit card and security companies have already implemented many new security features to prevent fraud as much as possible. One of these is EMV chips, which generate unique codes for each transaction, making it much harder to clone credit cards. Another layer of security that is commonly used is two-factor authentication. Many companies also recently started to use advanced technologies to try to detect and prevent credit card fraud as soon as it happens. Both artificial intelligence and machine learning are currently used to try to detect fraudulent transactions by analyzing tons of transactions and tracking patterns (this is how most fraud alerts are created). However, because fraud only occurs in a very small percentage of credit card transactions, this can make it very hard to identify. Additionally, as mentioned before, billions of dollars are still lost every year to credit card fraud despite all of these security measures.

Many of the issues mentioned before relating to criminals obtaining peoples’ credit card information can be very difficult to prevent. Additionally, even if we can find efficient ways to prevent some of the issues, there will always be new ways for criminals to steal information. As a result, I think it is most important to focus on identifying and reporting credit card fraud very soon after the information is stolen to prevent it from getting out of hand and completely wreaking havoc on peoples’ lives. Thus, I think the process of using machine learning and classification to look at individual transactions and identify if they are potentially fraudulent could be the most useful process in reducing credit card fraud. This can allow credit card companies to passively and easily monitor transactions and alert users if fraud is suspected. This would at least provide an extra layer of security for the entire system.

Planned Experiments

We used our dataset and python/machine learning techniques to create a model that can predict when a credit card transaction is an occurrence of fraud (which, according to the dataset, occurs in about 0.17% of the transactions). For our experiments, we started with some basic data exploration tasks, such as creating and displaying histograms of transaction amounts as well as finding the most common transaction categories (both overall and in the case that fraud is found). After this, we created several models using different classification algorithms. Overall, we created a decision tree, a k-nearest neighbors model, and a random forest, each of which were discussed in class this semester. We also experimented with a logistic regression model which required us to do a bit of research. In this process, we trained each classifier using the training data and then made predictions using the testing dataset. After creating each model, we calculated and display relevant statistics (accuracy, precision, recall, and f-measure) that give insight on the performance and ability to properly predict the target variable (whether or not fraud occurred) as well as a confusion matrix for each. From there, we evaluated each model and made an informed decision as to which model was the best choice for this dataset.

After this, we used feature importance to identify which variables are the most important when it comes to accurately classifying the target variable. We then created a new dataset using just the five most important features and the target variable. After this we recreated each model with the new dataset to see how the performance of them changed (if at all) now that they have less variables to focus on. We accomplished these tasks using available python libraries for machine learning and classification. This includes numpy, pandas, sklearn, matplotlib, and several other libraries for each individual model. Overall, there were no major changes from our original plans for this project.

Data Source Description

We found our dataset on Kaggle using the following:<https://www.kaggle.com/datasets/kartik2112/fraud-detection?select=fraudTrain.csv>

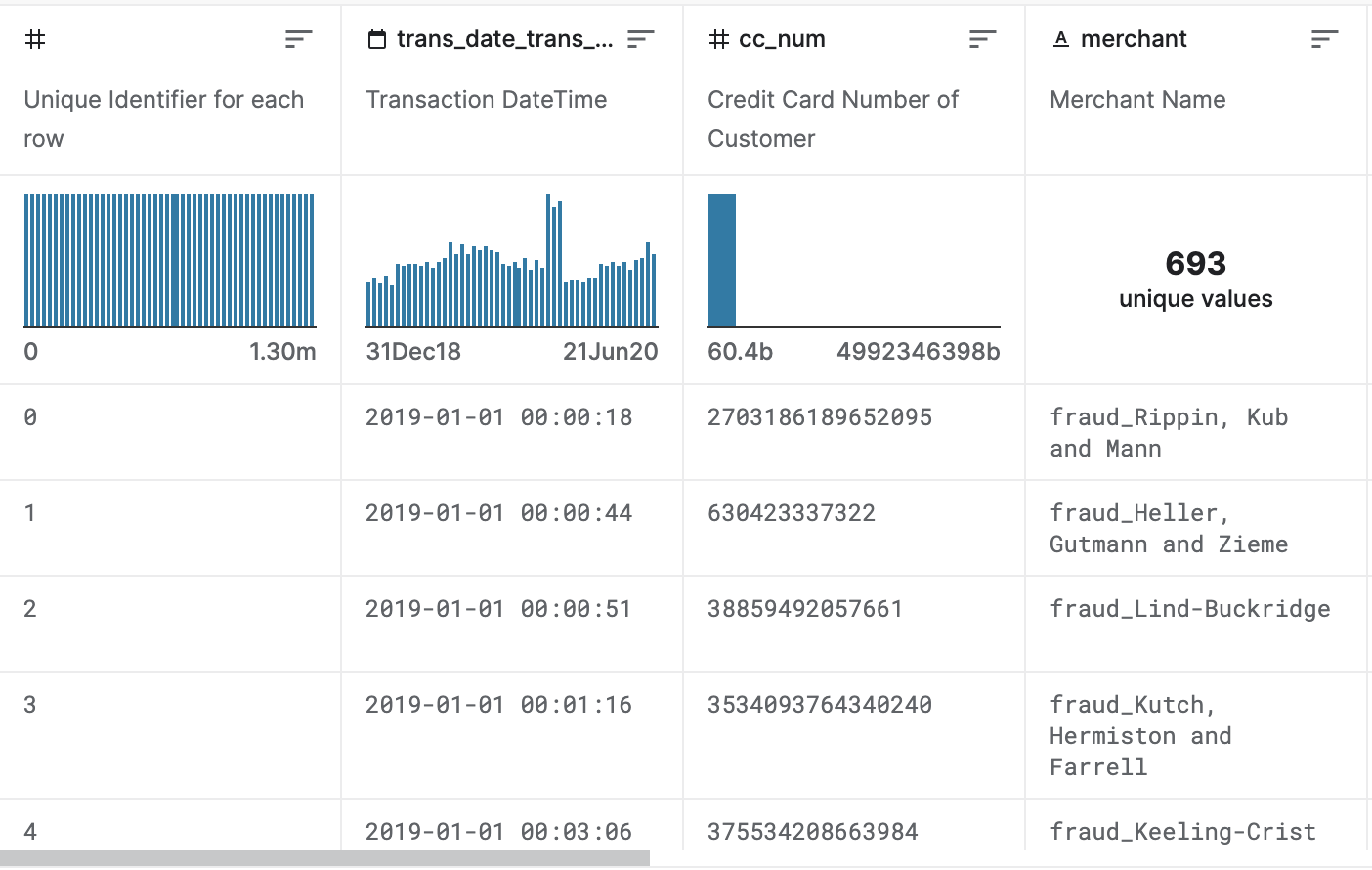
We first transformed our dataset into training and testing splits that were be used for classification. Because our dataset is very large, we used an 85-15 split, with a random 85% of the dataset being used for training and the rest for testing. Overall, we did not really have to do much data cleaning because it is a very popular dataset on Kaggle with tens of thousands of downloads and a high usability score. As a result, the dataset was likely already cleaned by the people who put it together and is ready to be used without having to really change anything. However, we did have to do several data transformations that we were not originally planning on. This is because we realized that our classification models needed to be trained with numerical data and many of our variables were categorical, thus causing errors. To work around this, we changed most of the categorical variables to a numeric form that could keep its original meaning. For example, we mapped gender values of ‘M’ and ‘F’ to 0 and 1, respectively. Additionally, for date of birth, we simply modified entries so that they were in ‘yyyymmdd’ form instead of ‘yyyy-mm-dd’ by removing hyphens. Finally, for the merchant, category, job, city, and state variables, we used the enumerate function to assign each unique value a number. We then applied these values to the original dataset using the map function. We were originally planning on using a OneHotEncoder object, but this conversion caused the session to crash due to not having enough RAM in Google Colab. After these conversions, the only categorical variables left were trans\_date\_trans\_time, first name, last name, street, and trans\_num. We agreed that none of these variables would likely have a large impact on classification, so we decided to ignore them.

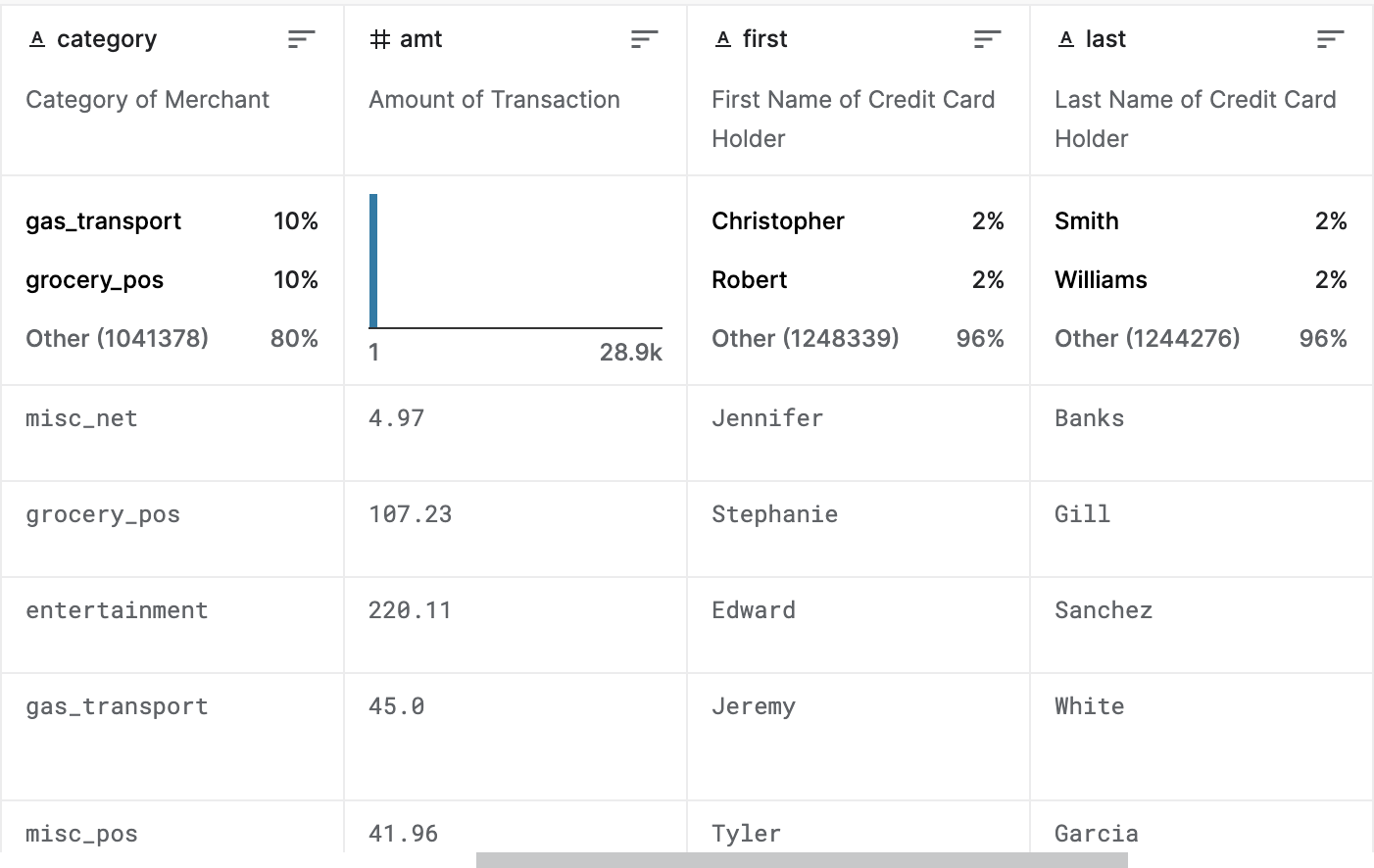
This dataset takes the form of a csv file with hundreds of thousands of entries. This site contains both a train and test file, but because of how large the files were, we decided to use the test file and further split it into train and test sets rather than use the already available split in order to speed things up when running code (I will include only the test file in the submission, however to use the train file you would simply go to the link, download it, and change the name of the file it’s reading at the start of the code). It is filled with simulated credit card transactions (both legitimate and fraudulent) from January 1st 2019 to December 31st 2020. It covers the credit cards of 1000 customers interacting with 800 merchants. Despite the large size of the dataset, we will store this dataset locally for the duration of the project just for ease of access. The dataset contains 21 different features relating to an individual credit card transaction. Additionally, there is one target variable which tells us whether or not the transaction was fraudulent. We will put screenshots of the first few rows for some of the variables below for reference.

We will now go through the different features of the dataset and briefly explain what each of them are. The first feature is the index, so it is just a unique identifier for each row/transaction. Next is trans\_date\_trans\_time which tells us the date and time for a transaction. Cc\_num is the credit card number for the customer. Merchant is the name of the merchant in the transaction. Once again, this comes from a pool of 800 different merchants. Category describes the category of the merchant and includes things like gas\_transport, grocery\_pos, home, kids\_pets, and many more. Amt is the amount of the transaction in dollars. First is the first name of the credit card holder (one of 1000 people) and last is the last name of the card holder. Gender relates to the gender of the card holder. Street is the street address of the card holder. City, state, and zip are all related to the position of the card holder as well. Lat and long are the latitude and longitude of the credit card holder. City\_pop is the population of the credit card holder’s city. Job is the job of the credit card holder which obviously will be a wide array of positions (just some examples are film/video editor, exhibition designer, naval architect, and materials engineer). Dob is the date of birth of the credit card holder. Trans\_num is the transaction number, each of which will be unique. Unix\_time is the UNIX time of transaction, which is the number of seconds elapsed since January 1st 1970. Merch\_lat and merch\_long describe the latitude and longitude of the merchant. Finally, is\_fraud is the target variable and tells whether or not the transaction is fraudulent (0 meaning no fraud and 1 meaning fraud).

Data Analysis

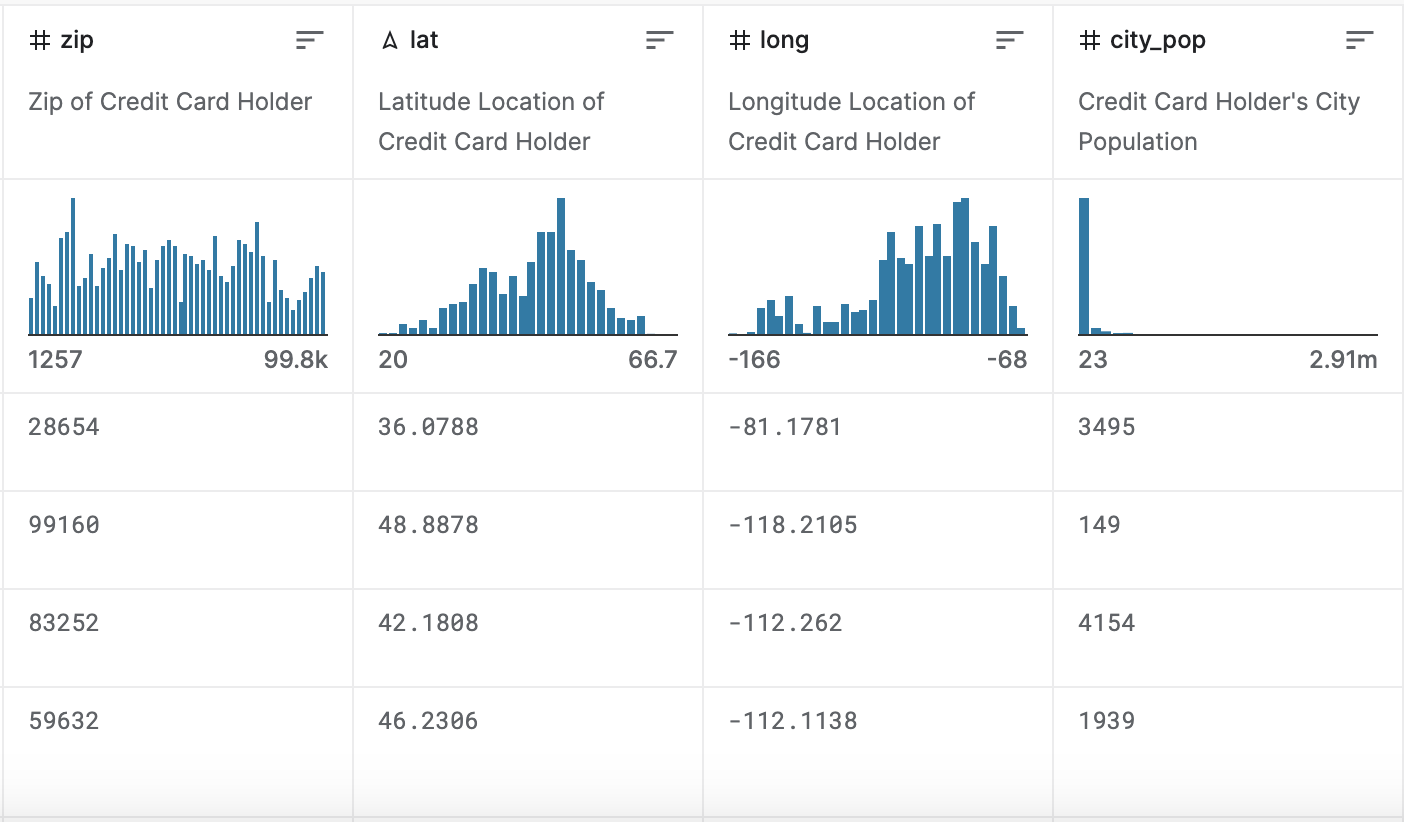
As mentioned above, the dataset contains 21 features and one target variable (fraud or no fraud) relating to individual transactions. We will put screenshots of the first few rows for some of the variables below for reference.





Table

Description automatically generated



After our data transformations, there were 16 features left that we decided to use. We decided to prepare the data the way we did because it allowed us to keep all of the features we thought might be important when classifying data while also still being able to use the models we originally planned to use. Additionally, using the dictionaries to map to numeric values is convenient because if we needed to check what a numeric value meant, we could simply print the dictionary and check (we commented out print lines for ease). Additionally, as discussed above, we decided to use an 85/15 train/test split because of how massive our dataset is. It allows us to give the models lots of training points to go off of while still having a large amount of data left for testing and predictions. The first exploratory data analysis task that we completed was to create a histogram of the amount of transactions, just to see what price most transactions were around, giving the following figure:

Chart, histogram

Description automatically generated

From this figure, we can see that transaction amount is unimodal and skewed right with many outliers and a median close to 0. Even with excluding many of the outliers by stopping the range at 1000, it was still difficult to truly see the distribution of transaction amount so we replotted the histogram but cut the range off at 100 instead. This gave us a more informative figure:

Chart, histogram

Description automatically generated

From this histogram, we can see that the majority of data for amount is around $10. However, the median is closer to $50 (47.29) because of all of the high outliers, such as the maximum which is over 20,000. After creating these histograms, we decided to look at the most common categories, and, more specifically, the most common categories where fraud occurs. We produced the following tables:

Table

Description automatically generated with medium confidence

A picture containing text

Description automatically generated

From these tables, we can see that gas and groceries are the most common transaction categories in general, while online shopping is the most common fraudulent transaction category. This makes sense based on what we know about card-not-present fraud.

Results

Overall, we found that most of the models did not perform particularly well with this dataset. This is likely a result of there only being fraud present in 0.17% of the data entries, making it much more difficult to find potential instances of fraud and accurately predict them. While all of the models had very high accuracy, this is misleading because they were trained to predict no fraud the vast majority of the time and the fact that the dataset is so large leads to a better accuracy score. However, if you analyze each model based on their F-measure score, we can see that they struggled. The logistic regression model was the worst as it did not predict any fraudulent instances. After this was the KNN model, with an F-measure around 0.37. The decision tree model was the next best, just barely above KNN. Finally, the random forest model was by far the best and actually performed decent, as it had an F-measure of 0.77 (which is especially good for a dataset where true instances for target variable are rather rare). Below is the confusion matrix for the random forest model:

Chart, treemap chart

Description automatically generated

After creating these four models, we calculated the feature importance values using the random forest feature\_importances\_ method. Below we have included a table of the importance value for each feature as well as a bar graph of these values:

Text

Description automatically generated

Chart, histogram

Description automatically generated

We can see that the five most important features for classification are amt, category, unix\_time, city, and merchant. We then created a subset of the data with only these features and the target variable and recreated all of our models to see how it would affect their performance. In general, we saw that the performance of our models either stayed the same or got worse. We think this is likely due to the fact that we narrowed down the dataset too much and the models did not have enough features to go off of when making decisions about classification. Despite this, our model rankings from worst to best stayed the same, with random forest by far the best. Below is a table of the statistics for each of our models:

Calendar

Description automatically generated

Conclusion

Overall, we were able to answer our researched questions by using machine learning models that we learned about in class as well as other models that we had heard about outside of the classroom but not discussed. We were able to utilize and add on to the knowledge we gained from assignment 4 while gaining more experience with machine learning and different classification models that are available in python. All of the methods at our disposal helped make answering our researched questions much easier than we thought it might be. In general, we learned how to create more classification models as well as utilize them in different ways to better explore a massive dataset. We also learned about feature importance and how it can play a role in classification of data. One of our biggest takeaways from this project is that, while machine learning can be very useful in today’s world, it can still be very difficult to use it to actually solve important problems. While we obviously did not think that we were going to be able to easily predict credit card fraud, this project showed us just how difficult of a problem it is to solve even with powerful machine learning tools. This is because even though it seems to happen often, in the grand scheme of all credit card transactions, it is extremely rare and thus hard to predict regardless of how much data you have available to train a model. In the end, this project provided us with valuable exposure and experience to using machine learning with real-world issues.