Financial Fraud Detection Using Machine Learning Techniques

Yordan Ivanov

1.Introduction

What we provide here is an extensive study of machine learning methods on different both real-world and simulated datasets in an attempt to provide better guidelines for fraud detection.

- specify the methods that are going to be used in the study

2. Literature Review

3. Methodology

Classification is one of the most widely used model framework used for the application of machine learning techniques in terms of FFD (Ngai et al. 2011). Some of the most common classification techniques include logistic regression, neural networks, support vector machine and decision trees.

3.1. Preliminaries

In the current section, we will give a description of the machine learning techniques that we apply to predict fraudulent transactions.

Let us define our binary classification dataset as $\{(x_1, y_1), (x_2, y_2), ..., (x_p, y_p)\}$, where $x_i \in \mathbb{R}^n$ represents an n-dimensional data point and $y_i \in \{-1, 1\}$ represents the label of the class of that data point, $i = 1, ..., p$.

3.2. Machine Learning Algorithms

3.2.1. Logistic Regression

The logistic regression framework falls under the category of generalized linear models and allows the prediction of discrete outcomes. Then, by defining the probability of a transaction being fraudulent by $p(X) = Pr(Y = 1|X)$, we can portray the relationship between the dependent and independent variables as follows:

$$p(X) = \frac{e^{\beta_0+\beta_1 X_1+...+\beta_p X_p}}{1 + e^{\beta_0+\beta_1 X_1+...+\beta_p X_p}}$$

The number of independent variables is indexed by p. After manipulating (), we can also see that

$$log(\frac{p(X)}{1-p(X)}) = \beta_0 + \beta_1 X_1 + ... + \beta_p X_p$$

with the LHS being called the logit. Using equation (), we will predict the probabilities of a transaction being fraudulent i.e. $p(Y = 1)$. The fitting of a logistic regression is done by the method of maximum likelihood (see Appendix). The logistic regression has been among the most widely used framework in fraud detection

(Ngai et al. 2011) due to simplicity of ease of implementation, but it does have its shortcomings - it tends to underperform when there are multiple or non-linear decision boundaries (*SEARCH FOR SOME PAPER OR BOOK ON IT?*)

3.2.2. Neural Networks

3.2.3. Support Vector Machines

Support Vector Machines, developed by Vapnik et. al. (Cortes and Vapnik 1995), have become a popular machine learning method that has seen its implementation rise in various domains that require the use of classification models (Batuwita and Palade 2013). Among the factors for its success is the fact that the SVMs are linear classifiers, which work in a high-dimensional feature that represents a non-linear mapping of the input space of the problem being dealt with (Bhattacharyya et al. 2011). Working in a high-dimensional feature space has its benefits - often, the problem of non-linear classification in the original input space is transformed to a linear classification task in the high-dimensional feature space.

The goal of the SVM classifier consists of finding the optimal separating hyperplane, which manages to effectively separate the observations from the data into two classes. As mentioned above, the observations are initially transformed by a nonlinear mapping function Φ. Thus, we can write a possible separating hyperplane that resides in the transformed higher dimensional feature space by:

$$w \cdot \Phi(x) + b = 0$$

with w the weight vector normal to the hyperplane.

We will further use two variations of the SVM soft margin optimization problem - one that assigns the same cost for missclassification of the different classes and one that penalizes more the missclassification of the minority class.

Non-cost sensitive learning

For the same missclassification cost case, we can write the soft optimization problem as follow:

$$\min(\frac{1}{2} w \cdot w + C \sum_{i=1}^{p} \xi_i)$$

$$s.t. \ \ y_i(w \cdot \Phi()x_i + b) \geq 1 - \xi_i$$

$$\xi_i \geq 0, i = 1, ..., p$$

The slack variables $\xi_i > 0$ hold for missclassified examples. Thus, the penalty term $\sum_{i=1}^{p}$ can be perceived as the total number of missclassified observations of the model. Thusm from (), we can see that there are two goals - maximizing the margin the minimizing the number of missclassifications. The cost parameter C controls the trade-off between them. The quadratic optimization problem in () can be represented by a dual Lagrange problem and then solved:

$$\max_{\alpha_i} \{ \sum_{i=1}^{p} \alpha_i - \frac{1}{2} \sum_{i=1}^{p} \sum_{j=1}^{p} \alpha_i \alpha_j \Phi(x_i) \cdot \Phi(x_j) \} \ \ s.t. \ \ \sum_{i=1}^{p} y_i \alpha_i = 0, \ \ 0 \leq \alpha_i \leq C, \ \ i = 1, ..., p$$

Thanks to another one of the strenghts of SVM - kernel representation - we don't need to explicitly know the mapping function $\Phi(x)$, but by applying a kernel function (i.e. $K(x_i, x_j) = \Phi(x_i) \cdot \Phi(x_j)$), we can rewrite () as:

$$\max_{\alpha_i} \{ \sum_{i=1}^{p} \alpha_i - \frac{1}{2} \sum_{i=1}^{p} \sum_{j=1}^{p} \alpha_i \alpha_j K(x_i, x_j) \} \ \ s.t. \ \ \sum_{i=1}^{p} y_i \alpha_i = 0, \ \ 0 \leq \alpha_i \leq C, \ \ i = 1, ..., p$$

Cost Sensitive learning

The regular SVM model has been effectively implemented when the dataset used has balanced classes, however it fails to produce good results when applied on imbalanced data (Batuwita and Palade 2013). When trained on a dataset with imbalanced classes, the SVM classifier tends to produce results that are biased towards the majority class and fail to efficiently predict the minority class.

Kernels

3.2.4. Classification and regression trees (CARTs)

Random Forests

Gradient Boosting Machines and Extreme Gradient Boosting Machines

3.3. Estimation

4. Data

4.1. Datasets

4.1.1. Real-World Datasets

UCSD-FICO Competition

Université Libre de Bruxelles

4.1.2. Simulated Datasets

PaySim

BankSim

4.2. Problems of Imbalanced Data and Data Sampling Techniques

4.2.1. Problem of Imbalanced Data

One of the biggest challenges faced in detecting fraudulent transactions is the one of unbalanced class sizes, with legitimate class outnumbering vastly the fraudulent one (Bhattacharyya et al. 2011). The application of data-sampling techniques has been widely used in the literature with various results when combined with different algorithms, as when such a problem occurs, it could hinder the model performances (Van Hulse, Khoshgoftaar, and Napolitano 2007). Moreover, in our particular case, the cost of missclassifying the minority class could prove to be a lot more costly than predicting wrongly the majority one.

4.2.2. Data Sampling Techniques

Random Oversampling (ROS)and Random UnderSampling (RUS)

The two techniques are the simplest and most common (Van Hulse, Khoshgoftaar, and Napolitano 2007). In minority oversampling (ROS), the observations from the minority group are randomly duplicated in order to balance the dataset. In majority undersampling (RUS), the aim is the same, but it is achieved by randomly removing observations of the majority class.

SMOTE

The Synthetic Minority Oversampling Technique (SMOTE), proposed by Chawla et al. (Chawla et al. 2002), artificial minority instances are created not simply through duplication, but rather with the extrapolation between preexisting observations. The technique starts by taking into account the k nearest neighbourhoods to a minority observation for every instance from that class. Then, the artificial observation are created, taking into account just a part of the nearest neighbours or all of them (with respect to the desired oversampling specification).

5. Results

6. Further Improvements

7. Conclusion

8. References

Batuwita, Rukshan, and Vasile Palade. 2013. "Class Imbalance Learning Methods for Support Vector Machines."

Bhattacharyya, Siddhartha, Sanjeev Jha, Kurian Tharakunnel, and J Christopher Westland. 2011. "Data Mining for Credit Card Fraud: A Comparative Study." *Decision Support Systems* 50 (3). Elsevier: 602–13.

Chawla, Nitesh V, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. 2002. "SMOTE: Synthetic Minority over-Sampling Technique." *Journal of Artificial Intelligence Research* 16: 321–57.

Cortes, Corinna, and Vladimir Vapnik. 1995. "Support-Vector Networks." *Machine Learning* 20 (3). Springer: 273–97.

Ngai, EWT, Yong Hu, YH Wong, Yijun Chen, and Xin Sun. 2011. "The Application of Data Mining Techniques in Financial Fraud Detection: A Classification Framework and an Academic Review of Literature." *Decision Support Systems* 50 (3). Elsevier: 559–69.

Van Hulse, Jason, Taghi M Khoshgoftaar, and Amri Napolitano. 2007. "Experimental Perspectives on Learning from Imbalanced Data." In *Proceedings of the 24th International Conference on Machine Learning*, 935–42. ACM.