Machine Learning Engineer Nanodegree

Capstone Proposal - Credit Card Fraud Detection

By -

Amber Ved

www.github.com/amberved

May 8th, 2017

Proposal

Domain Background

Big and high profile credit cards and data breaches have been dominating the headlines in the past couple of years across the world. These problem of Credit cards breaches in U.S. alone is responsible for 47 percent of the world's card fraud as of 2014. 15.4 million US consumers were affected by these kind of fraud in 2016, which is nearly 2 million more than in 2015. Dollar amount associated with such activities is 16 Billion in 2016 alone and significantly increasing at the rate of 61 Percent. Hence it is a big clear and present problem that needs Smarter soultions and machine learning can help.

According to data from the Federal Reserve, Credit card Fraud only impacts a fraction of all purchases made with Credit Cards but it represents one of the biggest concerns among consumers and also results into billions of dollars of losses to fininial companies be it bank, credit card companies, retailers & governments.

One can find lot of research and real world implementation done around this area like outlined at
https://www.research.ibm.com/foiling-financial-fraud.shtml
http://www.ulb.ac.be/di/map/adalpozz/pdf/Dalpozzolo2015PhD.pdf

Problem Statement

Credit card Fraud is a complex problem with large number of marid financial aspects. Consequently we need automatic systems able to support fraud detection and fightback. These systems are essential since it is not always possible or easy for a human analyst to detect fraudulent patterns in transaction datasets, often characterized by a large number of samples, many dimensions and online update.

The design of fraud detection machine learning algorithms is however particularly challenging due to the non-stationary distribution of the data, the highly unbalanced classes distributions and the availability of few transactions labeled by fraud investigators.

Listed below are few crucial issues any machine learning model will encounter and should attemp to address

i) Why and how undersampling is useful in the presence of class imbalance (i.e. frauds are a small percentage of the transactions)
ii) How to deal with unbalanced and evolving data streams (non-stationarity
due to fraud evolution and change of spending behavior)
iii) How to assess performances in a way which is relevant for detection and iv) How to use feedbacks provided by investigators on the fraud alerts generated.

Credit Card Fraud can in theory to detected based on various features like time of use, place of use, frequncy, unsual amount of spending, frequency of transection and many more such featues. But in general all this can be converted to mathematical values and machine learning models can be applied. In generic terms Credit Card Fraud identification can be treated as classicifcation problem.

Datasets and Inputs

I am planning to use Kaggel dataset on “Credit Card Fraud Detection” found at https://www.kaggle.com/dalpozz/creditcardfraud

This dataset presents Credit Card transactions, where it has 492 frauds out of 284,807 transactions. It contains around 30 features for each transection which are PCA transformation. All Features V1, V2, … V28 are the principal components obtained with PCA. Only 2 features which have not been transformed with PCA are ‘Time’ and ‘Amount’.

The intereseting aspect about using this data set is that ratio of frauds

vs total transactions is very low. Secountly, all the data is already convered into PCA so we can not judge the importance of any feature over other and have will force us to work with all of them equally and focus on the r2 or similar mathamatical relationships in place of human intution. Last point about this data set is that this is really good dataset as data preprocessing and cleansing is already done and with few changes can be fed into many supervised learning algorithms.

The major challenge with this data set is that it that fraud transactions like in real time is very small faction of over all transactions, hence we will have to find good strategy for data splitting in training/validation/testing subsets. In this case we will use *Resampling the dataset* methods
Essentially this is a method that will process the data to have an approximate 50-50 ratio. One way to achieve this is by OVER-sampling, which is adding copies of the under-represented class (better when you have little data). Another method could be UNDER-sampling, which deletes instances from the over-represented class (better when he have lot's of data). Hence we will try this schemes and measure performance by compare model with resampling and when not using it.

Solution Statement

This problem of identifying frudulant transection from valid credit card, can we taken as a classical ML Classfication problem. Simply put classification problems are tasked of identifying to which of a set of categories a new observation may belongs, on the basis of a training set of data containing observations whose category membership is

known.

This problem of identifying the fraulent transections can be broken into 3 steps.

1. Imbalanced in data - The ratio of valid vs fraud transection data avaliable to us.
2. Classifiction - I will use decisionTree, Randomforest, Naive Bayes classifier, Support vector machines & Quadratic classifiers algorithums.
3. Create a ensamble - The goal of ensemble will be to combine the predictions of several base models to built and improve generalizability / robustness over a single estimator. We will use both boosting ensemble methods to compare the with base estimators to reduce the bias of the combined estimator. The motivation is to combine several weak models to produce a powerful ensemble. I will test AdaBoost Classification, Gradient Tree Boosting Classification.

Benchmark Model

For Benchmark this problem, I will be using better results from either default out-of-box scikit implemenaton of Support Vector Machines (SVM) or Naive Bayes(NB) classifier algorithm.

Evaluation Metrics

Classifier performance depends greatly on the characteristics of the

data to be classified. There is no single classifier that works best on all given problems. The data set we are using for this problem is highly imbalanced and hence traditional popular measures like “precision and recall” and “receiver operating characteristic (ROC)” may not be very effective for this classification algorithms.

As a performance metric, I think the uncertainty coefficient will have a advantage over simple accuracy in that it is not affected by the relative sizes of the different classes. The uncertainty coefficient is useful for measuring the validity of a statistical classification algorithm and has the advantage over simpler accuracy measures in that it is not affected by the relative fractions of the different classes, i.e., P(x). It also has the unique property that it won’t penalize an algorithm for predicting the wrong classes, so long as it does so consistently (i.e., it simply rearranges the classes). This is useful in evaluating clustering algorithms since cluster labels typically have no particular ordering.

Suppose we have samples of two discrete random variables, X and Y. By constructing the joint distribution, PX,Y(x, y), from which we can calculate the conditional distributions, PX|Y(x|y) = PX,Y(x, y)/PY(y).

The uncertainty coefficient or proficiency is defined as:
U(X|Y)={{H(X)-H(X|Y)}{H(X)}}={{I(X;Y)}{H(X)}},

Where
The entropy of a single distribution is given as:
{H(X)=-\sum *{x}P*{X}(x)\log P_{X}(x),} H(X)=-\sum *{x}P*{X}(x)\log P_{X}(x),

while the conditional entropy is given as:
{H(X|Y)=-\sum *{x,~y}P*{X,Y}(x,~y)\log P_{X|Y}(x|y).} H(X|Y)=-\sum *{{x,~y}}P*{{X,Y}}(x,~y)\log P_{{X|Y}}(x|y).

Project Design

I intentd to follow the strategy to approach this problem:

A) Explore the dataset - Understand the data by finding basic relationships in data between various features and target variable.
B) Do necessary data preprocessing like feature reduction etc and
C) Use cross-validation to train several different ML algorithms. Get a baseline score and confusion matrix. We plan to use the following ML

1. K-means clustering algorithm,
2. Gaussian Mixture Model
3. Random random forest
4. Naive Bayes classifier
5. Logistic regression

D) Tune each model and validate noticable score improvements for each of them.

E) Train an ensemble using the stacking technique with previous models as base predictors. To get the final score. Model ensembling is a very powerful technique to increase accuracy on a variety of ML tasks as averaging multiple models often reduces the variance of single models.

As outlined confusion matrix could be a good way to evalute various models and algorithum effectiveness. Specially one can focus on having mild bias towards over reporting on -ve side (detecting transections as fraud in place of more relax where fraud transection are under reported). This could be better overall strategy as unreported fraud transection can result into financial loss. We dont have to different data sets for developing and benchmarking hence we will use random cross validation to compare various models. I will also like to benchmark our outcome against various other implementation avaliable on Kaggle.

Last but not the least, I will share the details of my model on Kaggel for other to review and share insight.

---

cited and references

https://www.javelinstrategy.com/coverage-area/2017-identity-fraud

https://www.businesswire.com/news/home/20170201005166/en/Identity-Fraud-Hits-Record-High-15.4-Million

https://www.kaggle.com/dalpozz/creditcardfraud

http://blog.kaggle.com/2016/07/21/approaching-almost-any-machine-learning-problem-abhishek-thakur/

https://en.wikipedia.org/wiki/K-means_clustering

https://en.wikipedia.org/wiki/Mixture_model

http://www.cs.cmu.edu/~guestrin/Class/10701-S07/Slides/clustering.pdf

http://www.ele.uri.edu/faculty/he/PDFfiles/ImbalancedLearning.pdf

http://www.ulb.ac.be/di/map/adalpozz/pdf/Dalpozzolo2015PhD.pdf