Fraud detection using Machine Learning

Aditya Oza - aditya19@stanford.edu

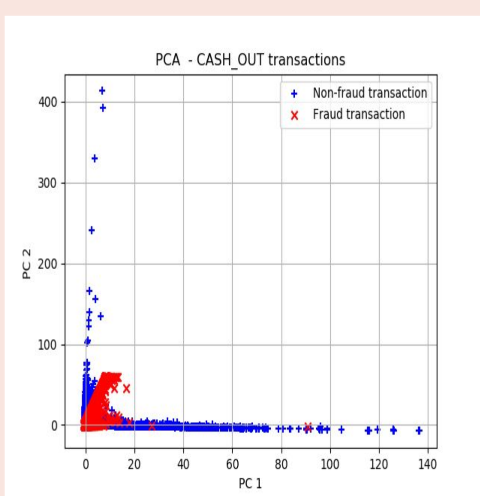
CS 229
Autumn 2018

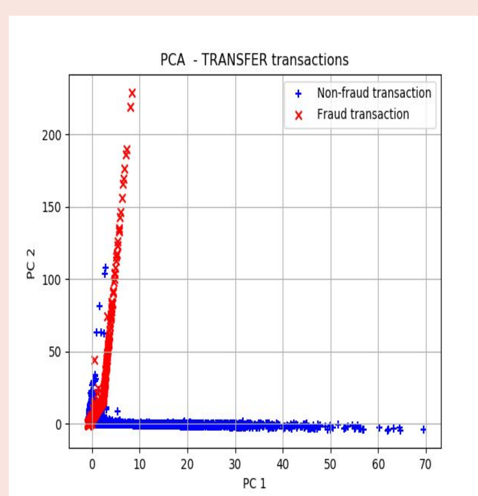
Introduction

- We build ML models to detect fraudulent activity in payment systems
- Used **PCA** for data visualization
- Build **binary classifiers** using Logistic Regression, Linear SVM, SVM with RBF kernel
- Developed approach to detect fraud with high accuracy and low number of false positives
- Achieved max recall - 99% on TRANSFER dataset

Dataset and Analysis

- **PaySim** - a Kaggle dataset for fraud detection
- **6 million +** mobile payment transactions
- 6 different categories of transactions
- **8312** fraudulent transactions
- Numerical and categorical features
- PCA on two categories - Transfer and Cash Out





Models

- **Logistic regression**

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} \varphi_{\text{logistic}}(y^{(i)}\theta^T x^{(i)}) = \frac{1}{m}\sum_{i=1}^{m} \log\left(1+\exp(-y^{(i)}\theta^T x^{(i)})\right)$$

- **Linear SVM**

$$min_{\gamma,w,b}\frac{1}{2}||w||^2 + C\sum_{i=1}^{m}\epsilon_i$$

$$s.t.\ y^{(i)}(w^T x^{(i)} + b) >= 1-\epsilon_i,\ i = 1,...,m$$

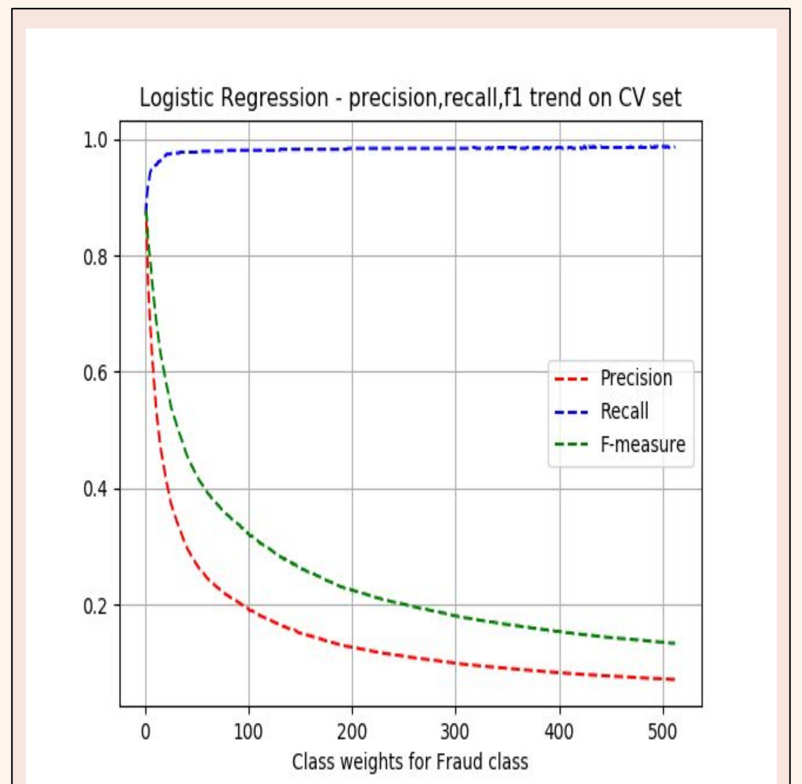
$$\epsilon_i >= 0,\ i = 1,...,m$$

- **SVM with RBF kernel**

$$K(x,z) = exp\left(-\frac{||x-z||^2}{2\sigma^2}\right)$$

Results

Weights tuning



- Precision, Recall trend for Cash Out for LR during training
- Tuned class weights by measuring recall, precision, f1-score on validation set

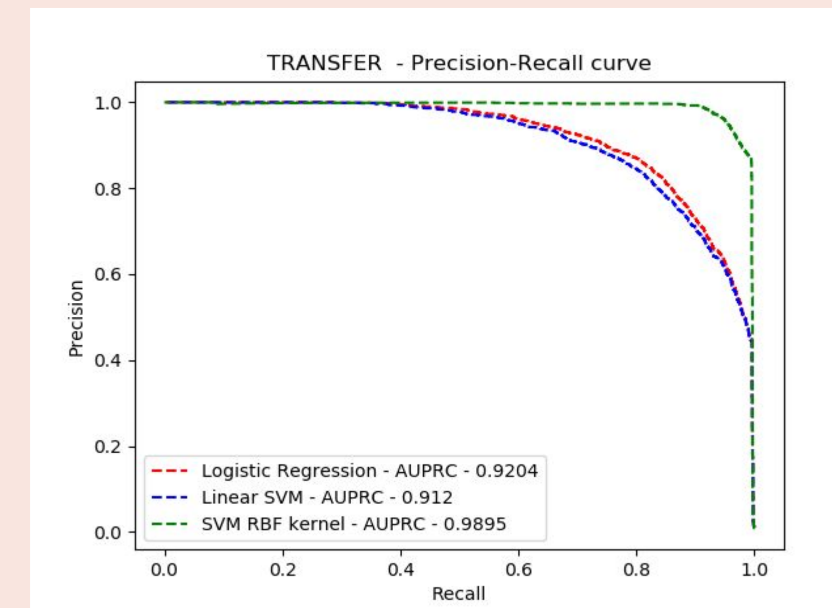
Precision Recall curve and statistics

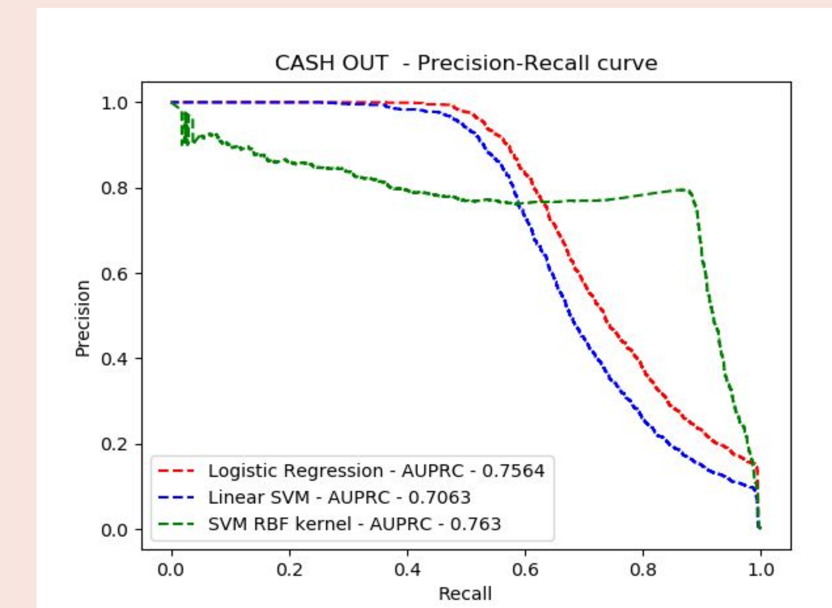
Train

TRANSFER transactions	Recall	Precision	f1 score	AUPRC
Logistic Regression	0.9958	0.4452	0.6153	0.9204
Linear SVM	0.9958	0.4431	0.6133	0.9121
SVM - RBF kernel	0.9958	0.6035	0.7515	0.9895

CASH OUT transactions	Recall	Precision	f1 score	AUPRC
Logistic Regression	0.9847	0.1541	0.2664	0.7564
Linear SVM	0.9361	0.1245	0.2199	0.7063
SVM - RBF kernel	0.9875	0.1355	0.2383	0.7631

Train



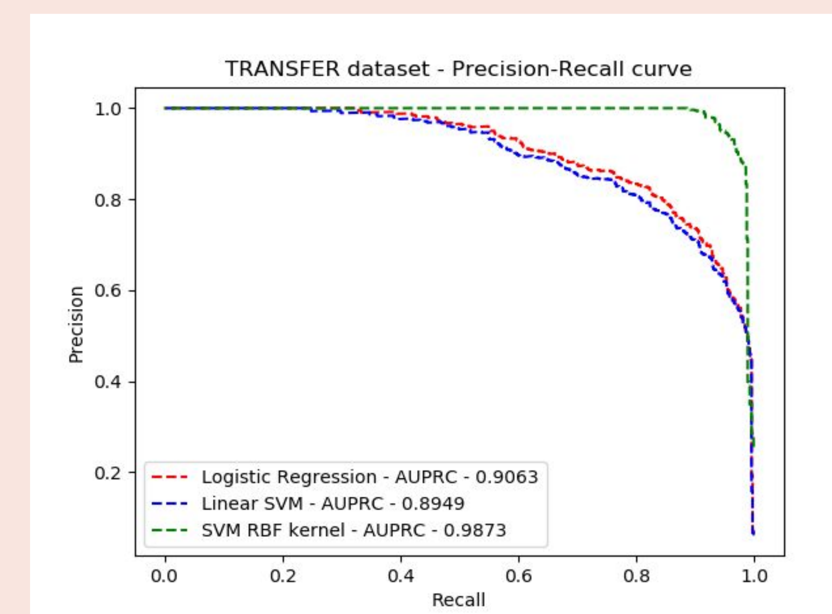


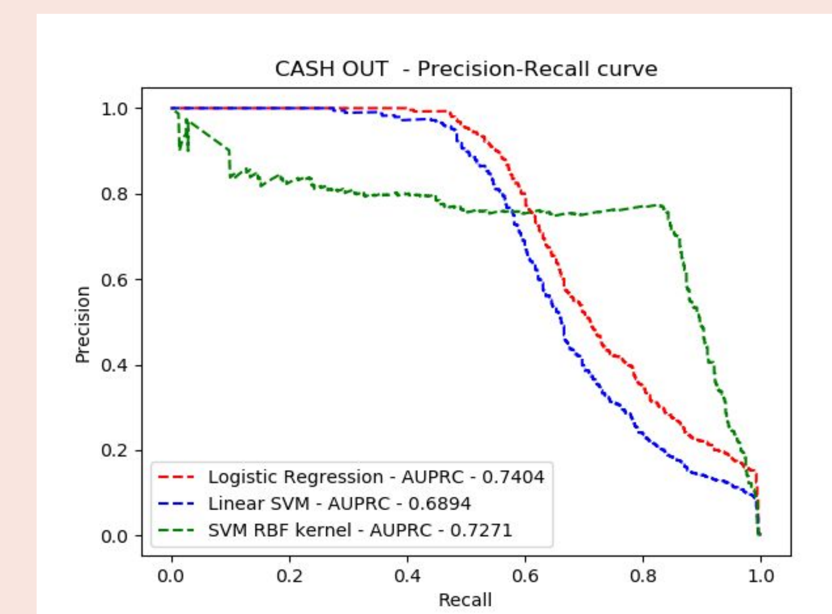
Test

TRANSFER transactions	Recall	Precision	f1 score	AUPRC
Logistic Regression	0.9951	0.4444	0.6144	0.9063
Linear SVM	0.9951	0.4516	0.6213	0.8949
SVM - RBF kernel	0.9886	0.5823	0.7329	0.9873

CASH OUT transactions	Recall	Precision	f1 score	AUPRC
Logistic Regression	0.9886	0.1521	0.2636	0.7403
Linear SVM	0.9411	0.1246	0.2201	0.6893
SVM - RBF kernel	0.9789	0.1321	0.2327	0.7271

Test





Class weight based approach

- In a fraud detection system, it's more critical to correctly detect fraud transactions and acceptable to misclassify certain number of non-fraud transactions.
- Penalize misclassification of fraud transactions more than non-fraud transactions
- Assign higher weights to fraud class to obtain high recall on that class and counter data imbalance.
- Ensure no more than ~1% false positives

Discussion

- We obtain very high AUPRC values for TRANSFER test set for all three methods - with ~0.98 highest value for SVM with RBF kernel
- Expected from PCA decomposition results as this category of transactions is linearly separable.
- Relatively lower recall, precision, AUPRC scores for CASH OUT test set.
- Further improvement on CASH OUT by setting higher threshold for false positives >> 1 %

Future work

- Decision Trees, Random Forests to leverage categorical features
- Time series based analysis for in context detection
- Customized user specific models based on user's past transactional activity.

References

[1] - A survey of credit card fraud detection - Sorournejad, Zojah, Atani et. al
[2] - Support Vector machines and malware detection - T.Singh, M.Stamp et. al
[3] - Paysim - A synthetic financial dataset for fraud detection - *https://www.kaggle.com/ntnu-testimon/paysim1*