Fraud Detection

Supervised Learning Approaches to Predicting Fraudulent Merchants

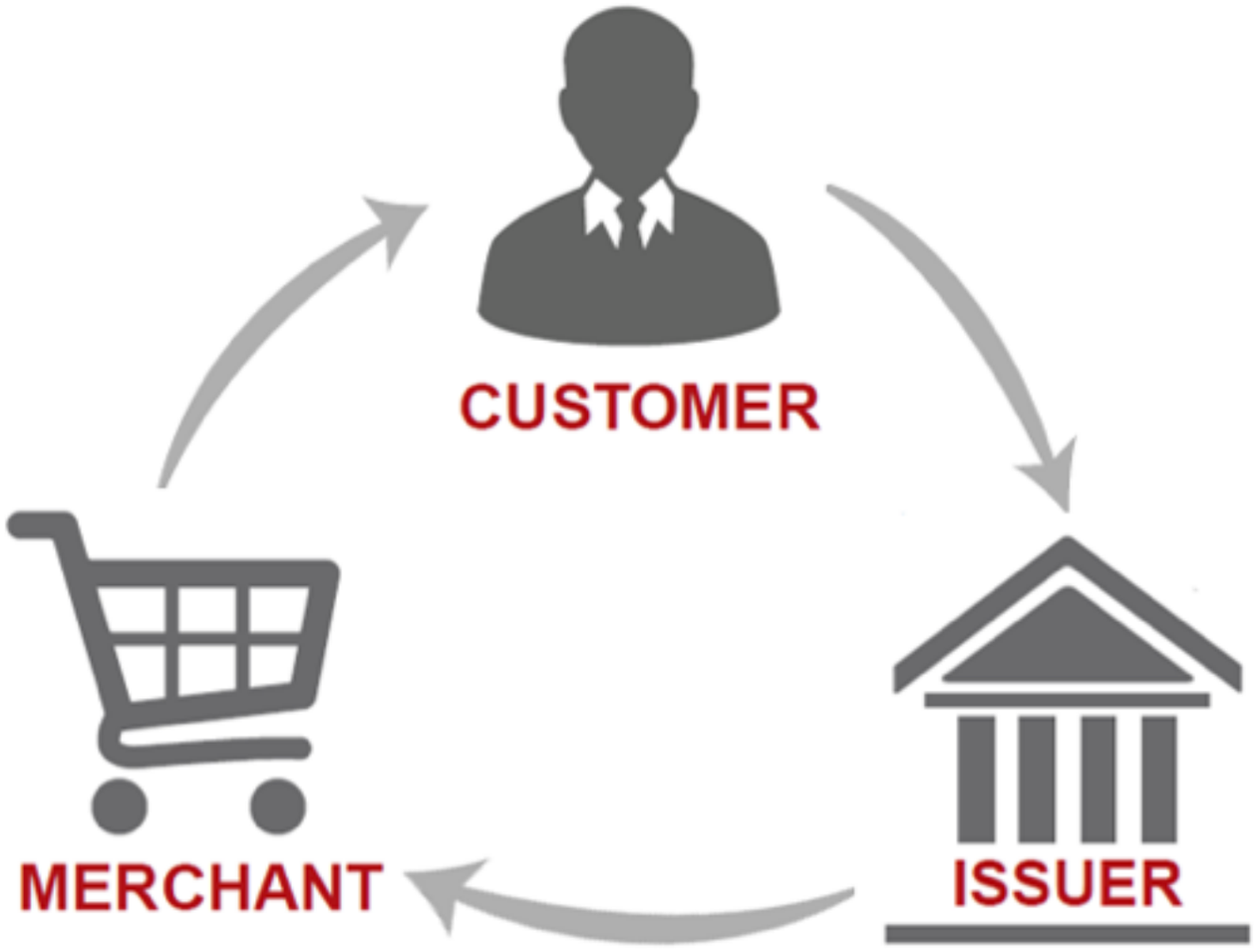
Merchant Fraud

- Apply for merchant account with payroll/other services provider; E.g. ADP, Intuit.

- E.g. an online store and generates lots of transactions from unsuspecting customers before busting out with the cash in their account

- E.g. fake ticketing websites, where duped customers don't realize it's a scam until their event ticket doesn't come in the mail

- E.g. fake online store and then use fraudulent payment methods such as cloned cards to 'buy' goods



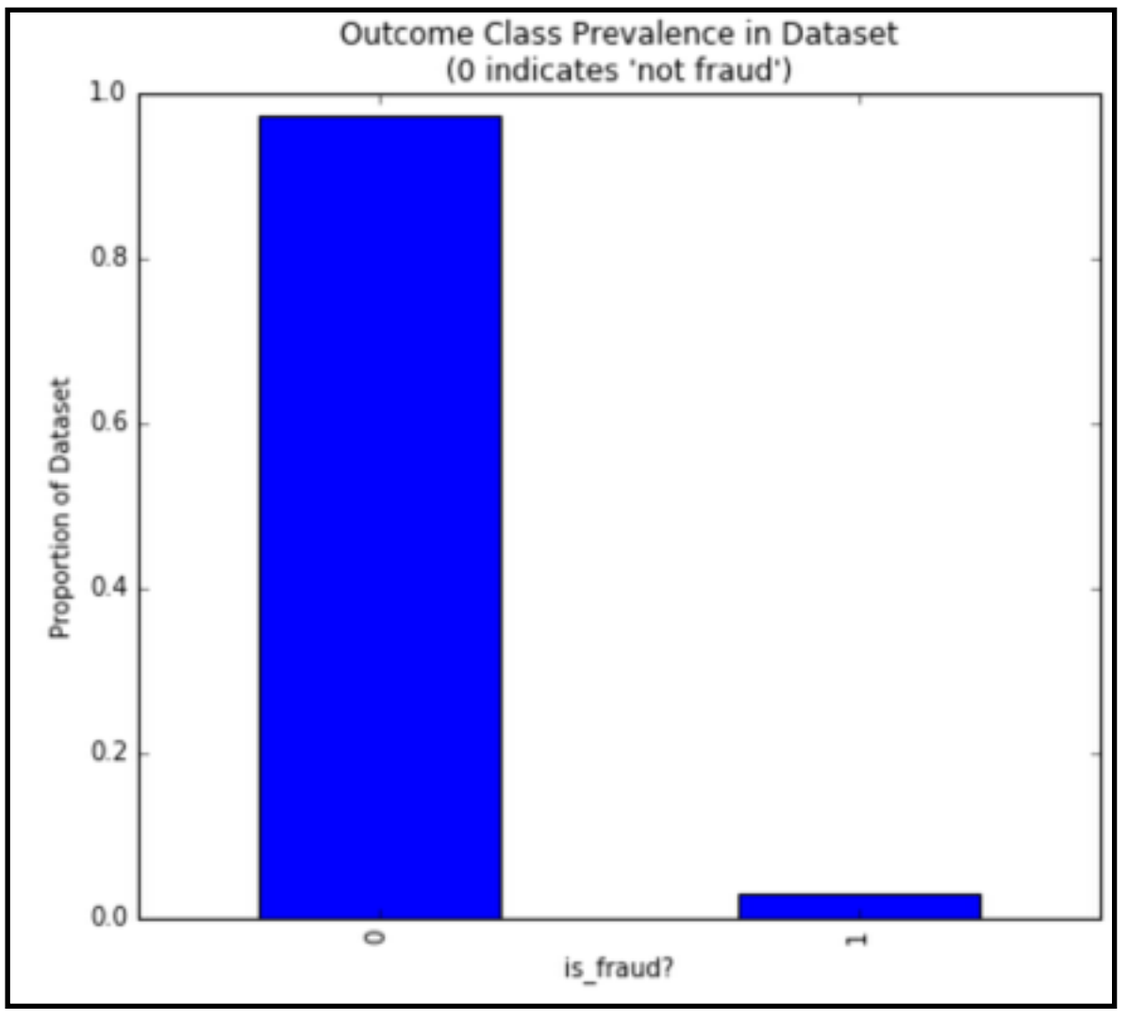
Goal

Assess the risk that an account will be closed for fraud.

Specifically, we want you to identify the 100 merchants opened that you consider to be of greatest risk for being closed for fraud.

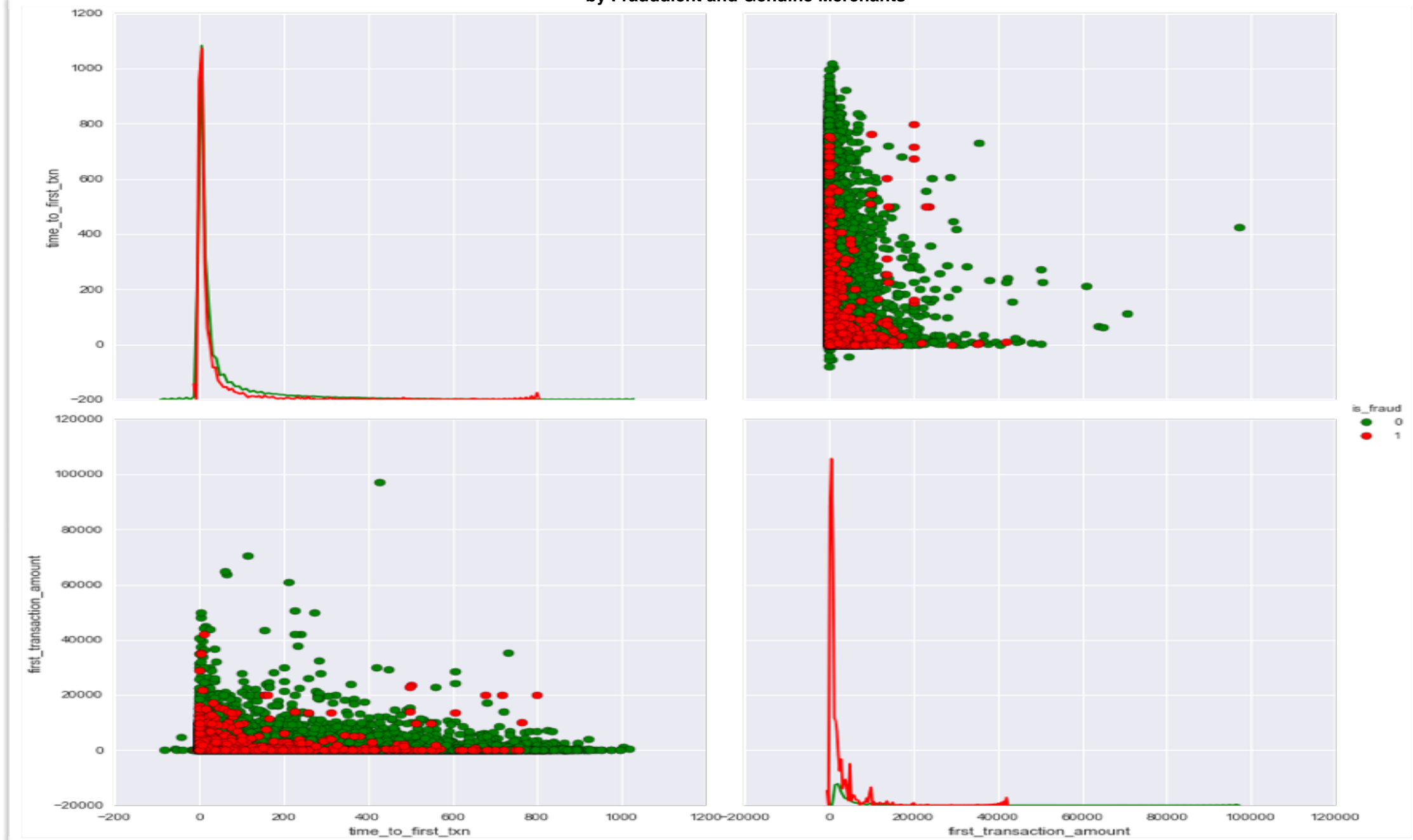
Dataset

- 130,888 Records
- 9 Merchant Application Features
- 2 First Transaction Features
- *Classified* into:
  - Genuine Merchants (97.12%)
  - Fraudulent Merchants (2.88%)



Missing / Erroneous Values

- IP Address Organization has only 4 missing
- IP/Pierced IP Location have approx. **4.2%** missing
  - Fraudulent merchant prevalence in missing locations is negligible: *approx.* ***0.05%*** *of merchants from unknown locations are fraudulent*
  - Label all missing locations as *Unknown*
- First transaction dates are erroneously entered as prior to merchant application date, or vice-versa. Impute *Time to First Transaction* with median value of Organization + Location



Time to First Transaction, First Transaction Amount

- *First Transaction Amount* seems to have a lot more cases of fraud (much higher density) at smaller transaction amounts.
- *Time to First Transaction* has erroneous values. **Imputed** using *median* of Organization + Location

First Transaction Amount (closer look)

- Do whole number amounts see more fraud activity overall?

  - 70.46% of fraudulent merchants have whole number first transactions

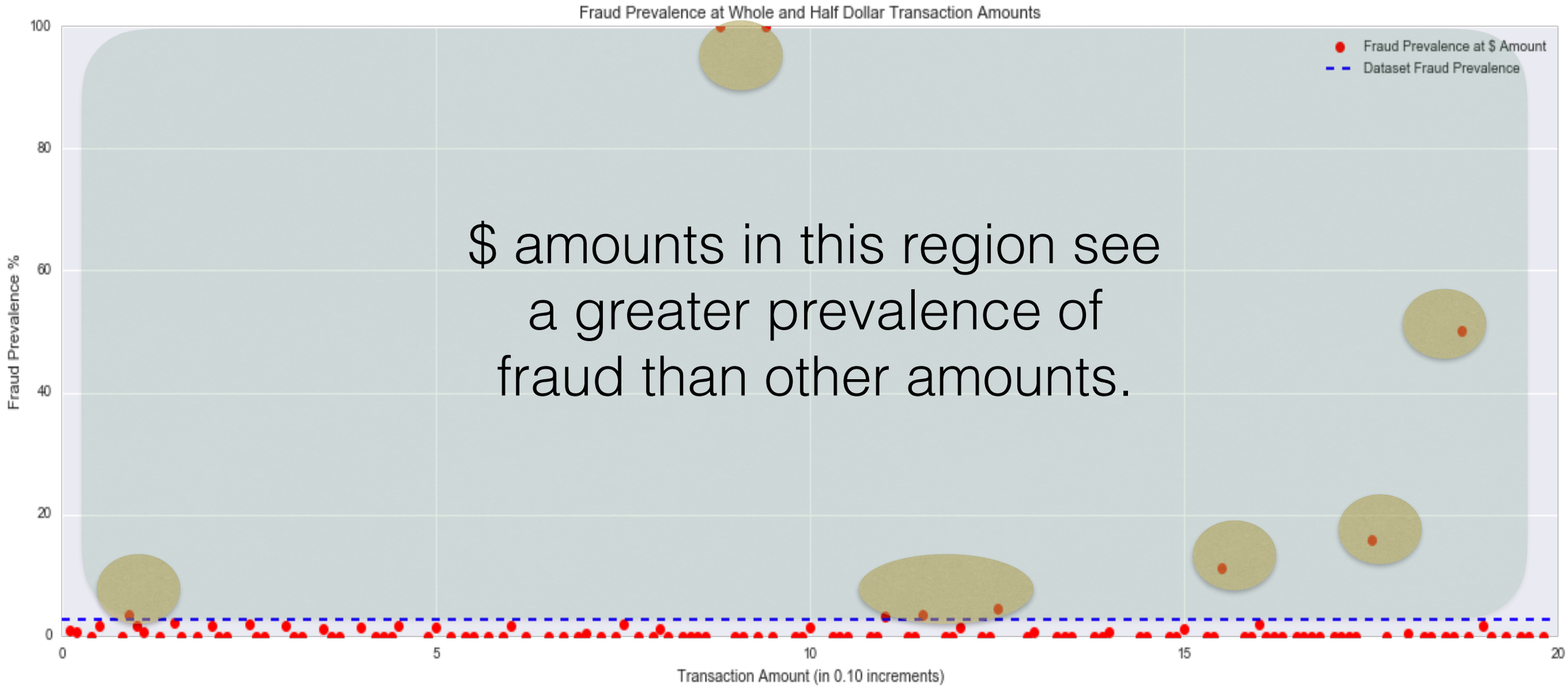
  - 70.53% of genuine cases have whole number first transactions

Not really.

	is_whole
is_fraud	
0	89661
1	2657

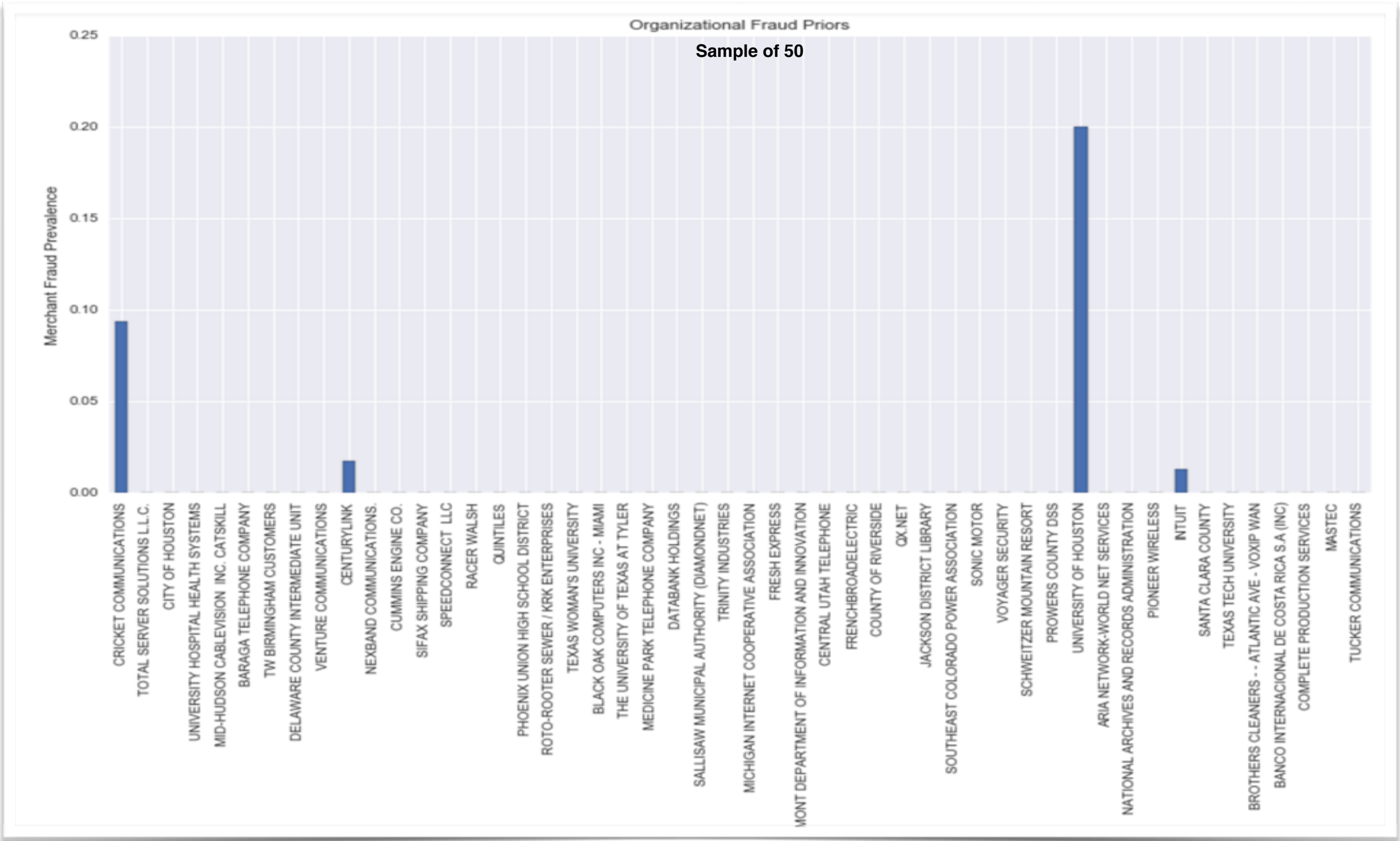
First Transaction Amount

Fraud Prevalence for certain $ amounts



Organizational Fraud Prevalence

How much do we know about fraudulent merchant subsidiaries of organizations (*priors*)?



Location Homogeneity Coefficient

A *Jaccard Similarity* based measure of the homogeneity of location signatures compared to application provided location.

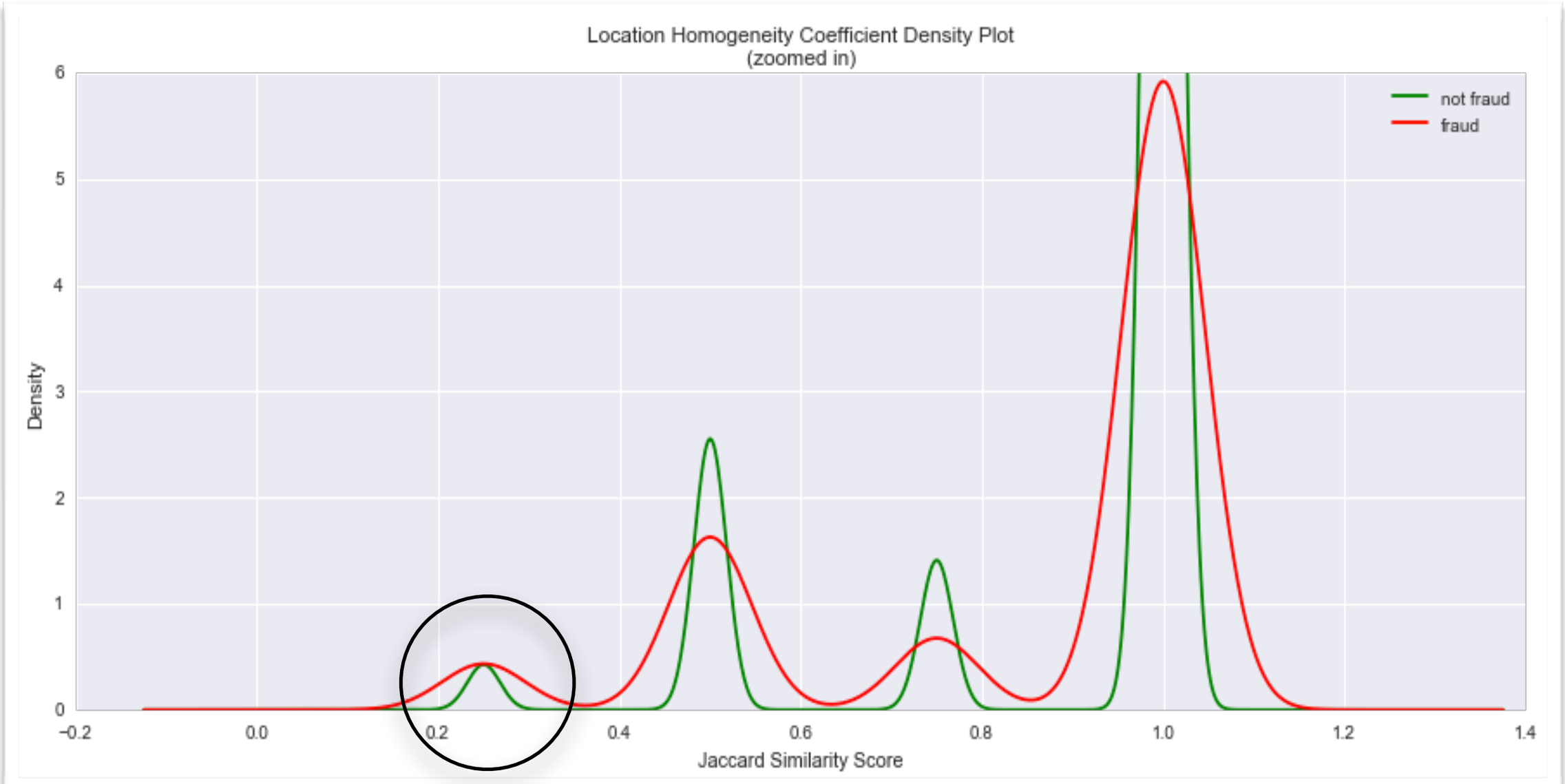
- Baseline a homogenous location structure from the Merchant application provided location, across 5 dimensions:
  e.g. *A = [Texas, Texas, Texas, Texas, Texas]*

- Compare actual location signature across 5 location dimensions:
  e.g. *B = [Texas, Texas, Alabama, Montana, Texas]*

- Jaccard Similarity Score = |A*int*B| / |AUB| = 3/5 = 0.6

Location Homogeneity Coefficient

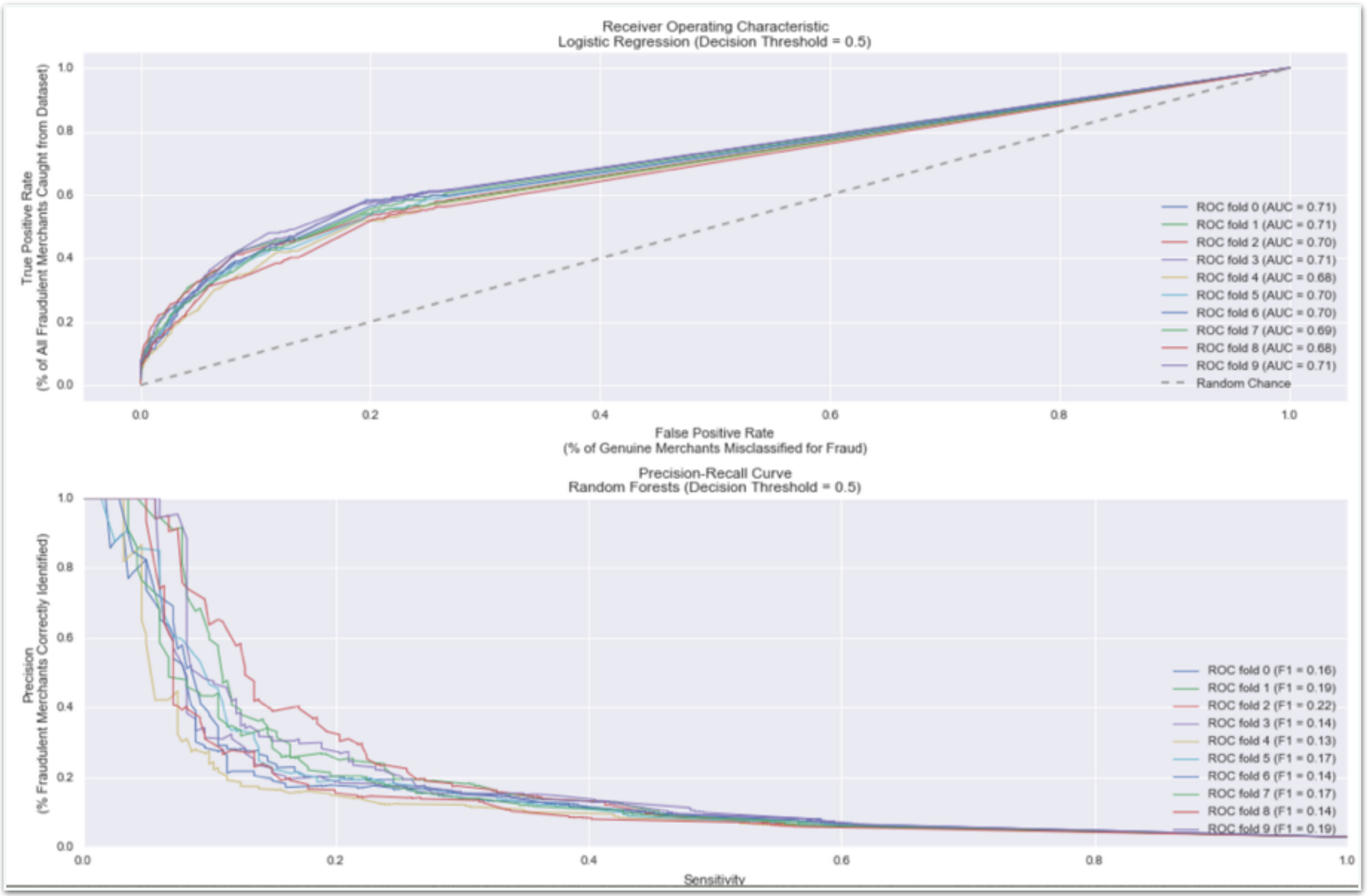
Higher density of fraudulent merchants at least homogenous location signatures.



Learning from Extremely Imbalanced Data

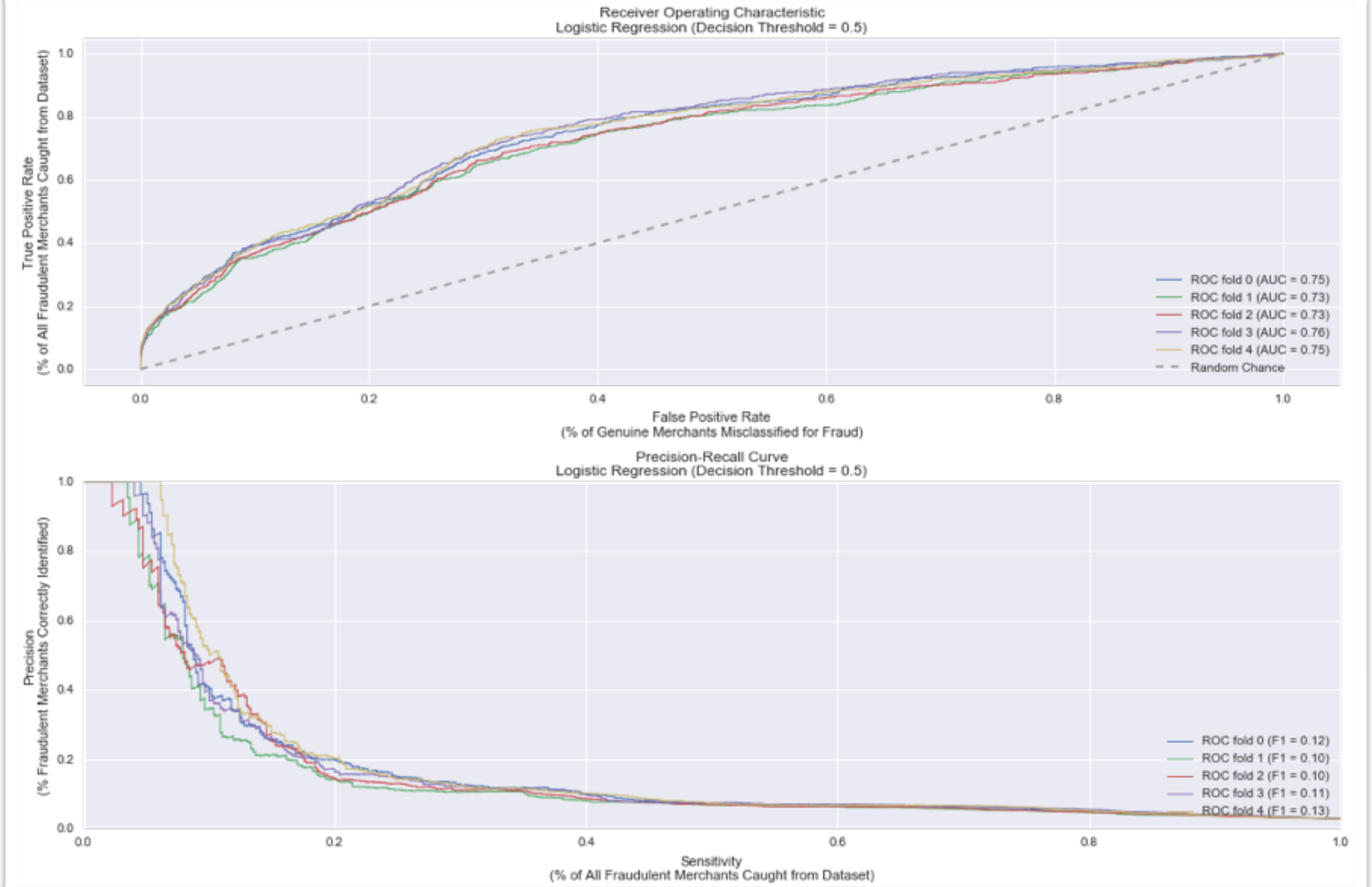
Part 1, Cost Driven Thresholding:

- Cross Validate Classifier on Imbalanced Data
- Tune Decision Threshold to Maximize Precision-Sensitivity (F1 score)
- Predict on Holdout Test Set using Tuned Decision Threshold



Classifying with Random Forests: DT = 0.5

	precision	recall	f1-score	support
fraud	0.40	0.12	0.18	961
not fraud	0.97	0.99	0.98	31761
avg / total	0.96	0.97	0.96	32722

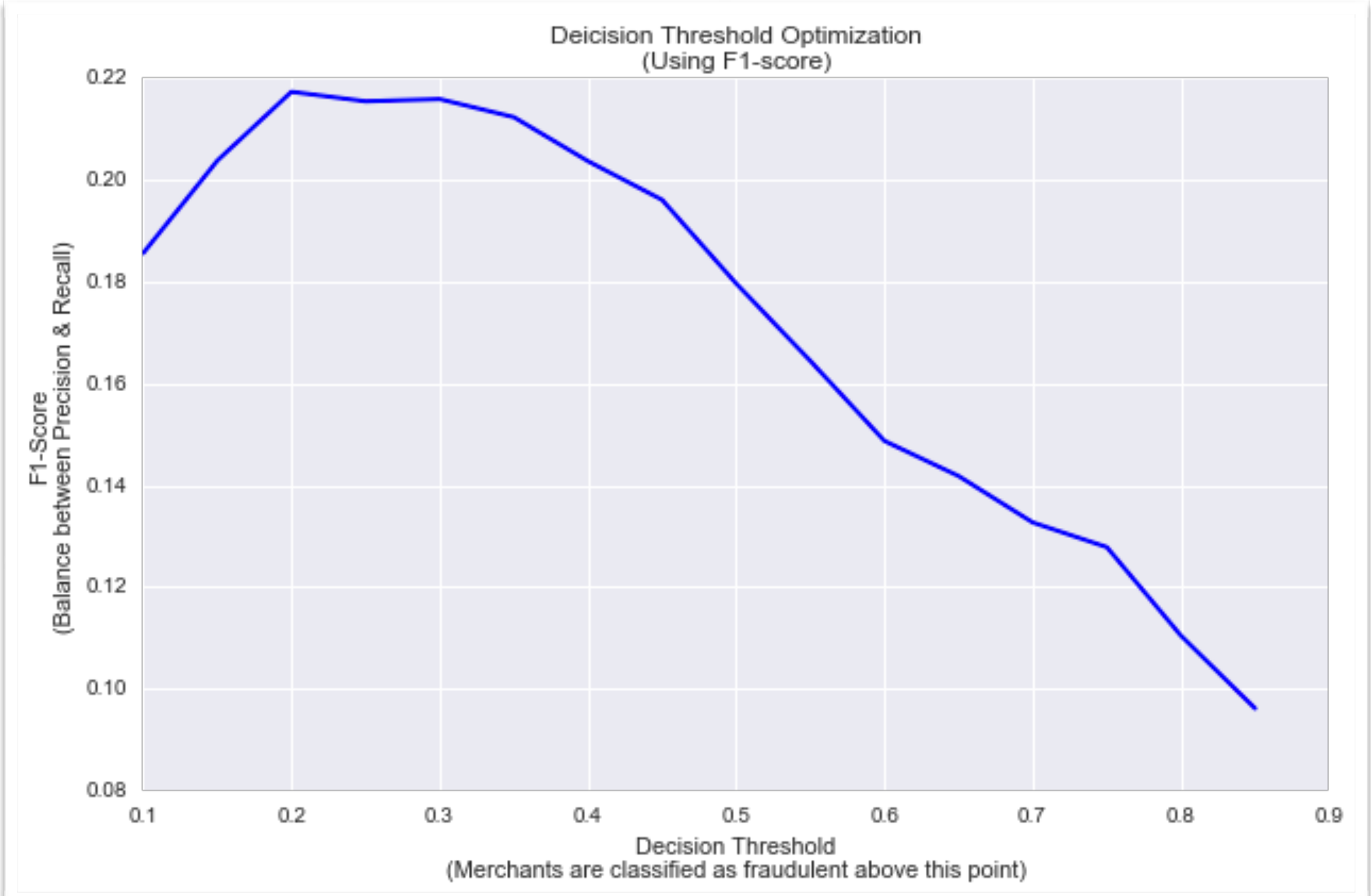


Classifying with Logistic Regression: DT = 0.5

	precision	recall	f1-score	support
fraud	0.64	0.05	0.10	922
not fraud	0.97	1.00	0.99	31800
avg / total	0.96	0.97	0.96	32722

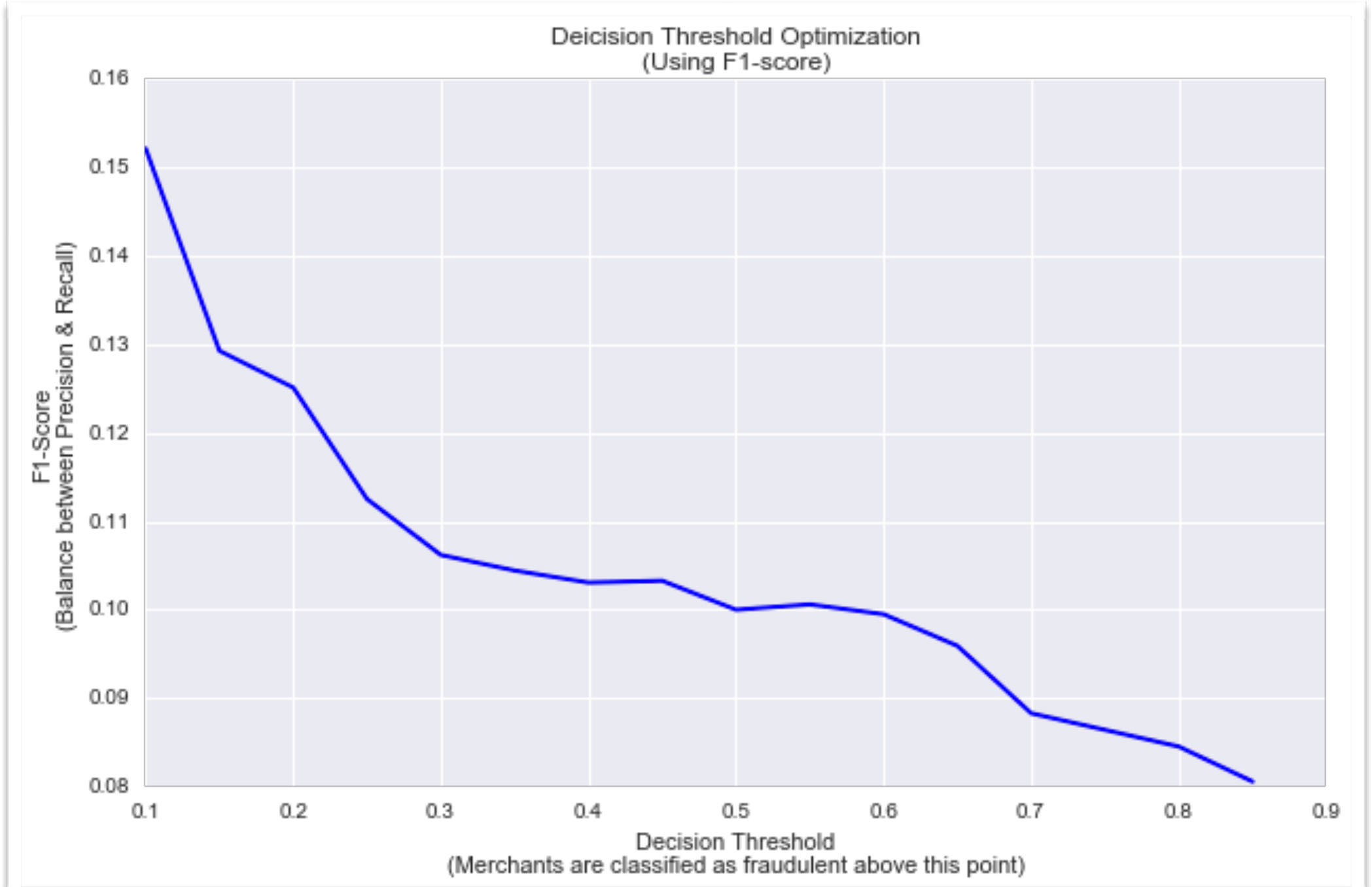
Optimizing Decision Threshold

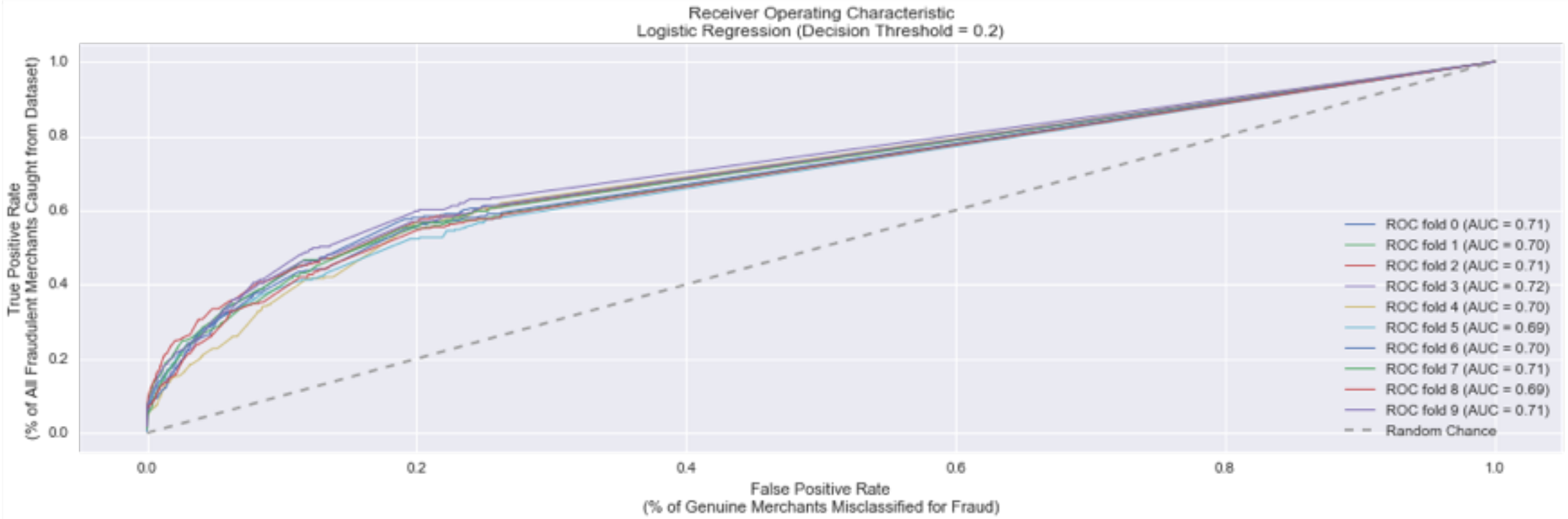
Optimal DT for Random Forest ~ 0.2 to 0.3

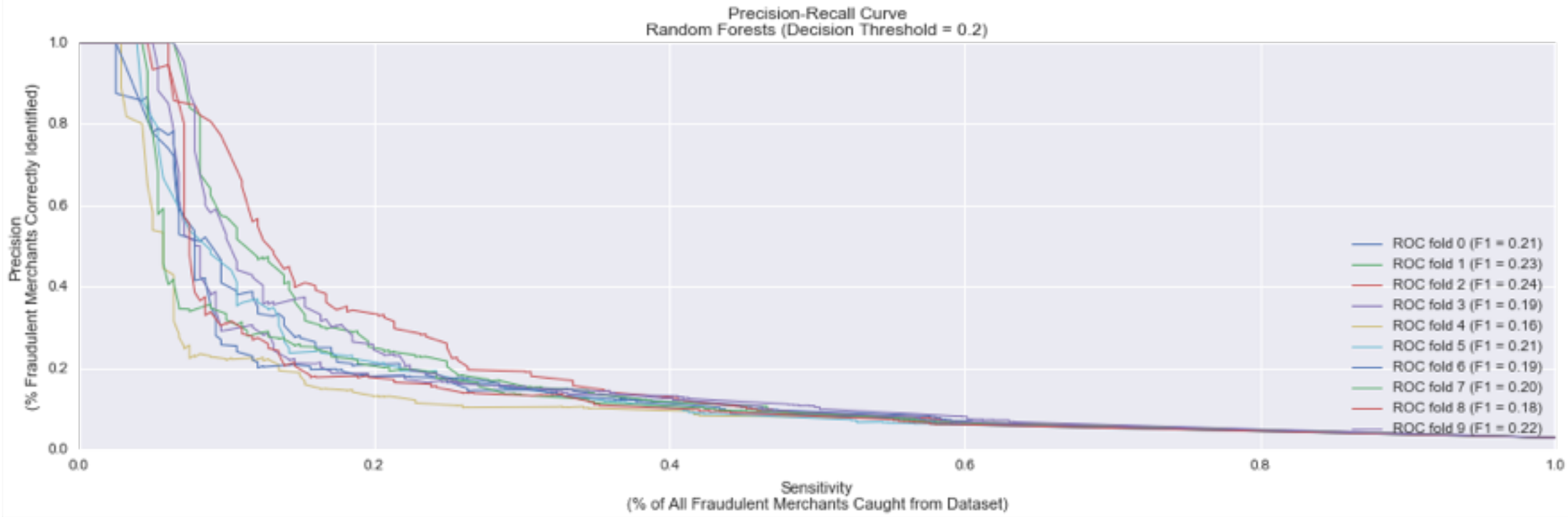


Optimizing Decision Threshold

Optimal DT for Logistic Regression ~ 0.1





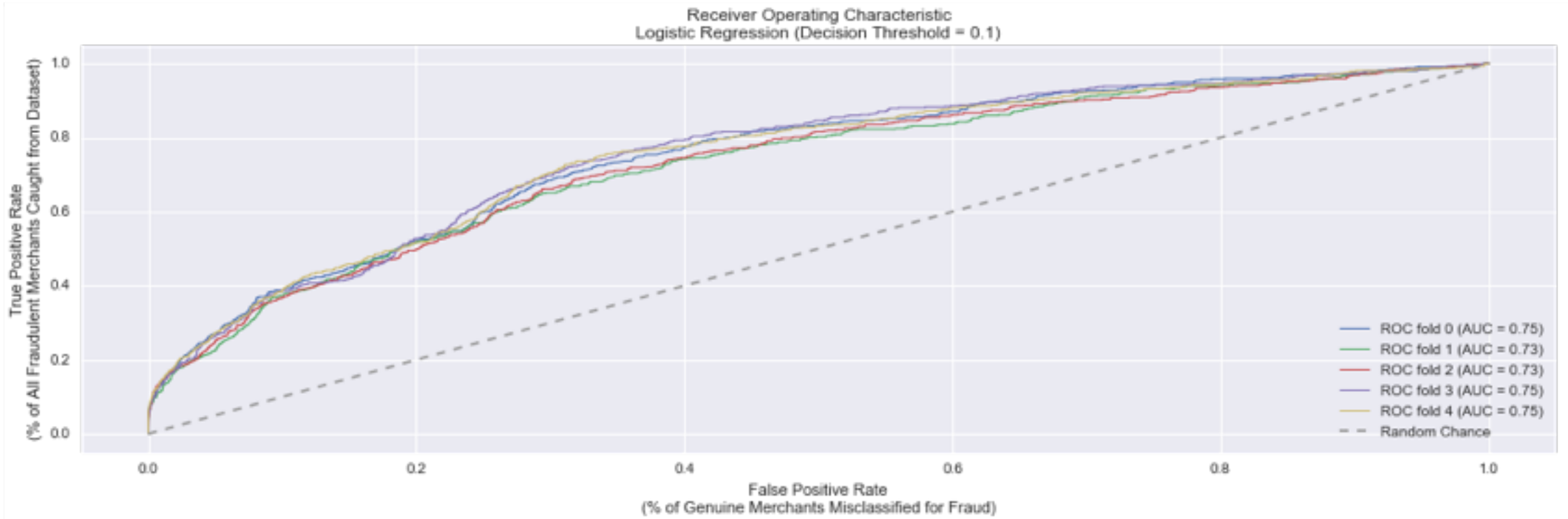


Classifying with RF: DT = 0.5

	precision	recall	f1-score	support
fraud	0.40	0.12	0.18	961
not fraud	0.97	0.99	0.98	31761
avg / total	0.96	0.97	0.96	32722

Classifying with RF: DT = 0.25

	precision	recall	f1-score	support
Fraud	0.21	0.22	0.22	961
Not Fraud	0.98	0.98	0.98	31761
avg / total	0.95	0.95	0.95	32722



Classifying with LReg.: DT = 0.5

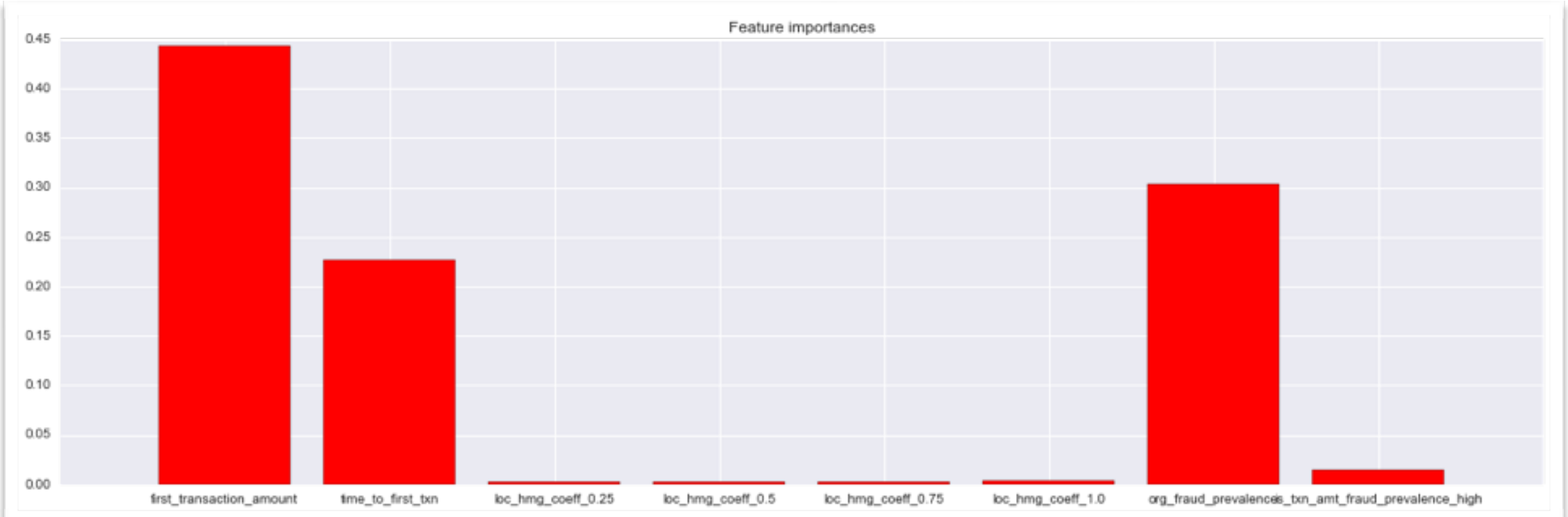
	precision	recall	f1-score	support
fraud	0.64	0.05	0.10	922
not fraud	0.97	1.00	0.99	31800
avg / total	0.96	0.97	0.96	32722

Classifying with LReg.: DT = 0.1

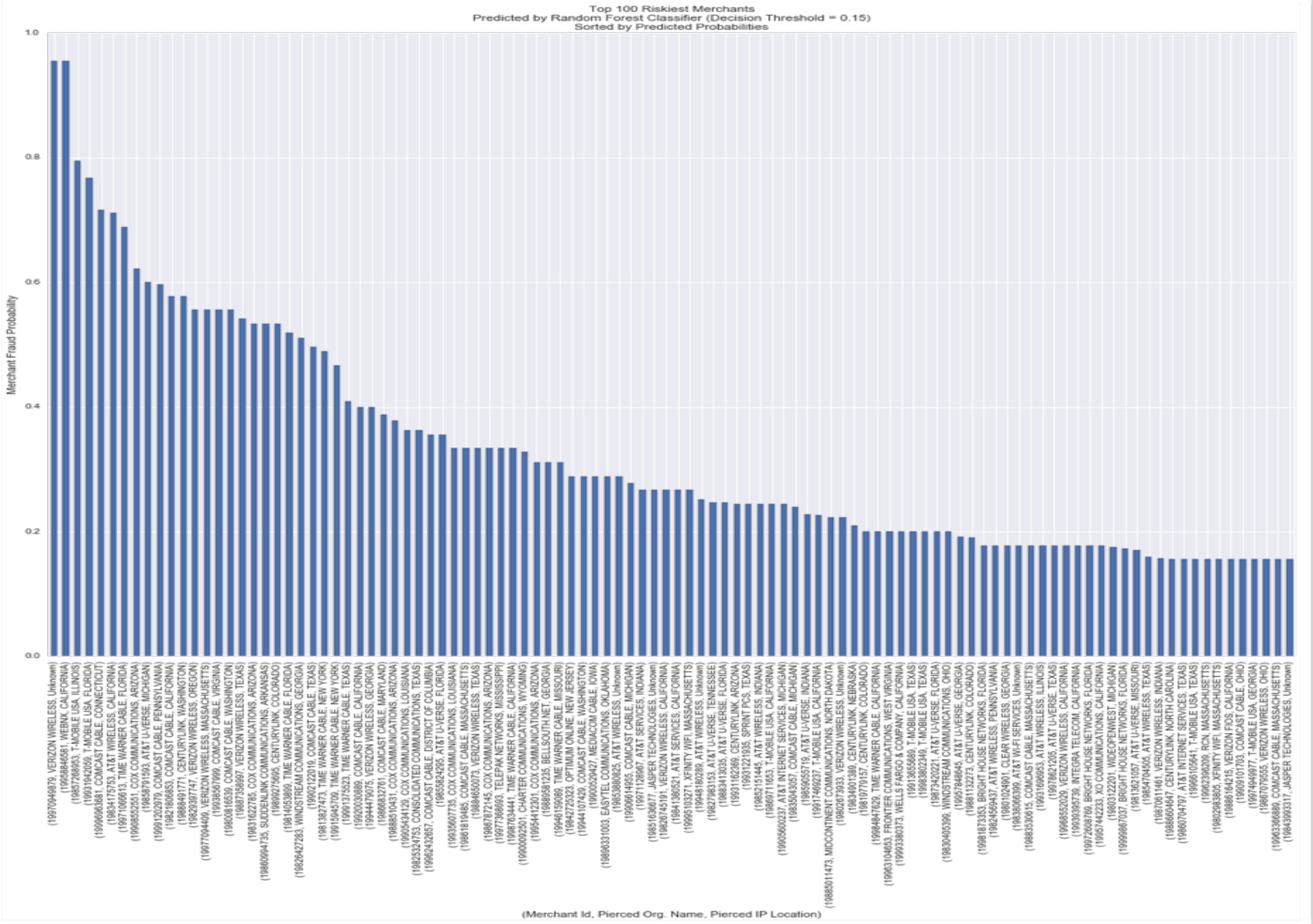
	precision	recall	f1-score	support
fraud	0.36	0.10	0.15	922
not fraud	0.97	0.99	0.98	31800
avg / total	0.96	0.97	0.96	32722

Recommendations and Important Features

	ML Algorithm	Decision Threshold
Business Priority: Maximize Fraud Penetration	**Random Forests**	**0.2 to 0.3**
Business Priority: Maximize Fraud Identification	**Logistic Regression**	**0.1**



Top 100 Riskiest Merchants



Learning from Extremely Imbalanced Data

Part 2, Balanced Sampling (Over/Under Sampling):

- Classifier constructs balanced trees as it makes decisions (equal representation of both classes)
- Sample Weights Formula:
    - *[Class Weights] = # samples / # of classes * [class frequencies]*
- Problem: Cross validation is not representative of real world data
- Validate on **Imbalanced** Hold Out Set
- Compare Against Original Run on Imbalanced Dataset

Weighted Sampling (Over/Undersampling)
Cross validation reveals misleadingly good results!

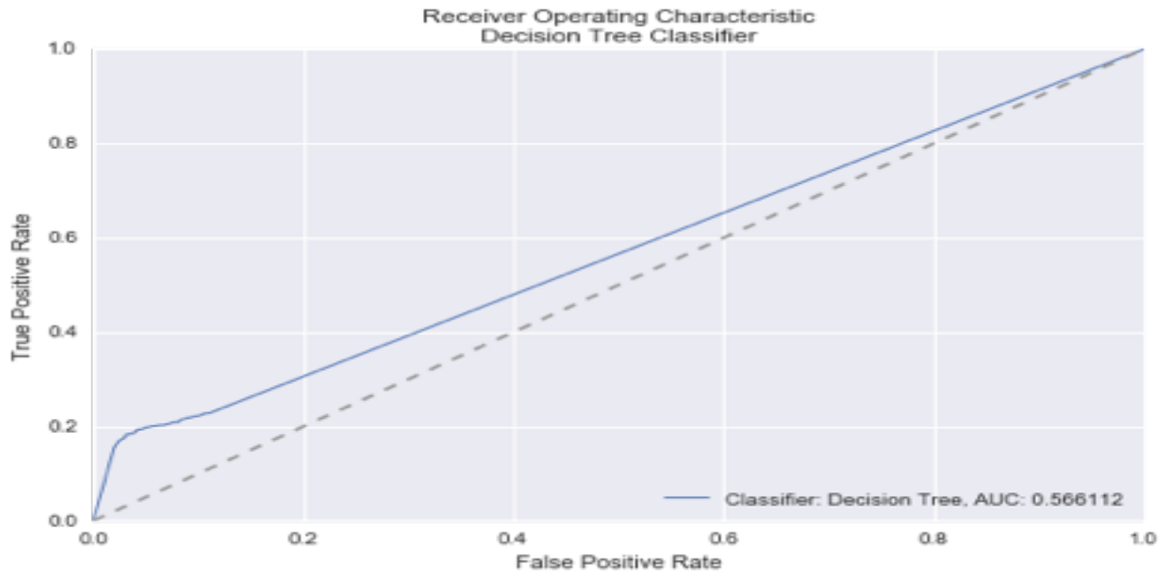
Weighted Validation

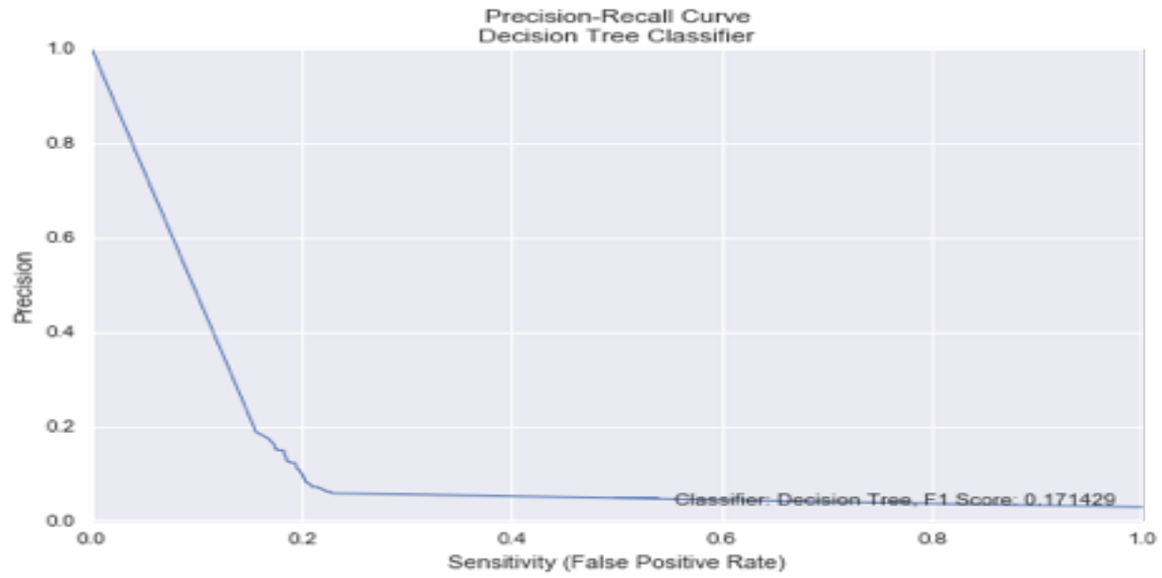
	precision	recall	f1-score	support
Fraud	0.64	0.16	0.25	32722.0
Not Fraud	0.51	0.91	0.65	31761.0
avg / total	0.58	0.53	0.45	64483.0

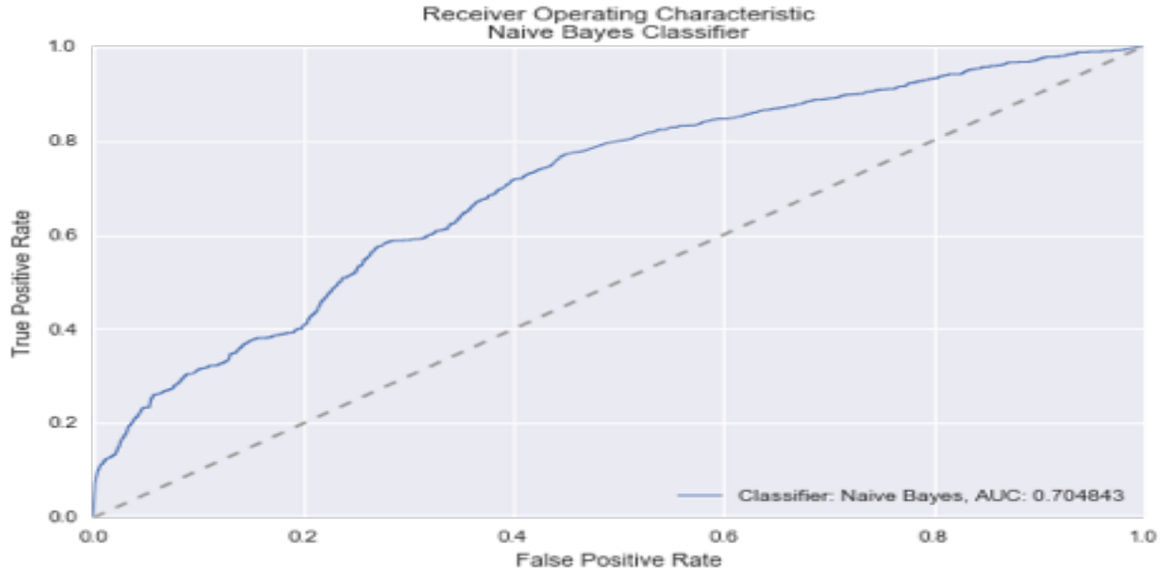
Real World Data

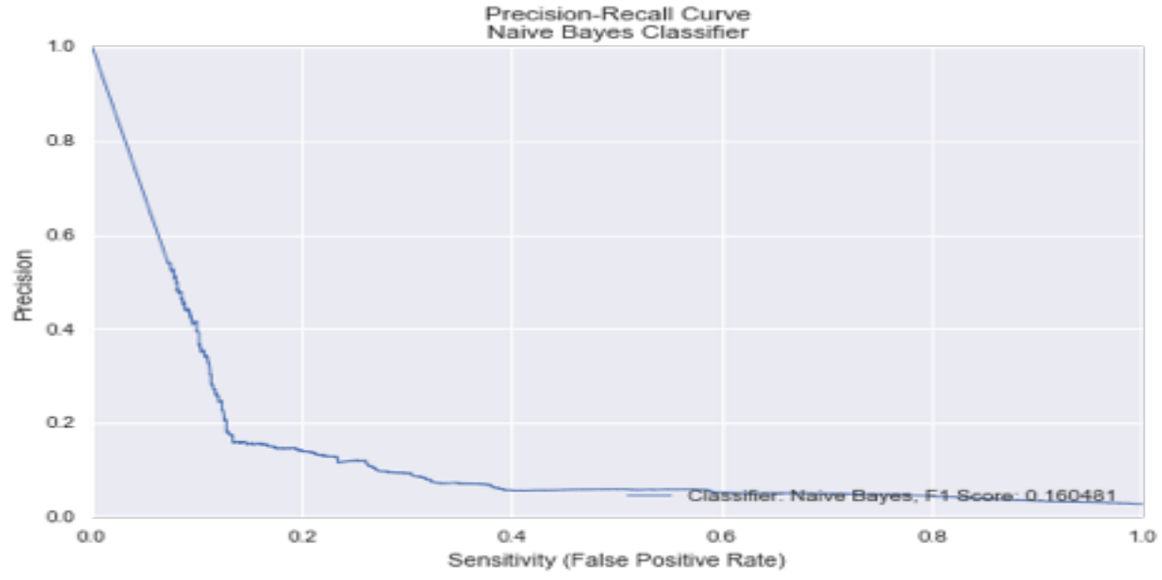
	precision	recall	f1-score	support
Fraud	0.07	0.27	0.10	961
Not Fraud	0.98	0.89	0.93	31761
avg / total	0.95	0.87	0.90	32722

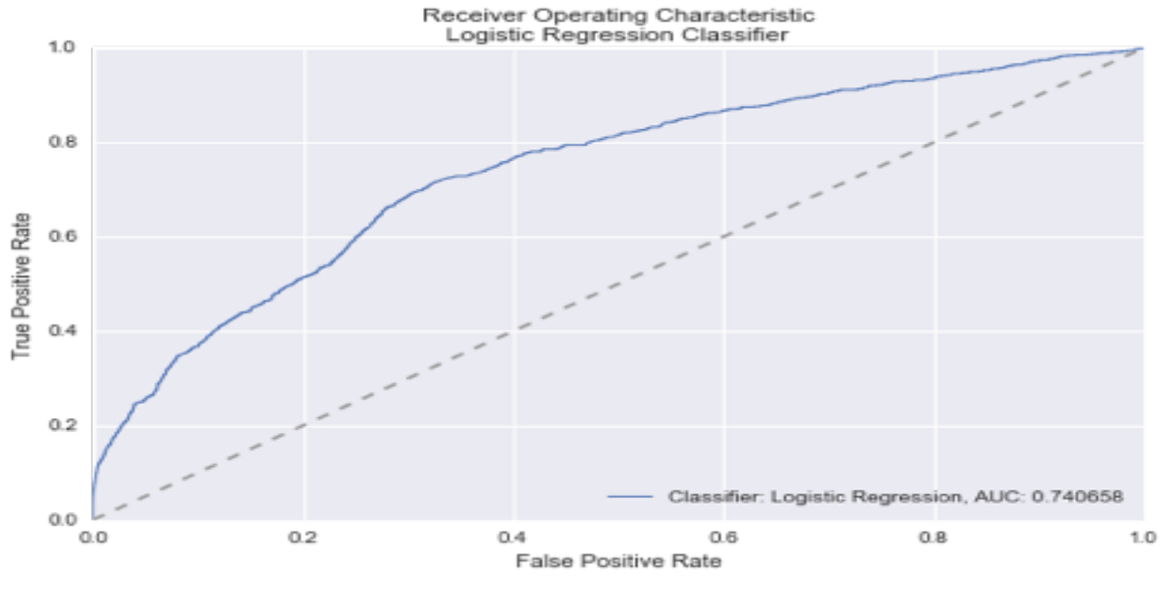
Generic Comparison of Classifiers on Imbalanced Data (DT = 0.5)

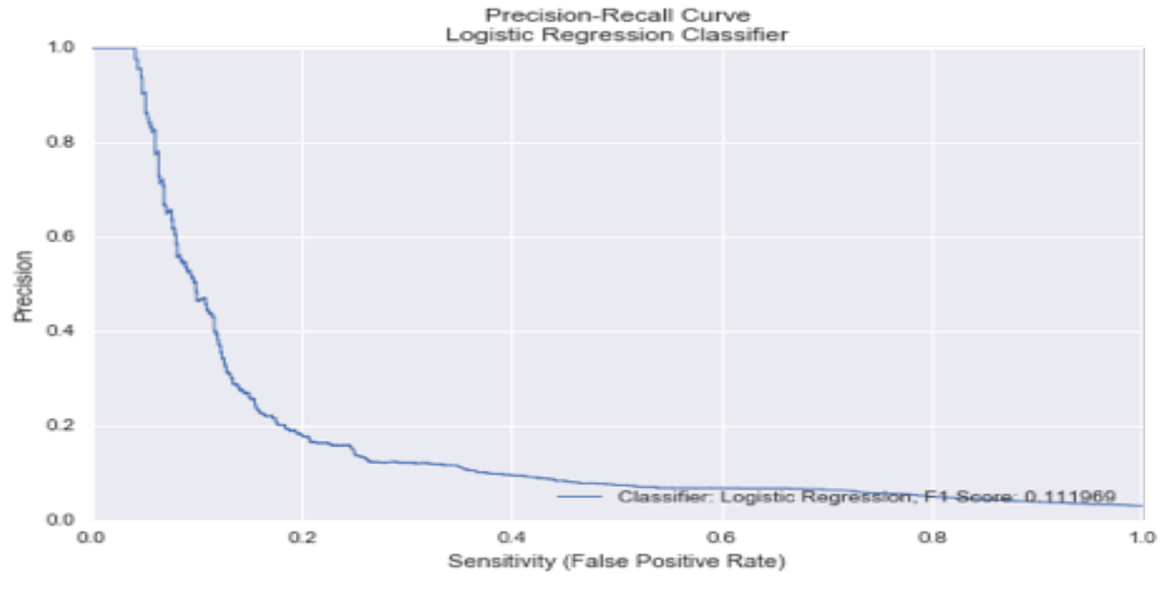








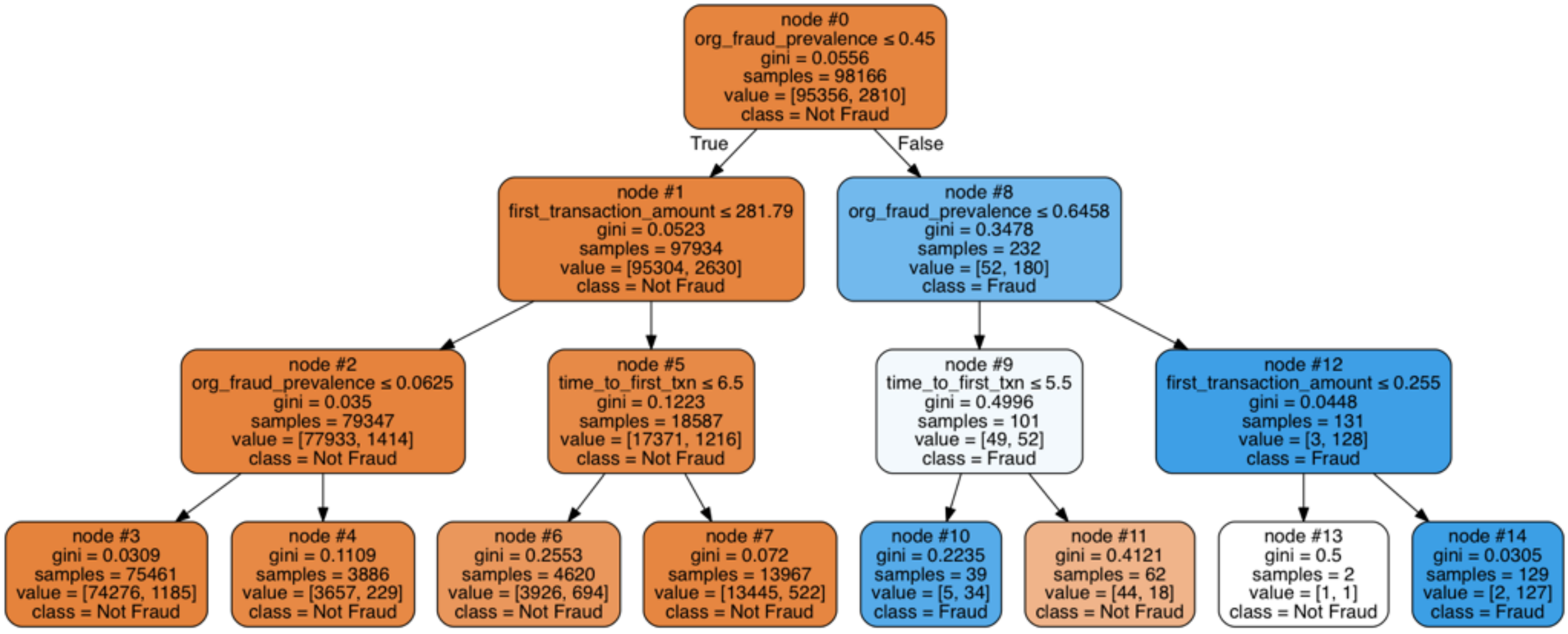




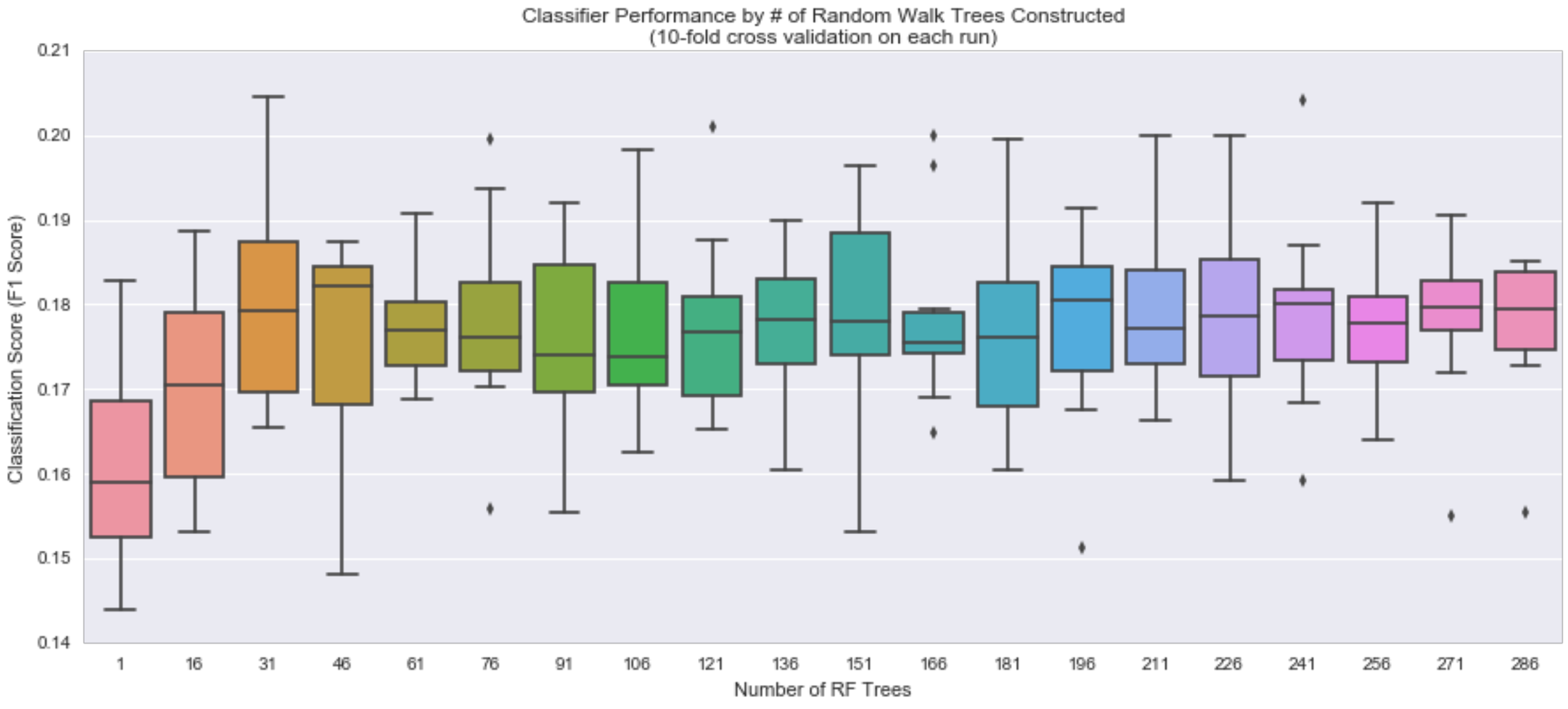
Next Steps

- Better Features
    - Location distances where possible (IP driven)
    - External sources of data: Organization BBB rating!
        - Incorporate *prior Fraud Prevalence* for certain $ amounts
- Experiment with other validation metrics
    - Cohen's Kappa Measure
    - Study explicit oversampling (SMOTE) some more
- Compare more classifiers, with intensive fine tuning of hyper parameters

Appendix A: Decision Tree, Depth 3 (SQL Query)



Appendix B: Tuning Optimal Number of Trees for Random Forest Using F1 Score



Location Homogeneity Coefficient