Leveraging Machine Learning Based Fraud Prevention for Customer Experience Optimization

Introduction

Fraudulent practices in the financial sector results in an annual loss of over $3.5 trillion worldwide.[1] Therefore, the prevention of fraudulent practices is one of the most important and challenging issues currently facing the global financial system.

The cryptocurrency market is a small segment of the finance industry; however, it is considered to be the next step in the evolution of the global financial ecosystem.[2] Bitcoin was introduced in 2008 and is the first blockchain-based digital currency. The underlying blockchain technology is a decentralized, distributed, and immutable public ledger that allows users to execute trustless, cryptographically secured peer-to-peer transactions. The global rise of cryptocurrency usage can be attributed to the lack of a centralized third party which leads to increased financial inclusion and more access to financial resources and services (e.g., money management, access to capital via loan services, money transfer services, insurance, etc.), particularly in countries with a high percentage of unbanked or underbanked populations. In addition, cryptocurrencies are long-term appreciating assets that can be used as a hedge against inflation and the debasement of fiat currency.

The cryptocurrency market has grown rapidly over the last decade and has a market capitalization of $2 trillion, primarily driven by bitcoin which has a market capitalization of $1 trillion.[3,4] Ethereum, the second most valuable cryptocurrency, was introduced in 2013 and currently has a market capitalization of $378 billion.[4] Unlike bitcoin, ethereum's blockchain ledger natively supports the creation of digital assets (e.g., nonfungible tokens) using smart contracts which increases ethereum's utilization.

One of the largest cryptocurrency exchanges in the world is preparing to launch an ethereum-based NFT marketplace which will compete with major players in this space, including OpenSea which has traded over $10 billion in volume in 2021. However, they recently experienced a security breach which resulted in the loss of account funds for thousands of customers. The hackers carried out a large-scale phishing campaign that allowed them to access email addresses, passwords, and phone numbers associated with their customer's accounts. In addition, the phishing campaign and additional social engineering attacks allowed hackers to gain access to the customers' personal email accounts. Hackers also exploited a vulnerability in the company's account recovery process allowing them to bypass the company's SMS multifactor authentication (i.e., one-time password) system. Therefore, despite the growing excitement surrounding the cryptocurrency ecosystem, the occurrence of fraudulent activity may prevent mainstream adoption.

Currently, the cryptocurrency exchange has asked customers to take extra steps to ensure the safety of their accounts, including using a more secure method of multifactor authentication (i.e., hardware-based security keys or authenticator applications with time-based one-time passwords). However, the customer experience team acknowledges that this process places the burden of security even more on the customer than previously. There is an understanding that many will not take the extra security measures leaving them vulnerable to future security breaches. Likewise, the additional steps needed to buy and trade cryptocurrency and NFTs may detract from the customer experience for those that do adopt the extra security measures. The team understands that proactively providing a positive customer experience is essential for reducing customer churn and improving customer retention, especially as the space becomes more crowded. As such, the team would like to examine fraud prevention strategies that could potentially provide a frictionless customer experience, especially since the launch of their ethereum-based NFT platform is expected to result in millions of additional transactions per month.

Therefore, the fundamental basis of the fraud prevention approach presented herein is to utilize machine learning based strategies for differentiating nonfraudulent from fraudulent accounts on the ethereum blockchain network. In the context of the cryptocurrency exchange, this process would likely occur when two accounts are attempting to transact a sale or trade. If an account is deemed fraudulent, the transaction will be blocked. Since a common disadvantage of using machine learning based methods for fraud prevention is that it may increase the probability of blocking valid transactions for accounts that are nonfraudulent, thresholding based on the false positive rate will be used to assist with this issue.

Methods

The Ethereum Fraud Detection dataset was obtained from the Kaggle repository. Nonfraudulent accounts were collected using the Etherscan API. Fraudulent accounts were collected using the EtherScamDB API. EtherScamDB was developed in July 2017 to track fraudulent addresses connected to ethereum phishing and scamming campaigns. For each account, features were obtained using the Etherscan API. Prior to modeling, features that demonstrated statistically significant differences between the 2 target groups (i.e., nonfraudulent and fraudulent; $P < 0.05$) were selected for further analysis. In addition, features demonstrating multicollinearity were removed prior to modeling.

Feature vectors were normalized using the RobustScaler method from the scikit-learn preprocessing module. The RobustScaler method, typically selected for its robustness to outliers, scales each feature independently according to its quantile range. The dataset was randomly partitioned into a training and test set with 75% used for training and 25% used for testing. Grid search-based hyperparameter tuning was conducted to determine the optimal parameters for the following machine learning algorithms: random forest, gradient boosting, support vector machine, K nearest neighbor, naïve bayes, and logistic regression. The hyperparameter combination resulting in the highest 5-fold cross-validation ROC AUC score was determined for each algorithm. Threshold-based precision-recall curves were generated to examine trade-offs between precision and recall across different thresholds based on the false discovery rate. Lastly, standard and FDR-adjusted confusion matrices were constructed.

Results and Discussion

Exploratory Data Analysis

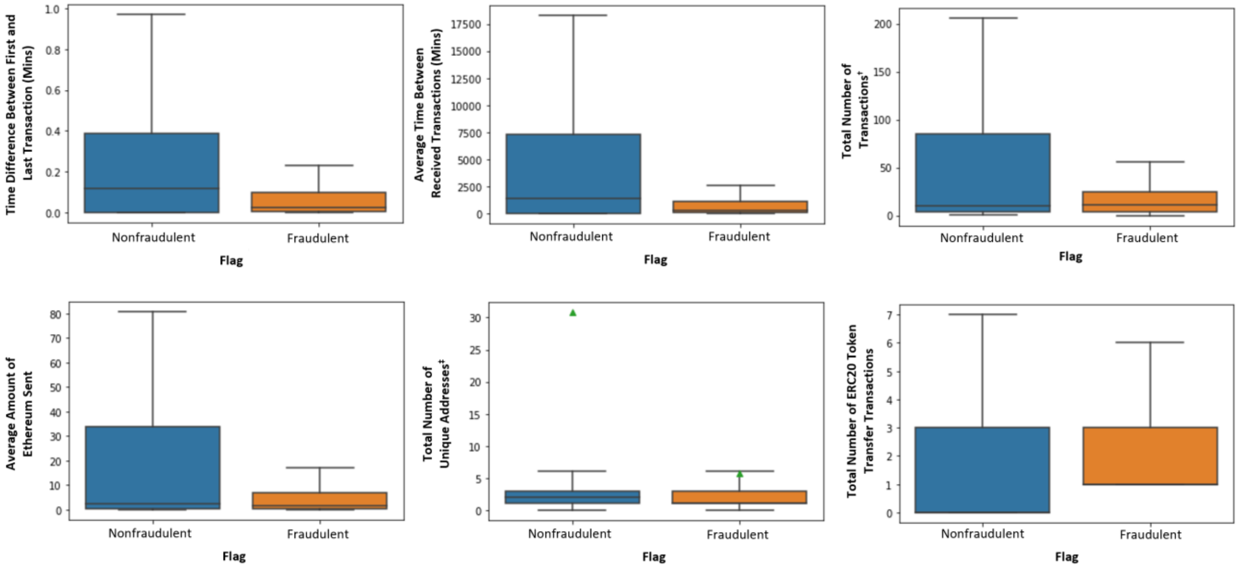
Of the 8,963 accounts included in the analysis, 7608 (87.3%) were nonfraudulent and 1,107 (12.7%) were fraudulent (**Supplemental Figure 1**). In total, 31 features were analyzed and there were statistically significant differences between the 2 target groups for 8 features: (1) average time between received transactions, (2) time difference between the first and last transaction, (3) total number of sent transactions, (4) total number of received transactions, (5) total number of unique addresses accounting for all outward-bound transactions, (6) average amount of ethereum sent, (7) total number of transactions, including contract development transactions, and (8) total number of ERC20 token transfer transactions (all $P < 0.05$).

Of the 8 statistically significant features, the total number of sent transactions and the total number of received transactions were removed due to strong multicollinearity: (1) total number of unique addresses accounting for all outward-bound transactions and total number of sent transactions (score: 0.68); (2) total number of sent transactions and total number of transactions, including contract development transactions (score: 0.73); (3) total number of received transactions and total number of transactions, including contract development transactions (score: 0.81).

Based on the exploratory data analysis, nonfraudulent accounts tend to have increased transactional activity compared to fraudulent accounts for all features except the total number of ERC20 token transactions (**Figure 1**). The decreased activity associated with fraudulent accounts can be attributed to the use of the accounts for solely fraudulent purposes. It's also possible that individuals carrying out fraudulent activities are more likely to have multiple accounts in order to evade detection.

The median total number of ERC20 transactions is higher for fraudulent accounts compared to nonfraudulent accounts. Of note, ERC20 is the most widely used ethereum token contract standard for fungible tokens. It’s essentially a set of guidelines all ethereum-based tokens must conform to so they can be utilized on the ethereum blockchain. There are hundreds of smart contract Ponzi schemes utilizing the ERC20 standard on the ethereum blockchain, most of which may originate from fraudulent accounts.[5]

Figure 1. Transaction Activity Differences Between Nonfraudulent and Nonfraudulent Accounts*



*$P < 0.05$ for all comparisons.

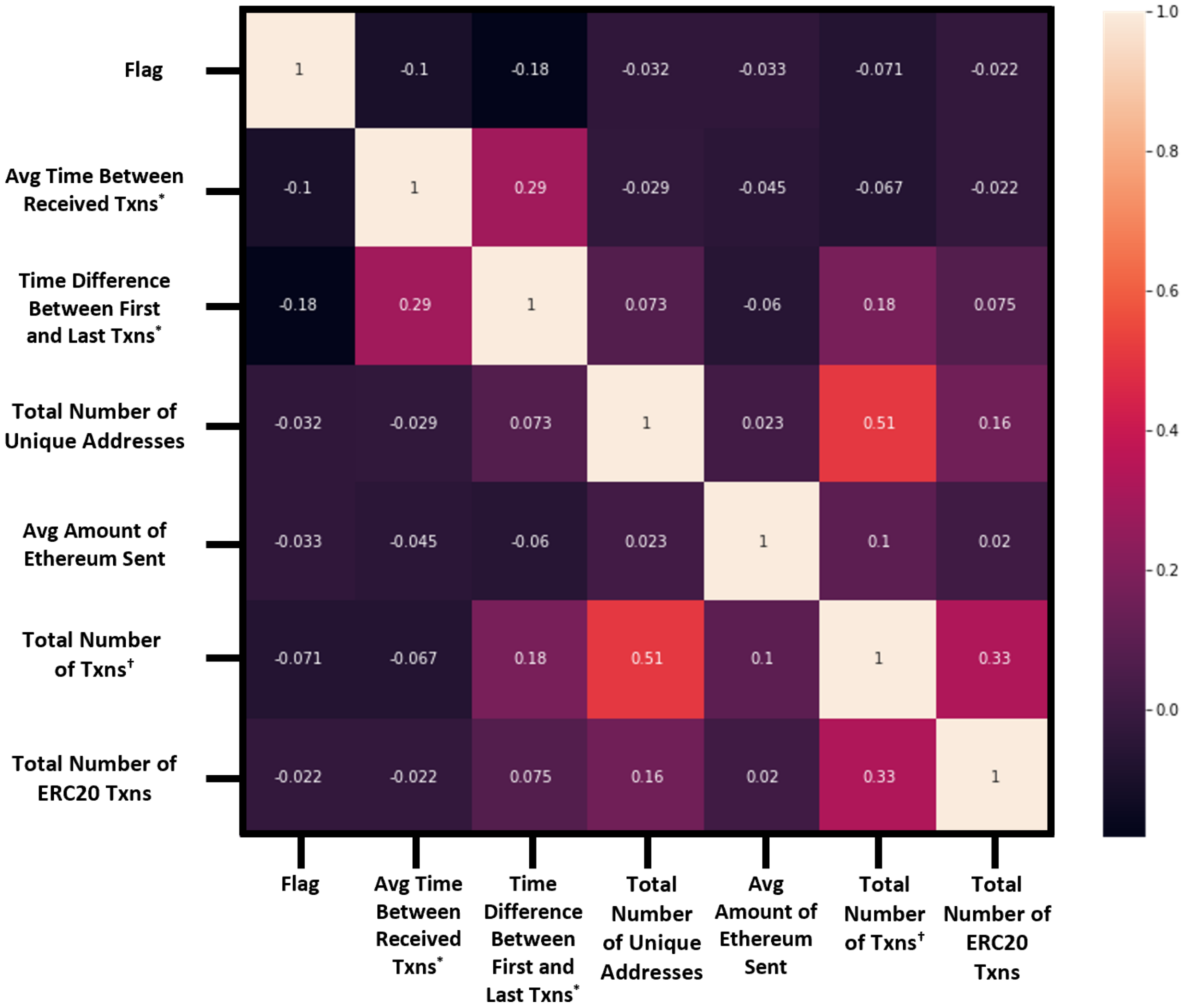
†Total number of transactions, including contract development transactions.

‡Mean number of unique addresses denoted by green triangle (nonfraudulent: 30.8; fraudulent: 5.7).

Feature Correlation Heatmap

The Pearson correlation matrix was visualized as a heatmap (**Figure 2**). The correlation between the target and the independent variables are as follows: (1) average time between received transactions (score: -0.1); (2) time difference between the first and last transaction on the account (score: -0.18); (3) total number of unique addresses accounting for all outward-bound transactions (score: -0.032); (4) average amount of ethereum sent (score: -0.033); (5) total number of transactions (score: -0.071), (6) total number of ERC20 token transfer transactions (score: -0.022).

Figure 2. Heatmap of Feature Correlation Coefficient Matrix



	Flag	Avg Time Between Received Txns*	Time Difference Between First and Last Txns*	Total Number of Unique Addresses	Avg Amount of Ethereum Sent	Total Number of Txns†	Total Number of ERC20 Txns
Flag	1	-0.1	-0.18	-0.032	-0.033	-0.071	-0.022
Avg Time Between Received Txns*	-0.1	1	0.29	-0.029	-0.045	-0.067	-0.022
Time Difference Between First and Last Txns*	-0.18	0.29	1	0.073	-0.06	0.18	0.075
Total Number of Unique Addresses	-0.032	-0.029	0.073	1	0.023	0.51	0.16
Avg Amount of Ethereum Sent	-0.033	-0.045	-0.06	0.023	1	0.1	0.02
Total Number of Txns†	-0.071	-0.067	0.18	0.51	0.1	1	0.33
Total Number of ERC20 Txns	-0.022	-0.022	0.075	0.16	0.02	0.33	1

Avg, average; Txns, transactions.

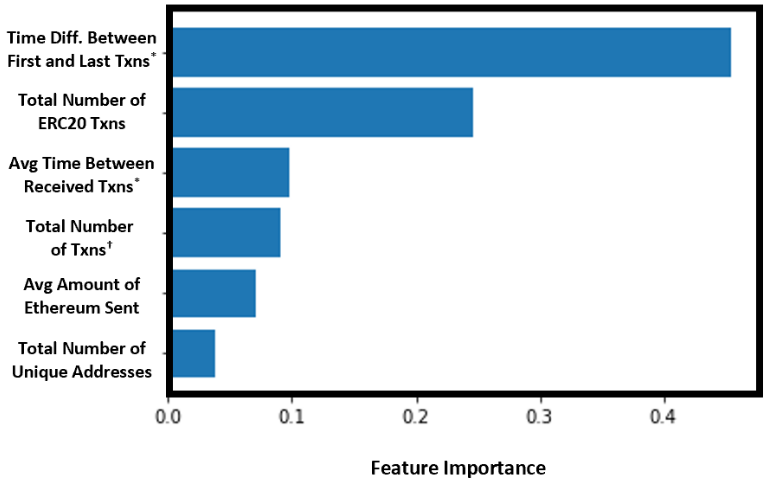
*Time in minutes

†Total number of transactions, including contract development transactions.

Feature Importance

Features were ranked by predictive performance (**Figure 3**). The three most important features include: (1) the time difference between the first and last transaction on the account, (2) total number of ERC20 token transfer transactions, and (3) average time between received transactions.

Figure 3: Feature Importance Analysis



Avg, average; Diff, difference; Txns, transactions.

*Time in minutes

†Total number of transactions, including contract development transactions.

Model Selection

The classification results and corresponding optimized hyperparameter values for the machine learning algorithms utilized in this study are shown in **Table 1**. The highest performing classifier was the random forest model with an ROC-AUC score of 0.9853. The gradient boosting model demonstrated a similar performance with an ROC-AUC score of 0.9852. The support vector classifier also performed very well and achieved an ROC-AUC score of 0.9538. The naïve bayes and logistic regression algorithms performed poorly relative to the other algorithms utilized in the study with ROC-AUC scores of 0.7442 and 0.7228, respectively.

Table 1. ROC-AUC Based Model Performance Comparison

Classifier	ROC-AUC Score	Hyperparameter Values

Random Forest	0.9853	criterion = entropy,
		max_depth = 100,
		max_features = sqrt,
		n_estimators = 100

Gradient Boosting	0.9852	learning_rate = 0.1,
		max_depth = 100,
		max_features = sqrt,
		n_estimators = 300

K Nearest Neighbor	0.9630	n_neighbors = 20,
		weights = distance,
		p = 1

Support Vector Machine	0.9538	C = 1,
		gamma = 1

Naïve Bayes	0.7442	var_smoothing = 0.1

Logistic Regression	0.7228	max_iter = 250,
		C = 0.001

ROC AUC, receiver operating characteristic area under curve.

Fraud Detection Model: Controlling the False Discovery Rate

The random forest model was selected for further analysis. Prior to thresholding by the false discovery rate, the random forest model resulted in the correct classification of 2092 (96%) accounts with 56 (2.6%) accounts incorrectly classified as nonfraudulent and 31 (1.4%) accounts incorrectly classified as fraudulent (**Supplemental Figure 2**). Precision and recall were 0.87 and 0.79, respectively.

Traditional financial institutions often have to choose between maximizing recall since false negatives are costly and maximizing precision which would assist in the prevention of inadvertently flagging nonfraudulent accounts that were misclassified as fraudulent. Maximizing recall may result in the minimization of payout caused by the inadvertent approval of transactions from fraudulent accounts, thereby resulting in an overall increase in profits. Maximizing precision would reduce the number of false positives. Therefore, the number of customers with nonfraudulent accounts incorrectly flagged as fraudulent would decrease; thereby, decreasing the number of customers inconvenienced by the misclassification. Therefore, precision and recall were calculated with different thresholds based on false discovery rates set to 2% and 5%, respectively.

Setting the false discovery rate to 2% resulted in the correct classification of 2098 (96%) accounts (**Supplemental Figure 3**). In total, 12 (0.5%) accounts were incorrectly classified as nonfraudulent and 69 (3.2%) accounts were incorrectly classified as fraudulent. Precision and recall were 0.94 and 0.74, respectively. Increasing the threshold from 2% to 5% resulted in the correct classification of 2021 (92.7%) accounts (**Supplemental Figure 4**). In total, 146 (7%) accounts were incorrectly classified as nonfraudulent and 12 (0.5%) accounts were incorrectly classified as fraudulent. Precision and recall were 0.64 and 0.96, respectively.

Conclusion

The utilization of a machine learning based fraud detection system would significantly improve the customer experience by shifting the burden of security from the customer to the company. Additionally, this will not only improve the customer experience but potentially improve customer acquisition, retention, and overall profitability. Also, traditional financial institutions are entering the cryptocurrency space at a rapid rate. These institutions already have robust security measures in place and do not solely rely on onerous multifactor authentication measures to keep their customers safe. These institutions will quickly become fierce competitors. As such, centralized cryptocurrency exchanges will need to adopt more sophisticated security processes to remain competitive.

In this study, the random forest classifier outperformed the gradient boosting, K nearest neighbor, support vector machine, naïve bayes, and logistic regression classifiers. However, prior to choosing the optimal detection threshold the client will need to determine if it is more beneficial to select a threshold that would minimize payout due to fraud or improve the customer experience by selecting a threshold that minimizes false positives. If the client is more concerned with minimizing payout, the optimal detection threshold would be based on a false discovery rate of 5%. If the client is more concerned with decreasing the number of false positives, the optimal detection threshold would be based on a false discovery rate of 2%.

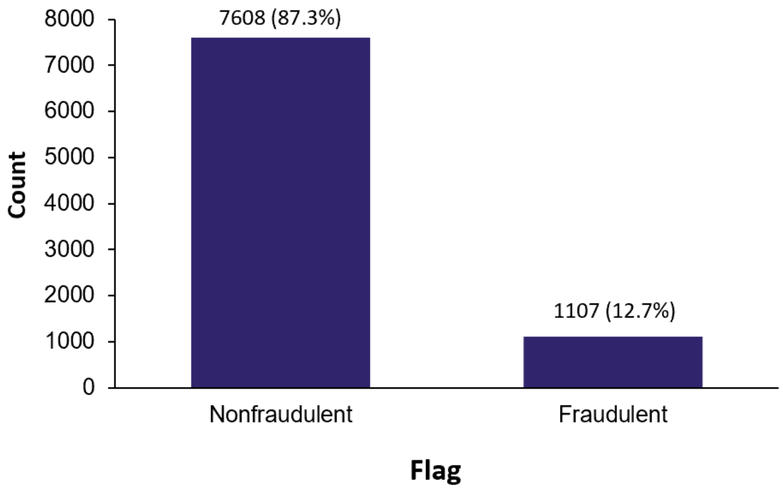
Additionally, it is important to note that the optimal fraud management solution will need to minimize the total cost of fraud (i.e., financial loss due to fraud and fraud prevention costs). Therefore, the company will need to determine the ideal level of review that would keep fraud losses under control. Lastly, as cryptocurrencies and NFTs on other blockchains become more popular (i.e., Tezos, Flow, Solana, Cardano), the fraud prevention strategy may need to evolve based on the data available on other blockchains.

References

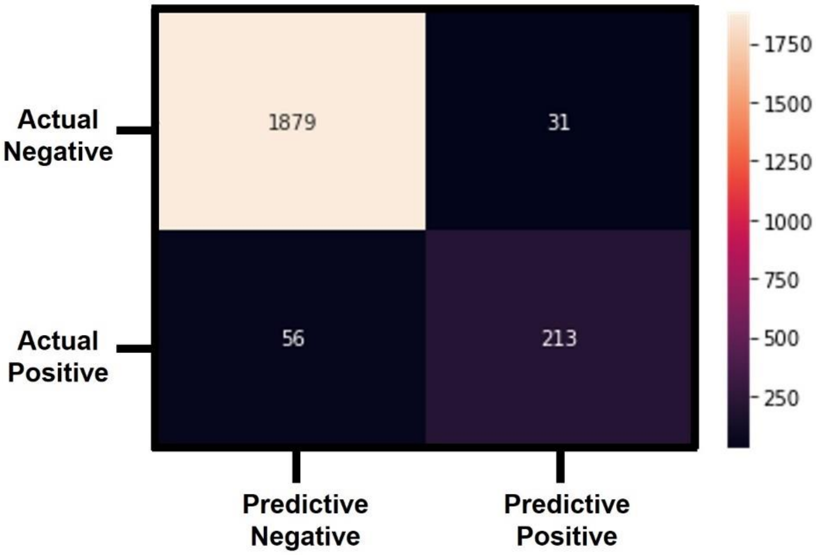
1. Bănărescu A. Detecting and preventing fraud with data analytics. *Procedia economics and finance*. 2015 Jan 1;32:1827-36.
2. Syed AM, Moge JA, Siddiqui MS. Cryptocurrency: Next Level in the Evolution of Money. Asian Journal of Research in Business Economics and Management. 2016;6(11):53-63.
3. Smutny Z, Sulc Z, Lansky J. Motivations, Barriers and Risk-Taking When Investing in Cryptocurrencies. *Mathematics*. 2021 Jan;9(14):1655.
4. Cryptocurrency Market Capitalization. https://coinmarketcap.com/. Accessed on October 20, 2021.
5. Chen W, Zheng Z, Ngai EC, Zheng P, Zhou Y. Exploiting blockchain data to detect smart ponzi schemes on ethereum. *IEEE Access*. 2019 Mar 18;7:37575-86.

Supplemental Appendix

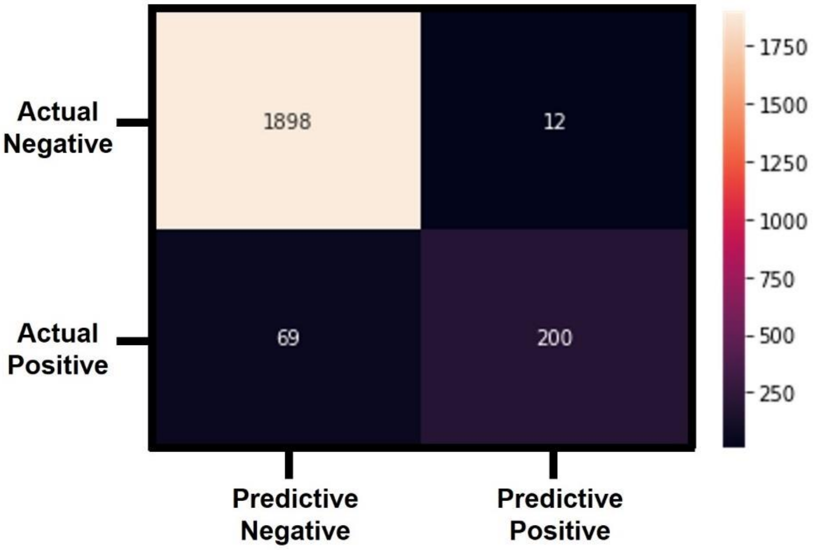
Supplemental Figure 1. Univariate Distribution of the Target Variable



Supplemental Figure 2. Predictive Performance of Optimized Random Forest Classifier

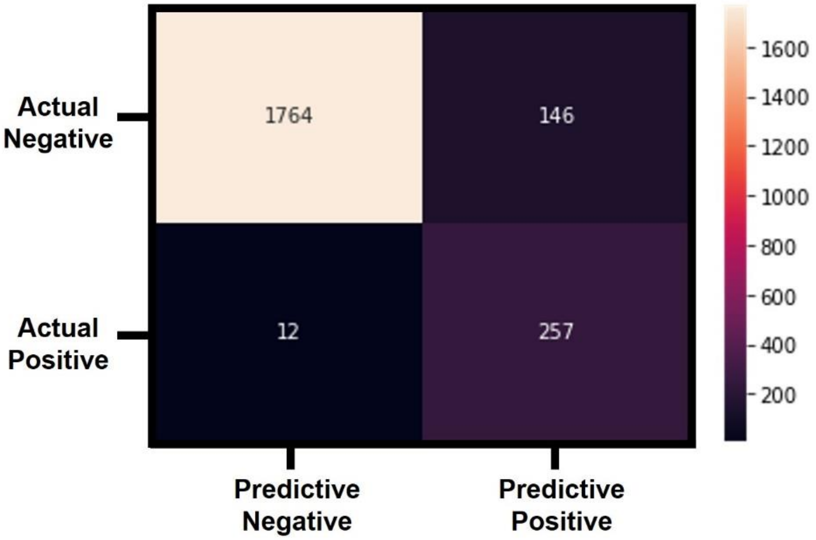


Supplemental Figure 3. Predictive Performance of Optimized Random Forest Classifier (FDR = 0.02)



FDR, false discovery rate.

Supplemental Figure 4. Predictive Performance of Optimized Random Forest Classifier (FDR = 0.05)



FDR, false discovery rate.

Supplemental Table 1. False Discovery Rate Based Thresholds for Precision and Recall

Threshold	Precision	Recall	FDR
0.00	0.123	1.000	0.099
0.01	0.315	1.000	0.094
0.02	0.389	0.996	0.089
0.03	0.439	0.996	0.083
0.04	0.476	0.996	0.080
0.05	0.499	0.993	0.077
0.06	0.521	0.993	0.070
0.07	0.528	0.978	0.065
0.08	0.553	0.978	0.064
0.09	0.563	0.974	0.062
0.10	0.578	0.974	0.060
0.11	0.597	0.970	0.056
0.12	0.611	0.963	0.054
0.13	0.619	0.959	0.051
0.14	0.631	0.959	0.050
0.15	0.638	0.955	0.049
0.16	0.655	0.952	0.048
0.17	0.661	0.952	0.048
0.18	0.665	0.952	0.047
0.19	0.674	0.944	0.046
0.20	0.687	0.937	0.044
0.21	0.699	0.933	0.044
0.22	0.709	0.933	0.043
0.23	0.712	0.929	0.042
0.24	0.720	0.926	0.042
0.25	0.724	0.918	0.040
0.26	0.731	0.907	0.038
0.27	0.739	0.903	0.037
0.28	0.748	0.903	0.037
0.29	0.753	0.896	0.037
0.30	0.761	0.888	0.035
0.31	0.765	0.885	0.035
0.32	0.771	0.877	0.035
0.33	0.783	0.870	0.035
0.34	0.790	0.866	0.035

FDR, false discovery rate

Supplemental Table 1. False Discovery Rate Based Thresholds for Precision and Recall (cont.)

Threshold	Precision	Recall	FDR
0.35	0.796	0.855	0.035
0.36	0.799	0.855	0.034
0.37	0.806	0.848	0.034
0.38	0.808	0.844	0.033
0.39	0.815	0.836	0.033
0.40	0.824	0.836	0.032
0.41	0.832	0.829	0.032
0.42	0.835	0.829	0.032
0.43	0.838	0.825	0.032
0.44	0.838	0.825	0.031
0.45	0.844	0.822	0.031
0.46	0.853	0.818	0.030
0.47	0.858	0.807	0.030
0.48	0.858	0.807	0.029
0.49	0.863	0.799	0.028
0.50	0.867	0.799	0.028
0.51	0.870	0.796	0.028
0.52	0.873	0.796	0.028
0.53	0.881	0.796	0.028
0.54	0.884	0.796	0.028
0.55	0.884	0.792	0.027
0.56	0.887	0.784	0.027
0.57	0.894	0.781	0.026
0.58	0.894	0.781	0.025
0.59	0.897	0.777	0.025
0.60	0.904	0.770	0.025
0.61	0.908	0.770	0.024
0.62	0.928	0.770	0.024
0.63	0.928	0.762	0.023
0.64	0.932	0.758	0.023
0.65	0.935	0.755	0.022
0.66	0.939	0.747	0.022
0.67	0.939	0.743	0.021
0.68	0.943	0.743	0.020
0.69	0.947	0.732	0.019

FDR, false discovery rate

Supplemental Table 1. False Discovery Rate Based Thresholds for Precision and Recall (cont.)

Threshold	Precision	Recall	FDR
0.70	0.952	0.732	0.019
0.71	0.952	0.732	0.017
0.72	0.951	0.729	0.016
0.73	0.951	0.717	0.015
0.74	0.950	0.706	0.014
0.75	0.949	0.691	0.014
0.76	0.954	0.688	0.014
0.77	0.953	0.684	0.012
0.78	0.953	0.677	0.011
0.79	0.953	0.677	0.010
0.80	0.952	0.662	0.010
0.81	0.951	0.654	0.009
0.82	0.951	0.647	0.008
0.83	0.950	0.636	0.007
0.84	0.950	0.636	0.007
0.85	0.955	0.625	0.007
0.86	0.954	0.617	0.006
0.87	0.964	0.595	0.006
0.88	0.975	0.580	0.006
0.89	0.974	0.550	0.006
0.90	0.973	0.532	0.005
0.91	0.986	0.517	0.004
0.92	0.993	0.506	0.004
0.93	1.000	0.468	0.004
0.94	1.000	0.387	0.004
0.95	1.000	0.353	0.004
0.96	1.000	0.353	0.001
0.97	1.000	0.309	0.001
0.98	1.000	0.264	0.001
0.99	1.000	0.219	0.001
1.00	1.000	0.108	0.001

FDR, false discovery rate