Credit Card Transaction Fraud Using Machine Learning Algorithms

Jiayi Huang

Shanghai Lixin University of Accounting and Finance, Shanghai, CN

angela@cas-harbour.org

Keywords: Credit card fraud, Z scale, Feature selection, Machine learning algorithms.

Abstract. Credit cards offered significant advantages over all forms of money: they're pocket size, easily portable, relatively secure and have no intrinsic value themselves. However, payment fraud is an ideal use case for machine learning algorithms and has a long track record of successful use. Machine learning has just been invented, or just been applied to payments fraud for the first time. This paper focuses on the main function of the feature selection in supervised model. The methods used to support the topic are neural net, boosted tree, random forest etc. and the material is credit card transaction data. The conclusion of the research is that banks should deny about 3% clients for balance the profits and loss of goods. A month was spent doing this research with the author's partners and professor for getting the results as accurate as possible.

1. Introduction

Fraud transaction fraud research has already been done by others, however, the algorithm was badly chosen and the results were inaccurate[5]. This paper uses the best fit algorithm which has been tested many times by the author to get the results. The whole process of building algorithms in supervised models is discussed, but the author puts much emphasis on feature selection[6]. Many algorithms may over-fit the original data or neglect the period of natural causes. In this paper, raw data contains many records, of which the important ones are the record number, card number, merchant number, merchant state, merchant zipcode, amount of transaction and fraud record. Besides, many variables in supervised models are built by these records. In supervised models, there are two methods used to solve the second step, one is "Autoencoder", which is a model trained to output the original vector input. After the model is trained, the difference between the original input vector and the model output vector is the fraud score for that record. The other is "Heuristic Function of the zscores". Finally, the author chooses the best algorithm to test the variables to get credit or fraud scores.

2. Methods and Analysis

2.1 Description of data and data cleaning

The data in this paper is from the bureau of the census, because the accuracy of the data could make the results objective and fair. The first step is using the algorithm to clean the data, including sorting and ranking the data, removing outlier and all but type P and filling missing values with the most common values. The raw data contains 96,753 records and 18 fields. 9 records are used as categorical variables. For card transaction data, only two missing values that need to be filled. Firstly, filling 'Merch state' by grouping by 'Merch zip', then fill 'Merchnum' by grouping by'Merch state' and 'Cardnum'.

Table 1 Raw data

	%populated	most frequent value	type
Recnum	100	NaN	int64
Cardnum	100	NaN	int64
Date	100	2010/9/7 00:00	datetime64[ns]

Merchnum	96.1	930090121224	object
Merch description	100	GSA-FSS-ADV	object
Transtype	100	P	object
Merch state	98.5	TN	object
Merch zip	94.4	38118.0	float64
Fraud	100	0	int64

The number of % populated shows available data and the type shows the type of data.

Table 2 Numeric variable

	mean	std	min	max	%populated	type
Amount	404.05	821.3	1.54	10774.26	100	float64

Table 2 use 1 field as numeric variable.

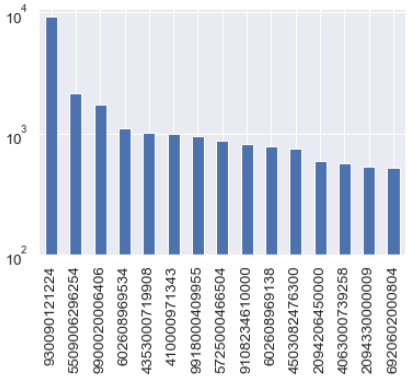

Fig. 1. Merchant number of transactions

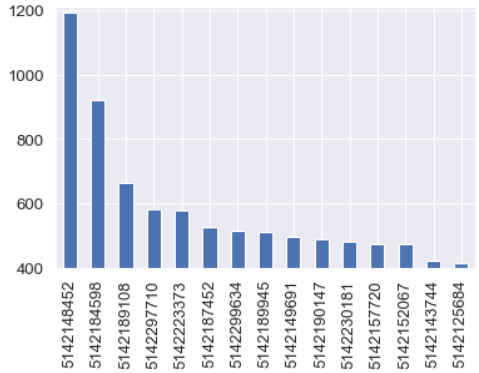

Fig. 2. Card number of transactions

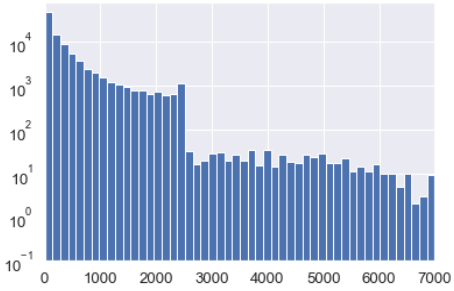

Fig. 3. Amount of money of transaction

Figure 3 shows that the amount of transaction has a gap around $2500 because in New York, if the amount of one transaction is over $2500, people have to ask for permission from the bank. Many people will not bother to make one transaction over $2500, so they may separate one transaction into several small transactions.

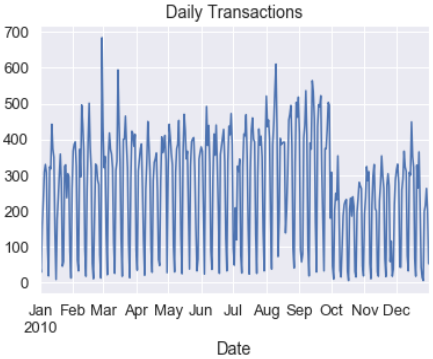

Fig. 4. Distribution of the number of transactions in each day

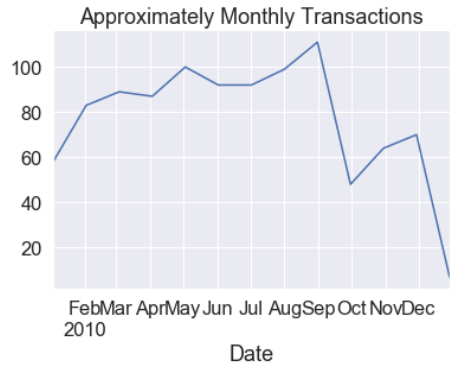

Fig. 5. Distribution of the number of transactions in each month

Figure 4 and 5 shows the number of transactions on each day and in each month. The author noticed the general upward trend through September, followed by a sharp drop in October. The monthly transactions are fewer in the last quarter of the year compared with other quarters. This is due to the government fiscal year which starts in October 1, and people tend to be more cautious with their money in the first three months of the new fiscal year.

2.2 Variable creation

a) Card Number Amount Expert Variables

Average Maximum Median Total Actual/average Actual/maximum Actual/median Actual/total	amount by/at this	card merchant card at this merchant card in this zip code card in this state	over the past	1 day 7 days 14 days 30 days

b) Merchant Amount Expert Variables

Meaning: finding the frequency of specific cards or the merchants' number in past days.

Number of transactions with this	card number merchant number card at this merchant card in this zip code card in this state	over the past	1 day 7 days 14 days 30 days

c) Day Since Expert Variables

Meaning: the time gap of the transaction of the same card number; the time gap of the transaction of the same merchant; the time gap of the same card and same merchant; the time gap of the same card in same zip code (region); the time gap of the same card in the same state.

Current date minus date of most recent transaction with same { card number, merchant, card at this merchant, card in this zip code, card in this state }

d) Velocity Expert Variables

Meaning: to point it out that the velocity of one card or one merchant in one day according to the past 7 days, 14 days, 30 days. The velocity could show the change of the transaction.

{ Number, Amount } of transactions with same { Card, merchant } over the past { 1 day }

Average daily { number, amount } of transactions with same { card, merchant } over the past { 7 days, 14 days, 30 days }

The general process is cleaning the data firstly and building variables, next step is using feature selection to build models and finally analyzing fraud savings.

2.3 Building supervised fraud model

2.3.1 Feature selection methods in supervised model

Feature Selection is one of the core concepts in machine learning which hugely impacts the performance of model. The data features that used to train machine learning models have a huge influence on the performance achieved[3].

a) Filter – independent of any modeling method

Pierson correlations, mutual info, univariate KS, univariate model performance measure(e.g. FDR, lift). For each variable make separate distributions for the two populations (good, bad). The amount of separation between the distributions is the importance of the variable[1].

A Pearson correlation is a number between -1 and 1 that indicates the extent to which two variables are linearly related. Pearson correlations are suitable only for metric variables (which include dichotomous variables).

Pearson Correlation: $c_{ij} = \frac{\sigma 2_{ij}}{\sigma_{ii}\sigma_{jj}}$, where i is x_i and j is y_j measures linear correlation between a feature and the label. We get the fisher score which could simply measure how the two means are separated.

Fisher Score: $F = \frac{n_0(\mu_0-\mu)^2+n_1(\mu_1-\mu)^2}{n_0\sigma_0^2+n_1\sigma_1^2}$

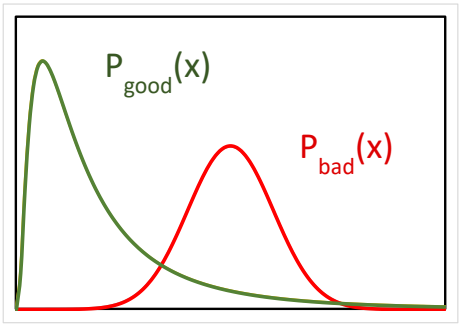

Fig. 6. Continuous variable

Fig. 7. Discrete variable

Kolmogorov-Smirnov, which is a nice, simple distribution/scale independent measure of distance between two distributions:

$$KS = max_x \int_{x_{min}}^{x} [P_{good} - P_{bad}]dx \quad (1)$$

$$KS = max_x \sum_{x_{min}}^{x} [P_{good} - P_{bad}] \quad (2)$$

Kulback-Leibler, which is an information theory-based measure. Be careful when the distributions go to zero:

$$KL = \int P_{good}(x) log \frac{P_{good}(x)}{P_{bad}(x)} dx \quad (3)$$

$$KL = \sum_i P_{good}(x_i) log \frac{P_{good}(x_i)}{P_{bad}(x_i)} \quad (4)$$

Information Value, which is a Symmetric version of Kulback-Leibler distance:

$$IV = \int [P_{good}(x) - P_{bad}(x)] log \frac{P_{good}(x)}{P_{bad}(x)} dx \quad (5)$$

$$IV = \sum_i [P_{good}(x_i) - P_{bad}(x_i)] log \frac{P_{good}(x_i)}{P_{bad}(x_i)} \quad (6)$$

Mutual Information, which is another information theory-based measure, requires joint distribution:

$$MI = \int_x \int_y P(x,y) log \frac{P(x,y)}{P(x)P(y)} dxdy \quad (7)$$

$$MI = \sum_x \sum_y P(X = x, Y = y) log \frac{P(X=x,Y=y)}{P(X=x)P(Y=y)} \quad (8)$$

b) Wrapper - Stepwise Selection Methods

A wrapper method has a model "wrapped" around the process. The model can be anything. So we could say there are n candidate features/inputs/variables.

Wrapper methods generally result in better performance than filter methods because the feature selection process is optimized for the classification algorithm to be used. However, wrapper methods are too expensive for large dimensional database in terms of computational complexity and time since each feature set considered must be evaluated with the classifier algorithm used[4].

c) Regularization

Some forms of stepwise logistic regression is suggested. Next step is to explore a handful of nonlinear models on the final data set, select best models and finalize. It's also good to use regularization in this step.

Regularization for Feature Selection:

Ridge regression:

Minimize: where $(y - \sum a_i x_i)^2 + \lambda \|a\|_2^2$ is the L_2 norm $\|a\|_2^2 = \sum a_i^2$

Lasso regression:

Minimize: where $(y - \sum a_i x_i)^2 + \lambda \|a\|_1$ is the L_1 norm $\|a\|_1 = \sum |a_i|$

Elastic Net regression:

Minimize: $(y - \sum a_i x_i)^2 + \lambda_1 \|a\|_1 + \lambda_1 \|a\|_2^2$

2.3.2 Model algorithms

a) Logistic Regression

A linear regression is appropriate when the output y has a continuous range. However, for classification problems where the output y takes one of two values (e.g., good/bad), the output of the model y is the probability that it takes a particular value. Since the output y only ranges from zero to one, a logistic regression is more appropriate[2].

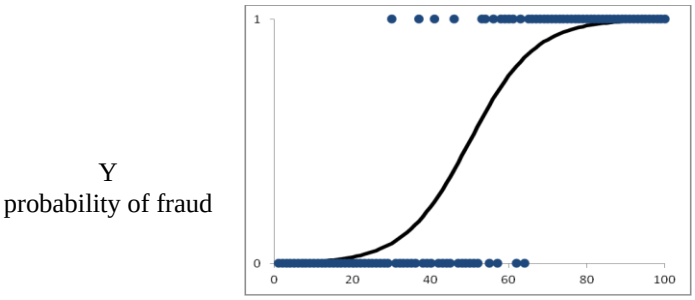

Fig. 8. Logistic regression

The logit function “squashes” the linear regression function at high and low values, restricting it to between 0 and 1.

b) Neural Net

A typical neural net consists of an input layer, some number of hidden layers and an output layer. All the independent variables (x’s) form the input layer. The dependent variable y is the output layer. The hidden layer is a set of nodes or “neurons”. Each node in the hidden layer receives weighted signals from all the nodes in the previous layer and does a transform on this linear combination of signals. The transform/activation function can be a logistic function (sigmoid), or something else.

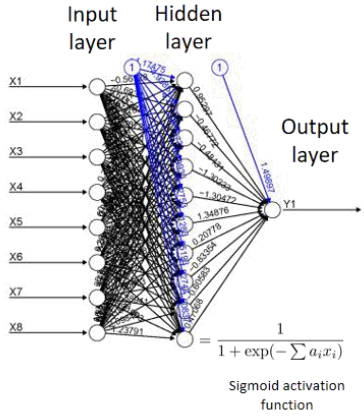

Fig. 9. Neural net

A neural net is a mathematical function that maps inputs to an output with a bunch of adjustable parameters. Typical error/loss function is the square of the errors: $E(y, \hat{y}) = |y - \hat{y}|^2$

In this work, it is a neural network with only one layer and two nodes. The activation function is logistic and the parameter before the regularization is 0.00001. The learning rate is 0.4 initially and it becomes smaller and smaller gradually.

c) Random forest (depends on decision tree model)

A decision tree model carves up space into boxes and then assigns a score for each box. A typical decision tree model will be in several or many dimensions with 10s or hundreds of leaf nodes. For lowing impurities on both sides, using variance for continuous outputs.

The total impurity of the resulting cut is: $I = I_1 n_1 + I_2 n_2$; I_i (i = 1, 2) means impurity of box i; n_i (i = 1, 2) means number records in box i.

Common measures of impurity:

- Variance: $I = \sum (y_i - < y_i >)^2$
- Gini index: $I = 1 - \sum p_i^2 = 1 - \left(\frac{n_g}{n}\right)^2 \left(\frac{n_b}{n}\right)^2$
- Entropy or Information Gain: $I = -\sum p_i ln(p_i) = -(p_g ln(p_g) + p_b ln(p_b))$

Then the author adds the impurities of both resulting boxes weighted by the number in each box and chooses the best cut point that results in the lowest impurity.

In this work, only one decision tree is not robust enough, so the author uses random forest, which expands from decision tree. Building many trees, each tree uses only a randomly-chosen subset of variables and combine all the results by averaging or voting.

d) Boosted tree

Boosting is a way of training a series of weak learners to result in a strong learner, and each weak learner is trained to predict the residual error of the current sum like Taylor Series. Adaptive boosting (AdaBoost) is assigning a new weight on each data record for training the next weak learner in the series (uses all records but with weights). AdaBoost increases the weights on misclassified records, so the next iteration can pay more attention to them. Popular AdaBoost algorithms: GentleBoost, LogitBoost, BrownBoost, DiscreteAB, KLBoost, RealAB.

2.3.3 Results analysis

After training these four algorithms and got the final results, the author finds out which algorithm is the best.

Table 3 FDR @ 3%

	FDR @ 3%		
	Training	Testing	Out of time
Logistic Regression	80.6	76	46
Neural Net	64.9	62	50.2
Random Forest	87.4	80	53.6
Boosted Trees	81.3	77.9	46.3

Table 3 shows that the random forest has the best results and does not overfit much, so we choose random forest as our best model to do the predicting. (Table 4, 5 and 6 show the results of the random forest in detail)

Table 4 Training results of the random forest

Training	# Records	# Goods	# Bads	Fraud Rate
	58,779	58,180	599	0.0102

	Bin Statistics					Cumulative Statistics						
Population Bin %	# Records	# Goods	# Bads	% Goods	% Bads	Total # Records	Cumulative Good	Cumulative Bad	% Good	% Bad (FDR)	KS	FPR
1	588	210	378	35.7	64.3	588	210	378	0.4	63.1	62.7	0.6
2	588	500	88	85.0	15.0	1,176	710	466	1.2	77.8	76.6	1.5
3	587	526	61	89.6	10.4	1,763	1,236	527	2.1	88.0	85.9	2.3
4	588	553	35	94.0	6.0	2,351	1,789	562	3.1	93.8	90.7	3.2
5	588	559	29	95.1	4.9	2,939	2,348	591	4.0	98.7	94.6	4.0
6	588	585	3	99.5	0.5	3,527	2,933	594	5.0	99.2	94.1	4.9
7	588	585	3	99.5	0.5	4,115	3,518	597	6.0	99.7	93.6	5.9
8	587	586	1	99.8	0.2	4,702	4,104	598	7.1	99.8	92.8	6.9
9	588	588	0	100.0	0.0	5,290	4,692	598	8.1	99.8	91.8	7.8
10	588	588	0	100.0	0.0	5,878	5,280	598	9.1	99.8	90.8	8.8
11	588	588	0	100.0	0.0	6,466	5,868	598	10.1	99.8	89.7	9.8
12	587	586	1	99.8	0.2	7,053	6,454	599	11.1	100.0	88.9	10.8
13	588	588	0	100.0	0.0	7,641	7,042	599	12.1	100.0	87.9	11.8
14	588	588	0	100.0	0.0	8,229	7,630	599	13.1	100.0	86.9	12.7
15	588	588	0	100.0	0.0	8,817	8,218	599	14.1	100.0	85.9	13.7
16	588	588	0	100.0	0.0	9,405	8,806	599	15.1	100.0	84.9	14.7
17	587	587	0	100.0	0.0	9,992	9,393	599	16.1	100.0	83.9	15.7
18	588	588	0	100.0	0.0	10,580	9,981	599	17.2	100.0	82.8	16.7
19	588	588	0	100.0	0.0	11,168	10,569	599	18.2	100.0	81.8	17.6
20	588	588	0	100.0	0.0	11,756	11,157	599	19.2	100.0	80.8	18.6

Table 5 Testing results of the random forest

Testing	# Records	# Goods	# Bads	Fraud Rate
	25,191	24,910	281	0.0112

	Bin Statistics					Cumulative Statistics						
Population Bin %	# Records	# Goods	# Bads	% Goods	% Bads	Total # Records	Cumulative Good	Cumulative Bad	% Good	% Bad (FDR)	KS	FPR
1	252	82	170	32.5	67.5	252	82	170	0.3	60.5	60.2	0.5
2	252	217	35	86.1	13.9	504	299	205	1.2	73.0	71.8	1.5
3	252	235	17	93.3	6.7	756	534	222	2.1	79.0	76.9	2.4
4	252	244	8	96.8	3.2	1,008	778	230	3.1	81.9	78.7	3.4
5	252	244	8	96.8	3.2	1,260	1,022	238	4.1	84.7	80.6	4.3
6	251	248	3	98.8	1.2	1,511	1,270	241	5.1	85.8	80.7	5.3
7	252	248	4	98.4	1.6	1,763	1,518	245	6.1	87.2	81.1	6.2
8	252	252	0	100.0	0.0	2,015	1,770	245	7.1	87.2	80.1	7.2
9	252	248	4	98.4	1.6	2,267	2,018	249	8.1	88.6	80.5	8.1
10	252	247	5	98.0	2.0	2,519	2,265	254	9.1	90.4	81.3	8.9
11	252	249	3	98.8	1.2	2,771	2,514	257	10.1	91.5	81.4	9.8
12	252	248	4	98.4	1.6	3,023	2,762	261	11.1	92.9	81.8	10.6
13	252	252	0	100.0	0.0	3,275	3,014	261	12.1	92.9	80.8	11.5
14	252	250	2	99.2	0.8	3,527	3,264	263	13.1	93.6	80.5	12.4
15	252	251	1	99.6	0.4	3,779	3,515	264	14.1	94.0	79.8	13.3
16	252	250	2	99.2	0.8	4,031	3,765	266	15.1	94.7	79.5	14.2
17	251	250	1	99.6	0.4	4,282	4,015	267	16.1	95.0	78.9	15.0
18	252	252	0	100.0	0.0	4,534	4,267	267	17.1	95.0	77.9	16.0
19	252	251	1	99.6	0.4	4,786	4,518	268	18.1	95.4	77.2	16.9
20	252	250	2	99.2	0.8	5,038	4,768	270	19.1	96.1	76.9	17.7

Table 6 Out of time results of the random forest

Out Of Time	# Records	# Goods	# Bads	Fraud Rate
	12,427	12,248	179	0.0144

	Bin Statistics					Cumulative Statistics						
Population Bin %	# Records	# Goods	# Bads	% Goods	% Bads	Total # Records	Cumulative Good	Cumulative Bad	% Good	% Bad (FDR)	KS	FPR
1	124	58	66	46.8	53.2	124	58	66	0.5	36.9	36.4	0.9
2	125	113	12	90.4	9.6	249	171	78	1.4	43.6	42.2	2.2
3	124	105	19	84.7	15.3	373	276	97	2.3	54.2	51.9	2.8
4	124	114	10	91.9	8.1	497	390	107	3.2	59.8	56.6	3.6
5	124	120	4	96.8	3.2	621	510	111	4.2	62.0	57.8	4.6
6	125	117	8	93.6	6.4	746	627	119	5.1	66.5	61.4	5.3
7	124	121	3	97.6	2.4	870	748	122	6.1	68.2	62.0	6.1
8	124	118	6	95.2	4.8	994	866	128	7.1	71.5	64.4	6.8
9	124	122	2	98.4	1.6	1,118	988	130	8.1	72.6	64.6	7.6
10	125	124	1	99.2	0.8	1,243	1,112	131	9.1	73.2	64.1	8.5
11	124	122	2	98.4	1.6	1,367	1,234	133	10.1	74.3	64.2	9.3
12	124	123	1	99.2	0.8	1,491	1,357	134	11.1	74.9	63.8	10.1
13	125	124	1	99.2	0.8	1,616	1,481	135	12.1	75.4	63.3	11.0
14	124	124	0	100.0	0.0	1,740	1,605	135	13.1	75.4	62.3	11.9
15	124	123	1	99.2	0.8	1,864	1,728	136	14.1	76.0	61.9	12.7
16	124	124	0	100.0	0.0	1,988	1,852	136	15.1	76.0	60.9	13.6
17	125	125	0	100.0	0.0	2,113	1,977	136	16.1	76.0	59.8	14.5
18	124	123	1	99.2	0.8	2,237	2,100	137	17.1	76.5	59.4	15.3
19	124	123	1	99.2	0.8	2,361	2,223	138	18.1	77.1	58.9	16.1
20	124	122	2	98.4	1.6	2,485	2,345	140	19.1	78.2	59.1	16.8

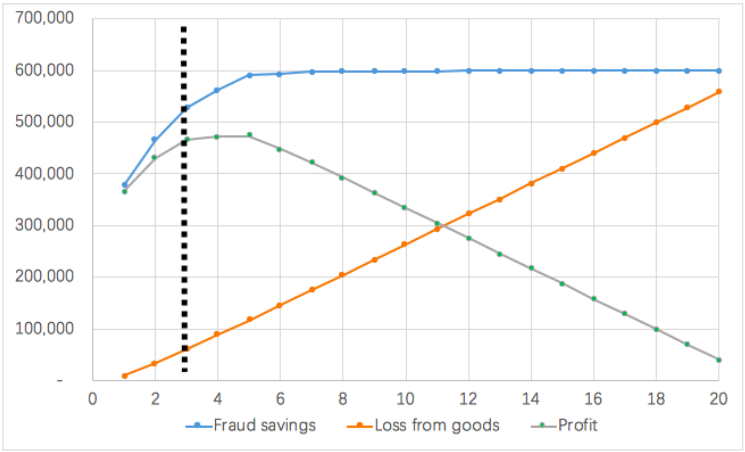

Fig. 10. Relationship between clients, money and profit

In figure 10, the x axis is the percentage of the clients denied and the units of y axis is dollar. The highest point of the profit and the black imaginary line is the number of how many frauds should be caught. If deny too many clients, the company will lose the good ones' credit, but if approve too many clients, the bank will lose money and the number of frauds will increase, which is not wise.

3. Discussion

3.1 Future expectation

The resulting model can be utilized in a credit card fraud detection system as well as in a New York property data. Similar model development process can be performed in related business domains such as insurance and telecommunications, to avoid or detect fraudulent activity.

It could also be found that using the scores of statistical significance in the logistic regression models as the criteria for selecting variables that high statistical significance score or the low p_value means strong correlation between fraud and related variable. The most important step is the construction of good expert variables that encode the signals of the problem dynamics as much as possible into clever variables. For this reason, linear and logistic regression models are so prevalent in the business field.

More data with more fields (for example, adding point of sale information, time of day, or other cardholder or merchant information) would certainly allow model performance to make improvements. Further model parameter tuning would also provide improvements to any of the models.

3.2 Suggestions

The fraud of transaction card number cannot depend on the data completely, the result could be less or more flexible, the rejection of the transaction should be controlled within a certain number so that could meet the companies' needs. The resources of transaction card frauds come from the loophole of system so if the government could adjust the system, the fraud number could be controlled dramatically.

4. Conclusion

This paper is written to explore the applications of linear and nonlinear statistical and machine learning models based on New York Property data and Credit Card Transaction data. The models are supervised fraud models that attempt to identify which transactions are most likely fraudulent. As expected, the nonlinear models slightly outperform the linear model, except for the artificial neural network. It is believed that this underperformance is due to two reasons. First, we have not sufficiently tuned the neural net model and improvement can be found with a different set of model parameters. Second, banks noting that the data set is substantially limited, and only has 179 that are labeled fraud events in this Out of Time data set. The results of any model, particularly a nonlinear one, can be volatile and sensitive to the statistical aberrations and the variation of model parameters. The random forest model performs the best and can detect about half of the fraud attempts within only top 3% data sorted as suspicious by the fraud algorithm score. However, the out-of-time results of random forest are very close to neural net because of small data set.

References

[1] Credit card fraud detection based on transaction behavior: Kho J R D , Vea L A . [IEEE TENCON 2017-2017 IEEE Region 10 Conference - Penang (2017.11.5-2017.11.8)] TENCON 2017-2017 IEEE Region 10 Conference-Credit card fraud detection based on transaction behavior[J]. pp.1880-884.

[2] X. T. Niu, L. Wang and X. L. Yang, A Comparison Study of Credit Card Fraud Detection: Supervised versus Unsupervised. Vol.1,Wed, 24 Apr 2019.

[3] L. Yu and H. Liu, Feature Selection for High-Dimensional Data: A Fast Correlation-Based Filter Solution, Department of Computer Science & Engineering, Arizona State University, Tempe, AZ 85287-5406, USA, Proceedings of the Twentieth International Conference on Machine Learning (ICML-2003), Washington DC, 2003.

[4] L. Weston, “Equifax Just Changed the Rest of Your Life,” March, 2018.[Online]. Avaliable: https://www.nerdwallet.com/blog/finance/equifax-changed-your-life/ [Accessed on Nov. 14, 2019]
[5] M.A.Jayaram and Asha Gowda Karegowda, A.S. Manjunath: Feauture selection Problem using Wrapper Approach in Supervised Learning, 2010 International Journal of Computer Applications (0975 - 8887) Vol. 1, No. 7.
[6] Parcollet, T., Morchid, M. & Linarès, G. Artif Intell Rev, A survey of quaternion neural networks, (2019). [Online]. Avaliable:https://doi.org/10.1007/s10462-019-09752-1[Accessed on Oct.15, 2019].