ISyE 6740 - Spring 2021
Final Project

Team Member Names: Homayoun Gerami, GT ID: 903551531, hgerami@gatech.edu
Alain Vandormael, GT ID: 903563449, avandormael3@gatech.edu

Project Title: Credit Card Fraud Detection Using Machine Learning Algorithms

1 Introduction

In Europe, the average number of credit cards carried per inhabitant ranges from 0.8 to 3.9.[1] With the widespread adoption of credit cards has come an increasing prevalence of credit card fraud transactions. Worldwide, credit card fraud has tripled from $9.84 billion to $32.39 billion between 2011 to 2020,[1] with European issued credit card fraud amounting to 1.83 billion Euros in 2020.[2, 3] It is therefore critical that credit card companies continue to develop methods to accurately and quickly identify and prevent fraudulent transactions.

In the past, criminals would steal numbers from credit or debit cards and print them onto blank plastic cards to use at brick-and-mortar stores.[4] To reduce this type of fraud, Visa and Mastercard, the two major credit card providers, mandated that banks and merchants introduce EMV chip card technology in 2015.[5] This would require a legitimate credit card user to enter a secret PIN for each transaction. Despite the protective measures of EMV chips, experts predict that the cost of other forms of fraud, notably online credit card fraud, is likely to increase substantially over time.[3]

In this study, we use machine learning (ML) methods to accurately classify online credit card transactions as fraudulent or not. Several papers have already used ML algorithms to classify fraudulent credit card transactions,[6, 7, 8, 9, 10] which will inform our learning. One of the challenges with credit card fraud data is that it tends to be highly imbalanced. Predictive modelling can be affected by imbalanced classifications, because most ML algorithms were designed to deal with approximately equal number of cases in each class.[11] An unequal distribution of cases occurs when the event of interest is rare or if there is biased sampling. The data is considered to be imbalanced when the ratio of the Minority Class (the class that has less examples) to the Majority Class (the class that has more examples) is greater than 1:4. Typically with credit card data, the imbalance can be severe and as 1:5000.[12]

In this paper, we leverage a Logistic Regression, Support Vector Machine (SVM), and Random Forest classifier to predict fraudulent transactions using real-world data. We use a credit card dataset to better understanding the performance of ML methods when making predictions with severely imbalanced data. Below, we describe the credit card dataset, our methodology to address imbalanced data, present our results, and evaluate the performance of the ML methods to correctly classify fraudulent credit card transactions.

2 Methods

2.1 Data

The credit card data comes from Kaggle (`https://www.kaggle.com/mlg-ulb/creditcardfraud`) and contains transactions made by European credit cardholders in two days of September 2013. The dataset contains 30 features, with two features `Time` and `Amount` in their original format. The `Time` variable represents the time in seconds since the first transaction over the 48 hour observation period. The remaining 28 features are numerical and were derived from a principal component analysis (PCA). This was done to protect user identities and sensitive features. These features are named `V1` to `V28`. The dataset is highly imbalanced,[12] with the Minority cases accounting for only 0.17% of all transactions.

2.2 Analytical approach

We performed several tasks to prepare the data for analysis using `Python` version 3.8. We decided to include all principal components in the final dataset (`V1, V2, ..., V28`). Because the range of values in the `Amount` feature was wide ($1–$25,000), we transformed this feature using log base 10. We dropped the original `Amount` feature and included the log 10 version in the final dataset. We decided to drop the `Time` feature from the final dataset since this was relative to the first transaction and not very informative. (This feature would have been informative if it gave the time of day that the transaction took place.) In all, the final dataset consisted of 5 features and the ouctome variable.

To classify credit card fraud, we evaluated the accuracy of three ML approaches: Logistic Regression, Support Vector Machine, and Random Forest Regression, which are available in the `sklearn` library. For the SVC model, we selected a `rbf` kernel. For the Random Forest model, we selected a maximum depth of 5 and 5 minimum samples per leaf. For the Logistic Regression, we used the default settings.

Next, we used Repeated Stratified KFold sampling with $k = 5$ and 3 repeats to split the data into training and testing datasets. For the kth training set, we did synthetic random sampling to balance the data using the `RandomUnderSampler` method from the `imblearn` library. To better assess the effect of undersampling, we undersampled 15%, 10%, and 2% of the majority cases.

After balancing the training datasets, we then used the trained models on the original (unbalanced) test datasets. To evaluate the models, we considered several metrics that are appropriate for imbalanced data and that emphasize the accuracy of predictions in the Minority class. That is, to evaluate the performance of the ML models to classify the small number of fraudulent cases, we focused on the F1, F2, Precision, Recall, and AUC using Precision and Recall (AUC-PR) scores. We justify these metrics in greater detail in the *Conclusion* section. For each model, we plotted the confusion matrices.

3 Results

There were 492 (0.17%) fraudulent transactions out of 284,627 total transactions. The data are severely imbalanced, which justifies our use of under-sampling. Figure 1 (top left panel) shows the distribution of all transaction amounts. There were 3 amounts $> \$15,000$ that were non-fraudulent transactions. To improve visualization of the data, we show the transaction amounts $< \$15,000$ by fraud status in (top right panel). The box plots show that all of the fraudulent transactions were $< \$2000$ (bottom left panel), with most of the transactions below \$100. Fraudulent transactions had a higher Q3 than non-fraudulent transactions.

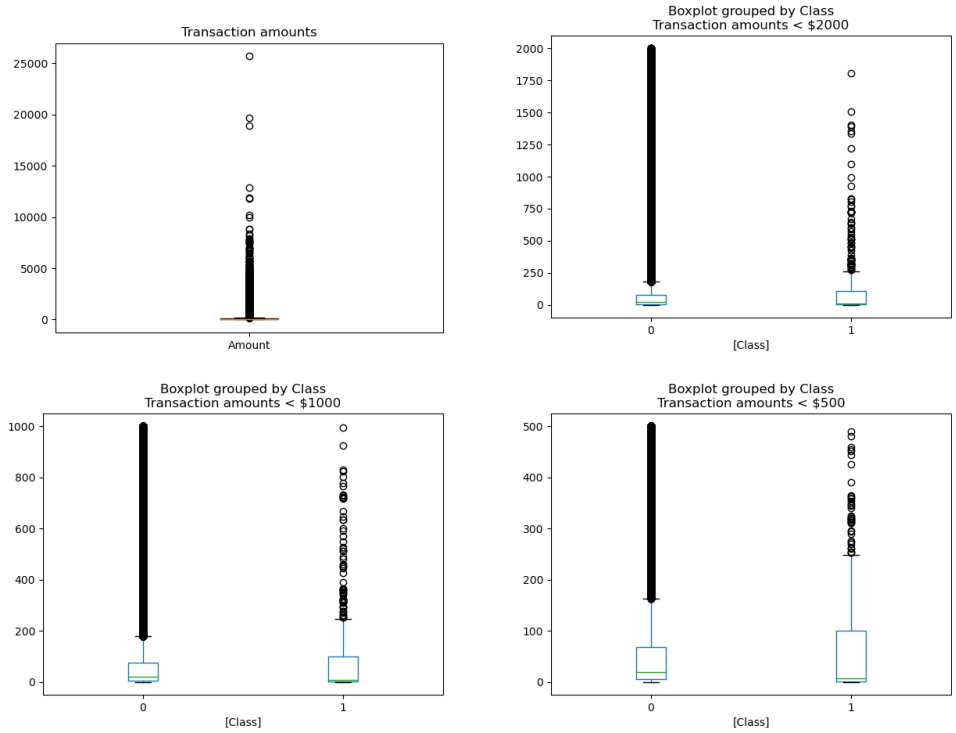

Figure 1: Box plots for all transaction amounts (top left) and transactions $< \$15,000$ (top right) and transactions $< \$2,000$ (bottom left) and $< \$500$ (bottom right) by fraud status.

In Figure 2, we show the number of fraud and non-fraudulent transactions by hour over the 48 hour observation period. The x-axis represents the hour since the first transaction, which is how the data was given to us. For the non-fraudulent transactions, the results for the fraudulent transactions (left panel) show two clear bi-modal distributions, which most likely corresponds with daytime transactions for each of the two days. For the fraudulent transactions, we do not see a clear pattern or distribution in the number of transaction

counts. The highest number of fraudulent transactions was 32, which occurred in the 28th hour.

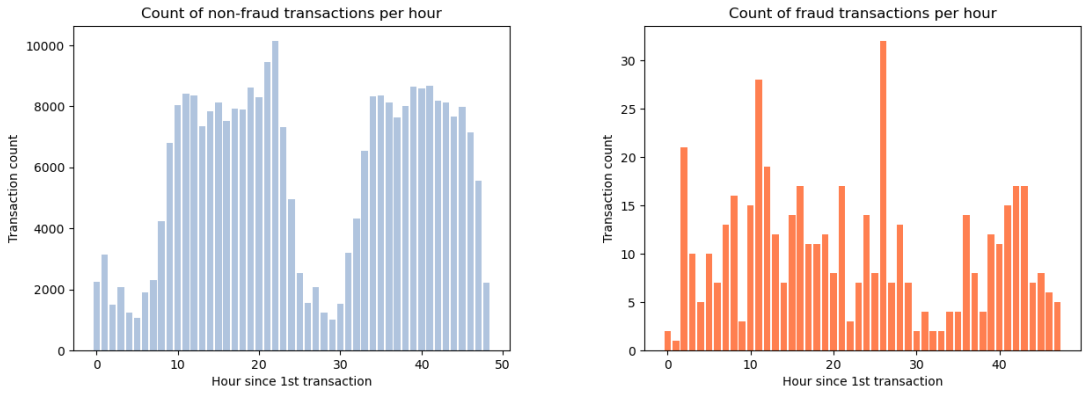

Figure 2: Density plot of credit card transaction amounts over time by fraud status.

Below, we report the results for each of the models and 2% under-sampling for the training datasets. The 2% undersampling strategy resulted in 19,700 non-fraudulent cases to 394 fraudulent cases in the training dataset. For the testing dataset, there were 98 fraudulent cases to 56,863 non-fraudulent cases. All three models had a very high Accuracy score, ≥ 0.99, as shown in Table 1 and an AUC-ROC $= 0.90$. This result is expected given the smaller number of cases in the Minority class. We discuss this implication further in the *Conclusion* section.

The F1 and F2 scores ranged between 0.76–0.81 for the three models and the Precision and Recall scores ranged between 0.71–0.82. For example, of the truly fraudulent cases, the Logistic Classifier correctly predicts 82% (Recall). Out of the transactions classified as fraudulent, 71% were truly fraudulent (Precision). The F1 scores range between 25–38%, with the SVM model having the highest score. The F2 scores are better than the F1 scores because of the higher weighting given to the Recall scores in this calculation. (This is discussed in greater detail in the next section.) In Figure 3 we show the confusion matrix and AUC-PR plot. The AUC-PR score suggests that the Random Forest Classifier performed the best of the three models.

The 10% undersampling strategy resulted in 3,940 non-fraudulent cases to 394 fraudulent cases. The performance metrics are shown in Table 1. Compared with the 2% undersampling results, we see lower F1 scores, while the Accuracy and AUC scores are mostly similar between the two sets of results. The confusion matrices and AUC-PR curves are shown in Figure 4 in the Appendix, where the Random Forest Classifier is again the best performer with respect to the AUC-PR. We see that the Precision scores for the three models are lower (ranging from 0.35–0.65) than those in the 2% undersampled dataset, whereas the Recall scores are higher (approx. 0.85). Overall, the F1 and F2 scores have slightly worse performance than the those in the 2% undersampled dataset.

The 15% undersampling strategy resulted in 2,626 non-fraudulent cases to 394 fraudulent cases. The performance metrics are shown in Table 1. Compared with the 2% and 10% undersampling results, we see higher AUC scores than before and the Accuracy is still $\geq 99\%$. We also see slightly lower F1 and F2 scores than before, but there is a slight increase in the Recall to approximately 0.85. The Random Forest has the highest AUC-PR (0.75), F1 (0.69), and F2 (0.77) scores. The confusion matrices and AUC-PR curves are shown in Figure 5 in the Appendix. Overall, the results indicate that the algorithms can be further improved to better classify fraudulent transactions.

	F1	**F2**	**AUC-ROC**	**AUC-PR**	**Precision**	**Recall**	**Accuracy**
2% Undersampling							
Logistic	0.76	0.79	0.90	0.75	0.71	0.82	0.99
SVM	0.79	0.80	0.90	0.75	0.78	0.80	0.99
Random Forest	0.80	0.81	0.90	0.79	0.80	0.81	0.99
10% Undersampling							
Logistic	0.50	0.66	0.92	0.71	0.35	0.85	0.99
SVM	0.65	0.75	0.91	0.73	0.54	0.84	0.99
Random Forest	0.73	0.80	0.92	0.75	0.65	0.84	0.99
15% Undersampling							
Logistic	0.39	0.58	0.93	0.72	0.25	0.86	0.99
SVM	0.60	0.72	0.92	0.73	0.47	0.84	0.99
Random Forest	0.69	0.77	0.92	0.75	0.59	0.84	0.99

Table 1: Perfomance of three ML algorithms to detect credit card fraud using 2%, 10% and 15% undersampling.

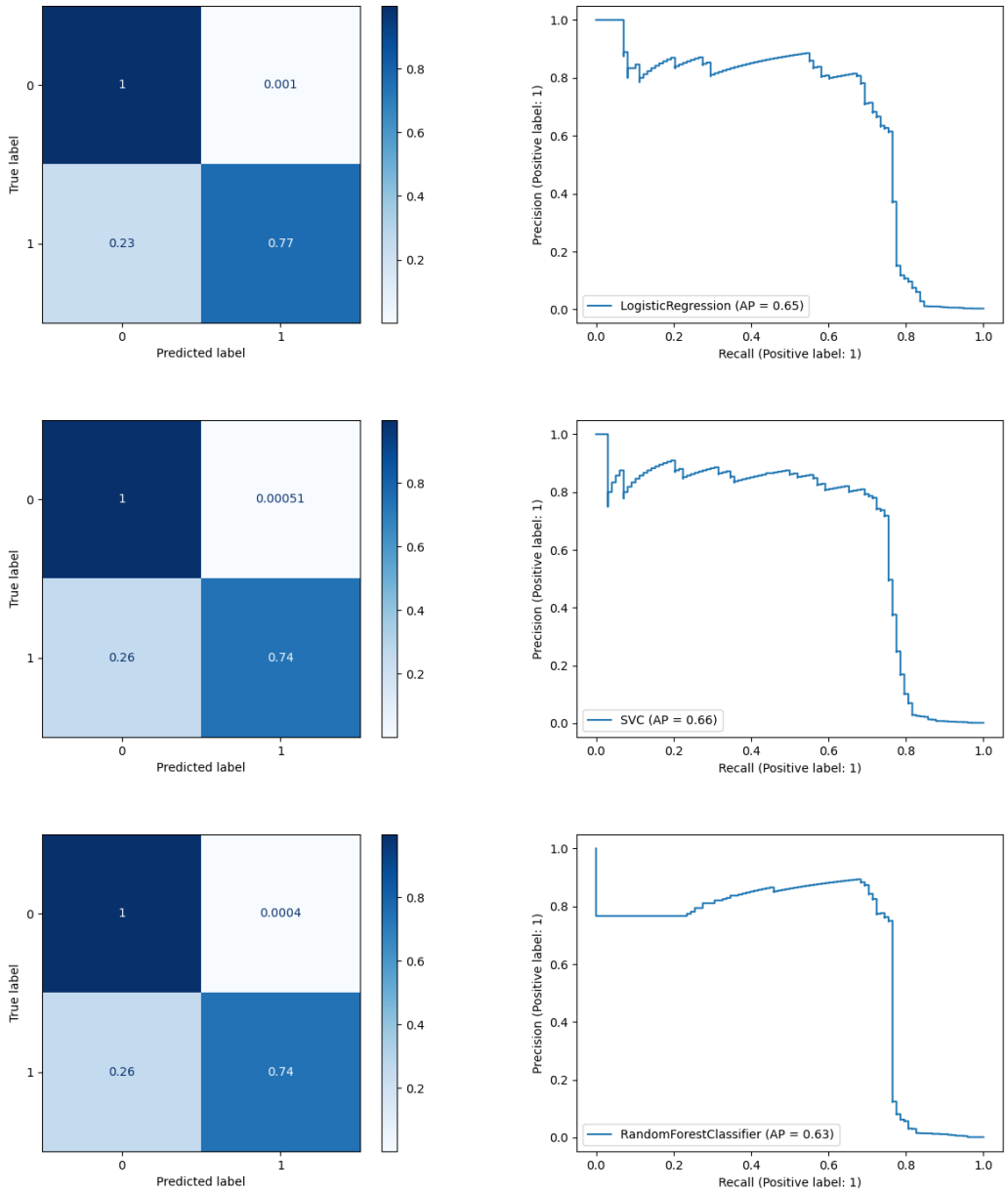

Figure 3: Confusion matrix and AUC curve for three ML methods using 2% under-sampling to address imbalancing. Note that these results are from a single fold rather than an average over all 5 folds.

4 Conclusion

In this paper we studied the performance of three ML methods to classify fraudulent transactions with data collected from online credit card transactions. The three ML methods that we evaluated were a Logistic Regression, Support Vector Machine, and Random Forest classifier. In the next sections, we summarize our results and describe some of the challenges that we encountered with the data.

The data that we used was cleaned and had already been tranformed using PCA, with the exception of the `Time` and `Amount` variable, which were in their raw formats. We decided to transform the `Amount` variable using base log 10 because of the wide range of values. We also decided to drop the `Time` since first transaction variable because it was not informative. (It was not clear how this information could be used as a reference point in future ML models.) One possible limitation of our analysis is that the data were already PCA transformed before we split it using stratified Kfold cross-validation. Such a transformation could create a dependency between the test and train datasets, potentially undermining the motivation for cross-validation. However, there is little we could do about this potential limitation as this is how we received the data and it was critical that the patient data remain anonymized (through the PCA transformation).

The biggest data challenge that we encountered was the severe imbalance in number of fraudulent versus non-fraudulent transactions.[12] The Minority (fraudulent) cases accounted for 0.17% of all the cases. We therefore had to spend some time researching the proper methods needed to prepare the data for the ML algorithms. This research included knowing how to select the appropriate metrics to analyze the performance of these algorithms.

Often, the Accuracy or ROC-AUC score is used to measure the performance of ML algorithms. But the Accuracy score can be a poor measure for imbalanced classifications, because the larger number of Majority cases can easily overwhelm the number of cases in the Minority class, such that unskilled algorithms can achieve Accuray scores of 99% or higher. This clearly evident in our results, where all models achived an Accuracy score $\geq 99\%$ across the three different undersampled datasets, as shown in Table 1.

The Precision, Recall, and F1 scores are recommended as alternatives to the Accuracy and AUC-ROC scores for imbalanced data.[11] Precision is defined as:

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}}. \tag{1}$$

Recall is defined as:

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}. \tag{2}$$

And the F1 as:

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{3}$$

These metrics are important because they calculate the accuracy for the Minority class, that is, the ability of the ML models to correctly classify the small number of fraudulent transactions in the dataset.

The choice of the Precision, Recall, or F1 score depends on the context and the cost of false negatives or false positives. Under the assumption that a Bank finances and manages the credit card transactions, we imagine that false negatives (truly fraudulent transactions that go undetected) would be more costly than false positives (non-fraudulent transactions that are flagged as fraudulent). Fraudulent transactions that go undetected can be costly. By the time it is eventually detected, the fraudulent transaction may not be able to be electronically reversed. Thus, the Bank would suffer the full cost of the fraudulent transaction because it would have to fully reimburse the customer, in addition to paying the costs associated with tracking down and resolving the fraudulent transaction.

With respect to accuracy, the Precision may be more appropriate when false positives are more costly and the Recall may be more important when false negatives are more costly. Thus, we focused on the Recall for minimizing the number of false positives, since we wanted less fraudulent transactions to go undetected. As shown in Eq (3), the F1 measure defines a way to combine the information provided by the Precision and Recall. It gives equal weight to both Precision and Recall, such that a good F1 score can only be obtained

if both measures are good. For this reason, we evaluated the ML models using the F1 score. In addition, because of we wanted to minimize false negatives, we also used the F2 measure, defined as:

$$F2 = \frac{(1+\beta^2) \times \text{Precision} \times \text{Recall}}{\beta^2 \times \text{Precision} + \text{Recall}}, \tag{4}$$

where $\beta = 2$ such that less weight is placed on Precision and more weight on Recall. In addition to the F1, F2, Precision, and Recall scores, we also plotted the AUC-PR curve, which is based on the Precision and Recall. Of all the metrics selected to evaluate the performance of the ML models, we are inclined to give highest weight to the F2 score because of how it weights the false positives and negatives.

Our results show a relatively good performance for the Random Forest classifier, with F2 scores of 0.81, 0.80, and 0.77 across the 2%, 10%, and 15% undersampled datasets, respectively. SVM was the next best classifier based on the F2 scores. Overall, Precision was good for the 2% undersampled dataset, but got worse with higher undersampling. However, we were less concerned about the Precision than the Recall. Our results show that the Recall was good, ranging from 0.80–0.84 across the three models and undersampled datasets. The Random Forest classifier performed the best of the three ML methods. Overall, we conclude that more work can be done with the ML methods to improve performance. This could involve tuning the models so that they can maximize the number of correct fraudulent predictions while minimizing the number of false negatives.

References

[1] Patrick Whatman. Credit card statistics 2021: 65+ facts for Europe, UK, and US. *https://blog.spendesk.com/en/credit-card-statistics*, 2021.

[2] Credit card fraud in Europe hits $1.83B — Payments Dive. *https://www.paymentsdive.com/ex/mpt/news/credit-card-fraud-in-europe-hits-155-billion-euros/?*

[3] European Central Bank. Sixth report on card fraud. *https://www.ecb.europa.eu/pub/cardfraud/html/ecb.cardfraudreport202008~521edb602b.en.html*, 2020.

[4] A Delmaire and J Pointon. Credit card fraud and detection techniques : a review Title Credit card fraud and detection techniques : a review Credit card fraud and detection techniques: a review. Technical Report 2, 2009.

[5] Tim Smyth. The EMV Credit Card Mandate – What it is and what it isn't. *https://www.smythretail.com/technology/emv-credit-card-mandate-isnt/*, 2014.

[6] Dejan Varmedja, Mirjana Karanovic, Srdjan Sladojevic, Marko Arsenovic, and Andras Anderla. Credit card fraud detection-machine learning methods. In *2019 18th International Symposium INFOTEH-JAHORINA (INFOTEH)*, pages 1–5. IEEE, 2019.

[7] Naoufal Rtayli and Nourddine Enneya. Selection features and support vector machine for credit card risk identification. *Procedia Manufacturing*, 46:941–948, 2020.

[8] RB Asha and Suresh Kumar KR. Credit card fraud detection using artificial neural network. *Global Transitions Proceedings*, 2021.

[9] Andrea Dal Pozzolo, Giacomo Boracchi, Olivier Caelen, Cesare Alippi, and Gianluca Bontempi. Credit card fraud detection: a realistic modeling and a novel learning strategy. *IEEE transactions on neural networks and learning systems*, 29(8):3784–3797, 2017.

[10] Fabrizio Carcillo, Yann-Aël Le Borgne, Olivier Caelen, Yacine Kessaci, Frédéric Oblé, and Gianluca Bontempi. Combining unsupervised and supervised learning in credit card fraud detection. *Information Sciences*, 2019.

[11] Jason Brownlee. Random Oversampling and Undersampling for Imbalanced Classification. *https://machinelearningmastery.com/random-oversampling-and-undersampling-for-imbalanced-classification/*, 2021.

[12] Raphael Pierre. Detecting Financial Fraud Using Machine Learning: Winning the War Against Imbalanced Data. *https://rpubs.com/chidungkt/448728*, 2018.

Appendix

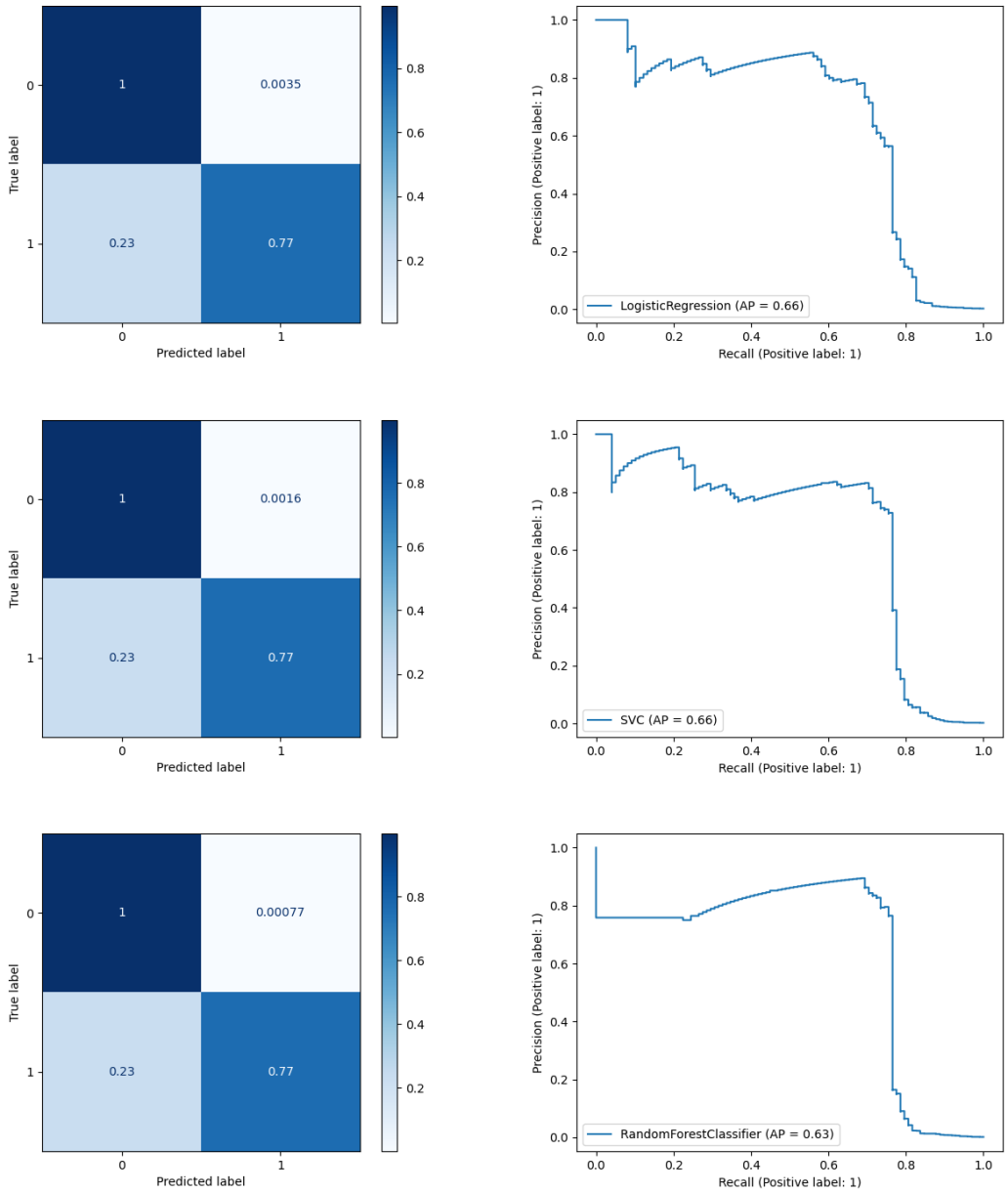

Figure 4: Confusion matrix and AUC curve for three ML methods using 10% under-sampling to address imbalancing. Note that these results are from a single fold rather than an average over all 5 folds.

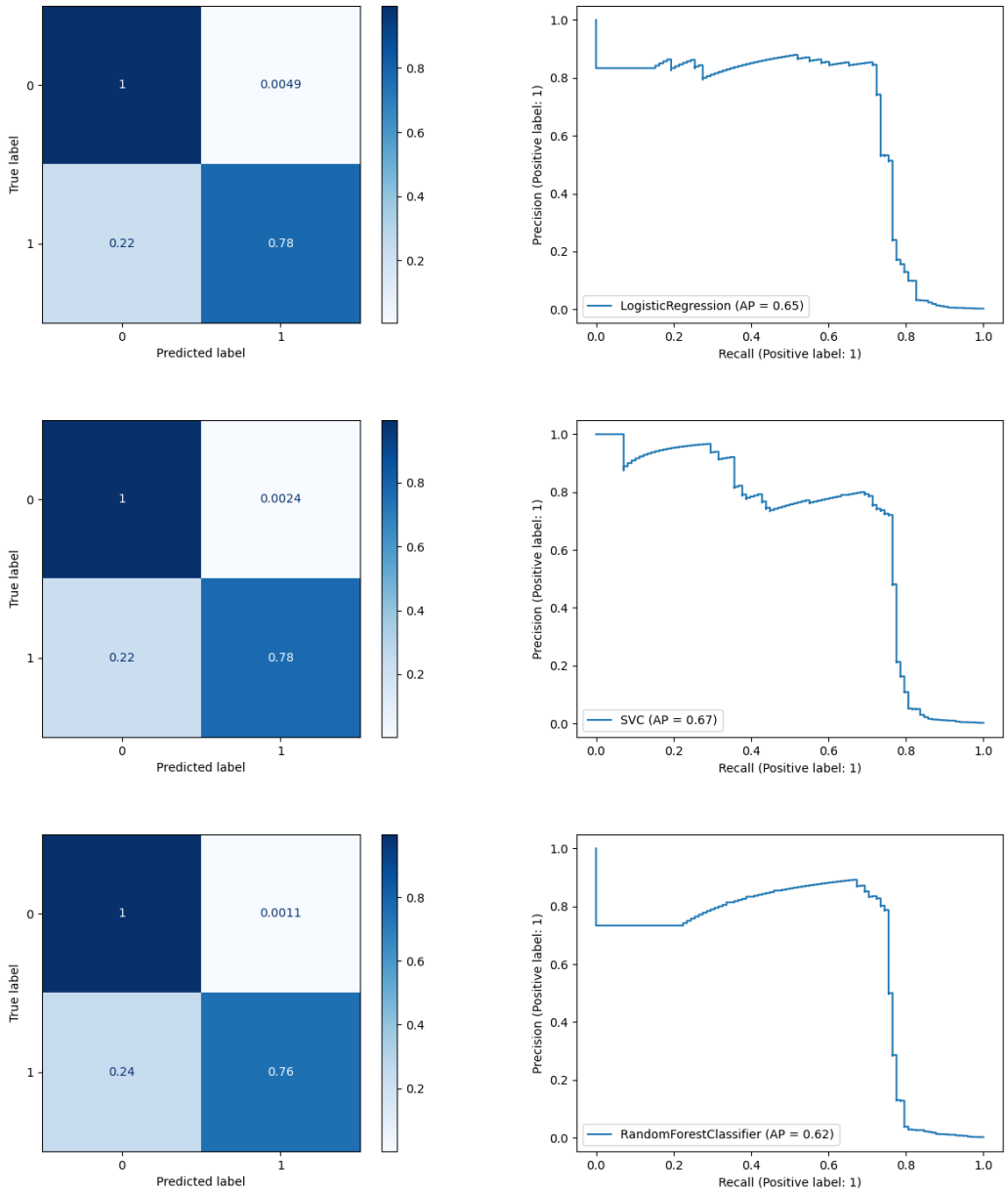

Figure 5: Confusion matrix and AUC curve for three ML methods using 15% under-sampling to address imbalancing. Note that these results are from a single fold rather than an average over all 5 folds.