Improving the Data Quality for Credit Card Fraud Detection

Rongrong Jing[1,2], Hu Tian[1,2], Yidi Li[2], Xingwei Zhang[1,2] ,Xiaolong Zheng[1,2],Zhu Zhang[1,3], Daniel Zeng[1,2,3]
[1]The State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100190, China
[3]Shenzhen Artificial Intelligence and Data Science Institude (Longhua), Shenzhen 518129, China
{ jingrongrong2019, tianhu2018, zhangxingwei2019, xiaolong.zheng, zhu.zhang@ia.ac.cn, dajun.zeng }@ia.ac.cn, liyidi19@mails.ucas.ac.cn

***Abstract*—Label imbalance and data missing are two major challenges in the problem of credit card fraud detection. However, existing matrix completion algorithms are generally difficult and cannot be easily applied to real-world credit card fraud detection since the scale of the normally used dataset is oversized. In this paper, we develop a spectral regularization algorithm to complete the large-scale sparse matrices, and further utilize an over-sampling algorithm to tackle the problem of the imbalance between positive and negative samples. Experimental results on a real-world dataset demonstrate that our model can outperform the state-of-the-art baseline methods. The proposed method could also be extended to other large-scale scenarios where data is missing or labels are imbalanced.**

Keywords—Credit card fraud Detection, Imbalanced data, Sparse matrix completion

I. Introduction

Nowadays, with the rapid development of mobile payment, credit card payment has become one of the dominant consumption ways. However, the rise of such cashless electronic financial transactions has also led to the emergence of new financial fraud behaviors [25]. Fraudsters often steal data and information from traders to make illegal deals in a short period. Therefore, financial institutions should use various methods in the real world and cyberspace to improve fraud detection performance on credit card consumption and protect the interests of customers.

Credit card fraud transactions mainly have the following two characteristics: (1) Time aggregation. Fraudsters are limited by the time of activity. Since a cardholder will freeze his/her card as soon as a suspicious transaction is discovered, fraudsters must reach the credit limit within a short time. This means that fraudulent transactions will be exposed because of the transaction amount within a limited time. (2) Spatial aggregation. Fraudsters are limited by the cost of transaction equipment and merchants, so fraudsters will use credit cards frequently to deal with few fixed merchants, which is spatially different from normal transactions.

Researchers have used statistical learning methods to solve these problems and achieved fairly good detection results. In fact, credit card fraud detection still faces several challenges: (1) The proportion of positive and negative samples in the actual dataset is extremely imbalanced. If targeted processing methods are not used, the classifier will inevitably bias the majority of the dataset, and even produce a unilateral classifier that has a negative impact on credit card fraud detection. The imbalance of the labeled data is not unique to credit card fraud detection problem but also common in various practical applications such as accident detection and disease risk prediction [26, 27]. A reasonable solution to the data imbalance in credit card fraud prediction will provide a valuable reference for solving similar problems in a wide range of practical applications. (2) Due to the limitations of sampling equipment and user privacy, there is a large percentage of data missing in the dataset, with the same situation as the Netflix dataset where the features are sparse. The success of fraud detection model generally depends on data representation. These deficiencies will lead the performance of the model different from the actuality, that is, the reduction of segmentation accuracy. Researchers choose to discard these data or adopt some data completion algorithms to increase the integrity of the data [26].

To address these limitations, we proposed a novel method to improve the data quality for credit card fraud detection by combining the data completion and sampling. Specifically, we firstly use the spectral regularization algorithm to complete the sparse matrix of the dataset to reduce the impact of missing data. Further, we adopt an over-sampling algorithm to solve the shortcomings of the imbalance of positive and negative samples, so that the proportion of samples in the dataset is consistent. We compare the performance of discarding missing values and traditional matrix completion algorithms with spectral regularization algorithms on the dataset. Then we compare the methods of different sample ratios through over-sampling and under-sampling. Last, we evaluated our method and baselines on a public credit card fraud detection dataset. Experiments show that the proposed spectral regularization and over-sampling algorithms can effectively improve various evaluation metrics on the credit card fraud detection dataset.

II. Related works

In this section, we discuss the existing research on dealing with imbalanced and missing data, which are closely related to this work. Then we review the research about card fraud.

A. Data Imbalance

There are two solutions to solving the problem of imbalanced training data: under-sampling majority type and over-sampling minority type. The one-sided selection proposed by Kubat et al. [1] is a typical sampling-based method, which can remove most training examples of noise, boundary, and redundancy. However, these steps usually only remove a small majority of instances, so they may not help much if the ratio of majority to minority exceeds 100:1. Multi-classifier training proposed by Chan et al. [2] and Bagging proposed by Breiman [3] are two other under-sampling methods. These methods do not directly deal with noise and boundary data but use a large number of sub-classifiers to reduce the prediction variance.

Over-sampling is the opposite of under-sampling. It replicates or inserts a few examples for reducing data imbalance. The over-sampling method assumes that the neighborhood of a positive instance is positive, and the instance between two positive instances is also positive. However, similar assumptions may depend on data. Chawla et al. [4] proposed an over-sampling algorithm SMOTE, which is based on interpolation to synthesize new samples for minority classes; Han et al. [5] proposed an optimized Borderline-SMOTE algorithm to optimize the performance of SMOTE.

B. *Missing Data Completion*

There is an army of missing data in credit card data due to privacy issues and collection process irregularities. How to tackle a high-dimensional sparse matrix for completing missing data has always been a research problem. For the matrix completion problem, Fazel [6] proved that the problem can be transformed to minimize the kernel norm of the matrix. Under the constraints of the missing rate, matrix rank, and sampling plan, the missing data of low-rank matrices can be recovered accurately with high probability. Toh et al. [7] proposed to relax the matrix completion problem into an unconstrained matrix least squares model in the presence of noise. Cai et al. [8] proposed to minimize the F norm of the matrix at the same time to ensure the low rank of the restored matrix. Rechet et al. [9] proved that the matrix restoration problem based on the kernel norm can be transformed into a matrix factorization model under certain conditions. Chiang et al. [10] proposed a matrix completion model with noisy edge auxiliary information. Wen et al. [11] proposed a nonlinear continuous over-relaxation iteration method (SOR) which is used to solve large-scale completion models based on decomposition methods. Mazumder et al. [12] proposed a spectral regularization algorithm to complete large-scale sparse matrices. Koren [13] used the restricted Boltzmann machine (RBM) and gradient boosting decision tree (GBDT) to solve the sparse matrix completion problem. We adopted the spectral regularization algorithm to complete the sparse matrix, which can effectively reduce the impact of missing data at a high rate of speed.

C. *Card Fraud Detection*

Kokkinaki [14] proposed the concept of similar trees to detect fraud. Randhawa et al. [15] proposed hybrid methods that used AdaBoost and majority voting methods to achieve good accuracy in detecting fraud cases in credit cards. Shailesh and Bamnote [23] used a hidden Markov model to simulate the sequence of operations in credit card transaction processing. A method of combining SVM, Random forests, Logistic regression, Self-Organizing Map Neural Network (SOMNN), Genetic Algorithm with behavior-based technique, and Hidden Markov Model (HMM) has been attempted [16,17].

Fu et al. [18] proposed a CNN-based fraud detection framework to capture the intrinsic patterns of fraud behaviors learned from labeled data. Sweers et al. [19] proposed a credit card fraud detection method using auto-encoder and variational auto-encoder based anomaly detection. Stolfo et al. [20] used meta-learning to solve credit card detection. Chan et al. [21] also proposed a multi-classifier meta-learning approach for fraud detection. Syeda et al. [22] proposed a parallel granular neural network (GNN) to speed up the data mining and knowledge discovery process for credit card fraud detection. GNN gives fewer average training errors with a larger amount of past training data. we adopted XGBoost as the main solution for prediction task because its tree structures with tree pruning is fit for this prediction task.

III. METHODS

In this section, we first introduce the principle and formulation of matrix completion, and present two solutions for imbalanced data sampling, SMOTE, and Bootstrapping. Then we introduce the XGBoost for the card fraud prediction task. The data goes through two stages of processing for eventual prediction task. With a simple but efficient algorithm for minimizing the reconstruction error, the missing data has been completed sufficiently. Besides, two sampling methods with low complexity and high computation efficiency are very suitable for this real-time data processing application. XGBoost performs great advantages because of its regularization for avoiding overfitting and reducing computation in the card fraud prediction task.

A. *Matrix Completion*

In many applications, data can be represented by a matrix $X_{m\times n}$, where only a relatively small number of entries are observed. *m* and *n* are often very large, and the sparse degree of the matrix may be pretty high. The problem called Matrix Completion is to complete the sparse matrix by predicting missing entries with the observed entries. To illustrate its significance, here is the example of credit card transaction records. There will be a lot of transactions (as the rows) with a lot of attributes (as the columns), such as trading time, platform, device, etc. These records are incomplete. The task is to predict the value of these attributes that have not yet been filled.

To complete the entire matrix, we need some additional constraints, and the most common one is to assume that the matrix $Z_{m\times n}$ is of low rank, that is, $Z \approx V_{m\times k} G_{k\times n}$, where $k \ll \min(n, m)$. Typically, we view the observed entries in X as the corresponding entries from Z contaminated with noise.

Given a matrix $X_{m\times n}$, let $\Omega \subset \{1, \dots, m\} \times \{1, \dots, n\}$ denote the indices of observed entries. Consider the following optimization problem:

$$\begin{aligned} &\text{minimize} \quad \operatorname{rank}(Z) \\ &\text{subject to} \quad \sum_{(i,j)\in\Omega} \left(X_{ij} - Z_{ij}\right)^2 \le \delta \end{aligned} \tag{1}$$

where $\delta \ge 0$ is a regularization parameter controlling the tolerance in training error. But there are two main problems: (1) the above problem is non-convex and NP-hard; (2) the dimensions of the matrix $X_{m\times n}$ are always very large. For a fully-observed Z, the solution is given by a truncated singular value decomposition (truncated SVD) of X. The following method can make the problem convex with a small modification:

$$\begin{aligned} &\text{minimize } \|Z\|_* \\ &\text{subject to } \sum\nolimits_{(i,j)\in\Omega} \left(X_{ij} - Z_{ij}\right)^2 \le \delta \end{aligned} \tag{2}$$

where $\|Z\|_*$ is the nuclear norm or the sum of the regular values of Z. By the properties of SVD decomposition, $\operatorname{rank}(Z) \le k$ is equivalent to $\|Z\|_* \le k$. Under many situations, the nuclear norm is an effective convex relaxation to the rank constraint. Using modern convex optimization software, the optimization of (2) is a semi-definite programming problem and can be solved efficiently for small problems. However, due to the large dimensions of the matrix $X_{m\times n}$, it will cost a lot of time and memory prohibitively. Equivalently, (2) can be reformulated in Lagrange form [13]:

$$\underset{Z}{\text{minimize}}\frac{1}{2}\sum_{(i,j)\in\Omega}\left(X_{ij}-Z_{ij}\right)^2+\lambda\|Z\|_* \qquad (3)$$

Where $\lambda \geq 0$ is a regularization parameter controlling the nuclear norm of the minimizer $\hat{Z}_\lambda$, and there is a one-to-one mapping between $\delta \geq 0$ and $\lambda \geq 0$ over their active domains. To solve the equation (3) faster, we adopted the numerical scheme Soft-impute [12].

B. Imbalanced Data Sampling

- Synthetic Minority Over-sampling Technique (SMOTE)

We regard the dataset as an imbalanced one if the classes are not approximately equally represented. Imbalance on the order of 100:1 is prevalent in fraud detection and even up to 10000:1 in other data mining applications. The performance of machine learning algorithms is typically evaluated using predictive accuracy. However, this is not appropriate when the data is imbalanced and/or the costs of different errors vary markedly. There are two strategies for the class imbalance problem, that is, sampling and cost-sensitive learning. The sampling methods are also divided into over-sampling and under-sampling. The SMOTE algorithm [4] is a commonly used method in over-sampling. It is an improved scheme based on a random over-sampling algorithm. Because random over-sampling adopts the strategy of simply copying samples to increase a few samples, it is easy to cause the problem of model overfitting. The basic idea of SMOTE is to analyze the minority samples and add the new samples into the dataset according to the minority samples.

A sampling ratio varies to determine the sampling multiplying power n according to the sample imbalance ratio. In the binary classification problem, sample imbalance ratio usually means the quotient of samples in higher percentage and that in lower percentage. For each minority sample x, a number of samples are randomly selected from its k-nearest neighbors, assuming that the chosen neighbor is x_n. For each x_n, we build a new sample x_{new}, following the formula below:

$$x_{new} = x + rand(0,1) * |x - x_n| \qquad (4)$$

- Bootstrapping

Bootstrapping is any test or metric that uses random sampling with replacement, and falls under the broader class of resampling methods. It provides a method other than confidence intervals to estimate a population parameter. Bootstrapping resamples a single dataset to create lots of simulated samples. This process allows us to calculate standard errors, construct confidence intervals, and perform hypothesis testing for numerous types of sample statistics. Bootstrap methods are alternative approaches to traditional hypothesis testing. They are easier to understand and valid for more conditions.

C. Prediction Model

In this task, we use a classic method of ensemble learning, XGBoost [29]. This model is an integrated tree approach that employs a gradient descent method to enhance the weak learner (usually CART) principles to form a strong classifier. This model is widely used in machine learning and data mining challenges.

XGBoost is a decision-tree-based gradient boosting framework which integrates tree models together to form a strong classifier. For a given data $D = \{(x_i, y_i)\}(x_i \in \mathbb{R}^n, y_i \in \mathbb{R}^n)$, and K additive functions, the tree ensemble model uses the summation method:

$$\hat{y}_i = \phi(x_i) = \sum_{k=1}^{K} f_k(x_i)\,, f_k \in \mathcal{F} \qquad (5)$$

where $\mathcal{F}$ is the space of CART trees, each f_k has its own independent tree structure q and leaf weight ω.

XGBoost improves the basic Gradient Boosting Machines framework through system optimization and algorithm enhancement. In terms of algorithmic enhancements, when minimizing the target function, XGBoost penalizes complex models by using L1 and L2 regularization in order to prevent overfitting:

$$\mathcal{L}(\phi) = \sum_i l(\hat{y}_i, y_i) + \sum_k \Omega(f_k) \qquad (6)$$

where $\Omega(f) = \gamma T + \frac{1}{2}\lambda\|\omega\|^2$. XGBoost accepts sparsity features of inputs naturally by automatically learning the best missing value based on the training loss, and handles the different types of sparsity patterns in the data more effectively. In terms of system optimization, XGBoost parallelizes the sequential tree building process while using tree pruning to greatly improve computational performance. Hardware resources are effectively utilized by allocating internal buffers within each thread to store cache awareness of gradient statistics.

IV. Experiment

A. Dataset

To evaluate the performance of our proposed method for credit card fraud detection, we select the dataset from Kaggle (https://www.kaggle.com/c/ieee-fraud-detection/data). Vesta Corporation provided the dataset for this task. Vesta Corporation is the forerunner in guaranteed e-commerce payment solutions. When a user uses a credit card to make a transaction, the Internet will record the user's identity information (ID, time, location, eta.), the network connection information associated with the transaction (IP, ISP, Proxy, etc.), and digital signature information (UA, browser, os, version, etc.). They are collected by Vesta's fraud protection system and digital security partners.

The data is separated into two datasets: information about the identities of the customers and transaction information. This data seems to have been cobbled together from various vendors without filtering out the duplicate columns. There are 590540 transactions in the training set and each transaction has 433 features. However, as shown in Fig.1, there are only 3.5% of fraud transactions in the dataset. The severe imbalance of data labels will lead to the tendency of the classifier to predict fraud-free behavior.

As shown in Fig.1, there are 3.5% of fraud transactions in the dataset, and there are 590540 transactions in the training set. The severe imbalance of data tags leads to the tendency of the classifier to predict fraud-free behavior.

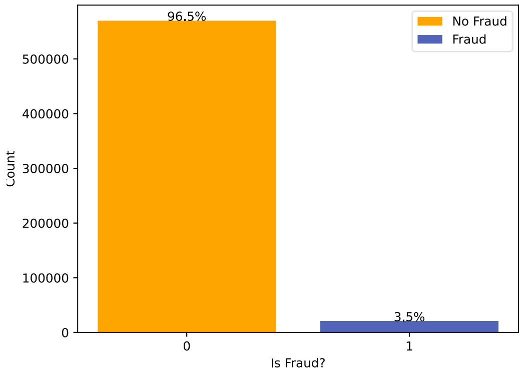

Fig. 1. The imbalanced fraud labels

As mentioned earlier, there are nearly 100 million data vacancies. As shown in Fig.2, almost all of the data's features are incomplete. More than 400,000 transactions are missing nearly half the features. The first thing we need to do is to complete the dataset by spectral regularization matrix completing method, we call this method SRMC.

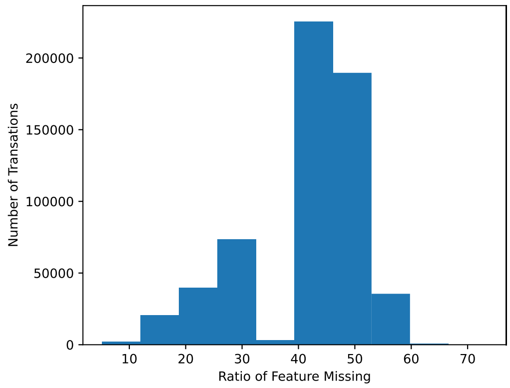

Fig. 2. The degree of features missing.

B. Baselines

We compared the following baselines with our method.

- The common data completing method that we choose to fill missing values with means (for float or int type) or zeros (for object type) is named CMC.
- The Under-Sampling method is to select a part of data from the majority set and recombine it with a few sets to form a new dataset. The under-sampling takes the idea of random under-sampling, which is to randomly select some samples from the majority class and eliminate them.
- Logistic Regression (LR) is a kind of generalized linear model, which is generally used to solve binary classification problems. The essence of LR is to assume that the data obey Logistic distribution and then use maximum likelihood estimation as parameter estimation. Logistic distribution is a continuous probability distribution.
- Support Vector Machine (SVM) is a kind of supervised learning algorithm, which is widely used for both classification and regression problems. For linear separable datasets, SVM is to find an optimal classification hyperplane in the feature space. As for the undivided linear sample space, SVM maps the undivided linear samples to another high-dimensional linear space by the kernel.

C. Parameter Settings

There are three types of hyper-parameters in our proposed method: the matrix completion hyper-parameters, the sampling hyper-parameters, and the classifier hyper-parameters. The proportion of the under-Sampling reduction is set to 80% to make the number of two labels equal while the proportion of the SMOTE and Bootstrapping increasing is set to 80 after practicing. For SMOTE, we choose the minority strategy to resample only the minority class. We set the neighbors to 5 and random state to 0.45. For LR, we choose L2 regularization and newton-cg solver and set the number of iterations to 1000. For SVM, we use the Gaussian kernel function and set the number of iterations to 500 because of the huge amount of computation.

D. Results

Table 1 shows the performances of our proposed method and baselines. For missing input data, we choose two completion methods: the commonly used method of number completion and our method. For the problem of sample imbalance, we also use two optimization schemes: under-sampling and over-sampling. Finally, we used three classification decision-makers: Logistic Regression, Support Vector Machine (SVM), and XGBoost to perform experiments, and evaluated the model with the f1-score of Fraud category (Fra.) and Non-Fraud category (NoFra.). Experimental results demonstrate that our model can outperform the state-of-the-art baseline methods.

Table1. The performance comparison of various methods.

Model	**XGBoost(F1)**		**LR(F1)**		**SVM(F1)**	
	NoFra.	***Fra.***	***NoFra.***	***Fra.***	***NoFra.***	***Fra.***
Imbalance + CMC	**0.99**	0.71	0.98	0.01	0.73	0.16
Imbalance + SRMC	**0.99**	0.59	0.99	0.36	0.74	0.36
Under-Sampling + CMC	0.85	0.85	0.73	0.69	0.52	0.67
Under-Sampling + SRMC	0.88	0.88	0.79	0.78	**0.79**	0.47
Bootstrapping + CMC	0.95	0.95	0.71	0.63	0.56	0.61
Bootstrapping + SRMC (Our model)	0.97	**0.97**	**0.80**	**0.79**	0.77	**0.72**
SMOTE + SRMC (Our model)	0.97	**0.97**	0.77	0.76	0.76	0.71

We used the imbalanced data and common completion method as the baseline. Because the imbalanced data is not processed, the f1-score of the negative samples in this scheme is higher than the other data-processing methods under the three algorithms. But the f1-score of the positive samples is relatively low. This is due to the high proportion of negative samples in the imbalanced data. It can be seen that the scores of the negative samples in the two control groups using the original imbalanced dataset are very high. A large number of negative samples make the precision and recall of the model close to 1. For positive samples, the model's ability to judge positive is relatively weak, so the recall is low, and the f1-score is lower than the negative samples. For LR and SVM with weak classification performance, it is more prominent. Regardless of the subsequent under-sampling or over-sampling algorithms, the f1-score of positive samples does not vary from that of negative samples greatly. Therefore, imbalanced samples have a great impact on the performance of data mining algorithms, and it is necessary to preprocess the samples before training. In particular, comparing the first two sets of experiments, the matrix completion algorithm we used reduces the f1-score of positive samples on the ensemble learning algorithm XGBoost. For weaker classifiers, matrix completion can improve the ability to classify positive samples to a certain extent.

In the third and fourth groups of experiments, we adopted an under-sampling method for negative samples. In each iteration, negative samples were randomly under-sampled to make the ratio of positive and negative samples equal. Since the number of positive samples is extremely rare, the entire training set only accounts for 7% of the original dataset after under-sampling. Although resampling is performed during each iteration, at the formal of total iterations, it is easy to cause under-fitting. It can be seen that the f1-score of negative samples in the third and fourth groups of experiments is lower than the rest of the groups as a whole. However, since the imbalance of positive and negative samples has been solved after re-sampling, the f1-score of the positive samples has been greatly improved, numerically close to the negative samples.

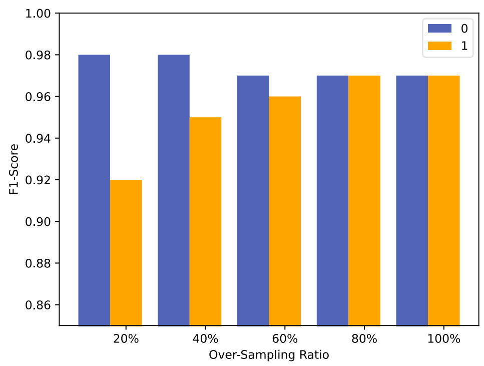

Fig. 3. The over-sampling ratio of SMOTE method

We adopted the over-sampling method on the positive samples in the last three groups of experiments. The fifth group used the normal matrix filling method, and the six or seven groups used Bootstrapping and smooth over-sampling methods based on our matrix completion method. The comparison shows that the over-sampling method is better than the under-sampling method in dealing with this kind of data imbalance problem. The increase in the number of positive samples makes its f1-score of both positive and negative samples significantly increase. Similarly, since the sample equalization problem has been solved, the scores of the positive and negative categories are close to each other. We also performed experiments on sampling rates of 20%, 40%, 60%, 80%, and 100%. The results are shown in Fig.3. While the over-sampling rate is increasing, and that of the negative samples is decreasing. The score of the positive samples has an upward trend within a certain range. In particular, the over-sampling of data can easily cause over-fitting problem.

In practical applications, it is a safer and more effective way to use the combination of over-sampling and under-sampling. The performance of the model fluctuates during the process of changing the sampling rate respectively. In the training process of subsequent models, the regular term for the sampling rate can be considered to learn the most appropriate one. Although the f1-score of positive and negative samples tends to be the same when the sampling rate is 100%, the sampling rate should be selected based on the current situation and comprehensive consideration of the cost of misjudgment in different situations.

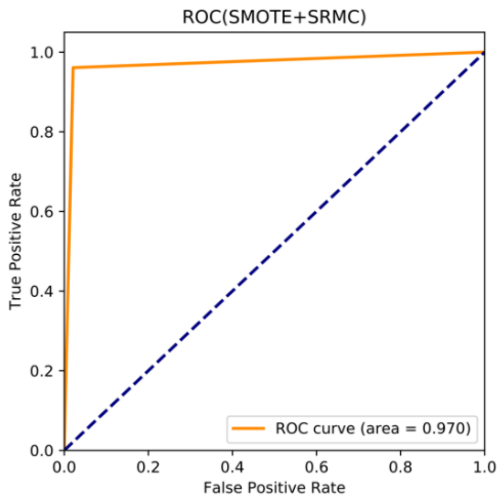

Fig. 4. ROC of the incorporation of SMOTE and SRMC

In general, the matrix completion algorithm we used can effectively improve the credit card fraud detection performance on the dataset. Compared with the baselines, the score of the positive samples is greatly improved without reducing that of the negative samples. The ROC curve of the model is shown in the Fig.4, and we can see that our model has pretty good performance.

V. Conclusions

In this paper, we proposed a novel method for credit card fraud detection. Specifically, we combined a spectral regularization algorithm of completing large-scale sparse matrices and the over-sampling algorithm in our method. Our method could solve the challenges of data sparsity and class imbalance. Experiments show that our method is superior to competitive baseline methods on the credit card dataset from Kaggle. Our method could be extended to other large-scale scenarios where data is sparse or labels are imbalanced.

Acknowledgment

This work is supported by the Ministry of Health of China under Grant No.: 2017ZX10303401-002 and 2017YFC1200302, the Natural Science Foundation of China under Grant No.: 71472175, 71602184 and 71621002, and the National Key Research and Development Program of China under Grant No. 2016QY02D0305.

References

[1] Kubat M, Matwin S. Addressing the curse of imbalanced training sets: one-sided selection[C]//Icml. 1997, 97: 179-186.

[2] Chan P K, Stolfo S J. Learning with non-uniform class and cost distributions: Effects and a distributed multi-classifier approach[C]//In Workshop Notes KDD-98 Workshop on Distributed Data Mining. 1998.

[3] Breiman L. Bagging predictors[J]. Machine learning, 1996, 24(2): 123-140.

[4] Chawla N V, Bowyer K W, Hall L O, et al. SMOTE: synthetic minority over-sampling technique[J]. Journal of artificial intelligence research, 2002, 16: 321-357.

[5] Han H, Wang W Y, Mao B H. Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning[C]//International conference on intelligent computing. Springer, Berlin, Heidelberg, 2005: 878-887.

[6] Fazel S M. Matrix rank minimization with applications[J]. 2003.

[7] Shen Z, Toh K C, Yun S. An accelerated proximal gradient algorithm for frame-based image restoration via the balanced approach[J]. SIAM Journal on Imaging Sciences, 2011, 4(2): 573-596.

[8] Cai J F, Candès E J, Shen Z. A singular value thresholding algorithm for matrix completion[J]. SIAM Journal on optimization, 2010, 20(4): 1956-1982.

[9] Recht B, Fazel M, Parrilo P A. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization[J]. SIAM review, 2010, 52(3): 471-501.

[10] Chiang K Y, Hsieh C J, Dhillon I S. Matrix completion with noisy side information[C]//Advances in Neural Information Processing Systems. 2015: 3447-3455.

[11] Wen Z, Yin W, Zhang Y. Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm[J]. Mathematical Programming Computation, 2012, 4(4): 333-361v.

[12] Mazumder R, Hastie T, Tibshirani R. Spectral regularization algorithms for learning large incomplete matrices[J]. The Journal of Machine Learning Research, 2010, 11: 2287-2322.

[13] Koren Y. The bellkor solution to the netflix grand prize[J]. Netflix prize documentation, 2009, 81(2009): 1-10

[14] Kokkinaki A I. On atypical database transactions: identification of probable frauds using machine learning for user profiling[C] //Proceedings 1997 IEEE Knowledge and Data Engineering Exchange Workshop. IEEE, 1997: 107-113.

[15] Randhawa K, Loo C K, Seera M, et al. Credit card fraud detection using AdaBoost and majority voting[J]. IEEE access, 2018, 6: 14277-14284.

[16] Carminati M, Caron R, Maggi F, et al. BankSealer: A decision support system for online banking fraud analysis and investigation[J]. computers & security, 2015, 53: 175-186.

[17] Srivastava A, Kundu A, Sural S, et al. Credit card fraud detection using hidden Markov model[J]. IEEE Transactions on dependable and secure computing, 2008, 5(1): 37-48.

[18] Fu K, Cheng D, Tu Y, et al. Credit card fraud detection using convolutional neural networks[C]//International Conference on Neural Information Processing. Springer, Cham, 2016: 483-490.

[19] Sweers T, Heskes T, Krijthe J. Autoencoding Credit Card Fraud[J]. Bachelor Thesis, 2018.

[20] Stolfo S, Fan D W, Lee W, et al. Credit card fraud detection using meta-learning: Issues and initial results[C]//AAAI-97 Workshop on Fraud Detection and Risk Management. 1997: 83-90.

[21] Chan P K, Fan W, Prodromidis A L, et al. Distributed data mining in credit card fraud detection[J]. IEEE Intelligent Systems and Their Applications, 1999, 14(6): 67-74.

[22] Syeda M, Zhang Y Q, Pan Y. Parallel granular neural networks for fast credit card fraud detection[C]//2002 IEEE World Congress on Computational Intelligence. 2002 IEEE International Conference on Fuzzy Systems. FUZZ-IEEE'02. Proceedings (Cat. No. 02CH37291). IEEE, 2002, 1: 572-577.

[23] Dhok S S, Bamnote G R. Credit card fraud detection using hidden Markov model[J]. International Journal of Soft Computing and Engineering (IJSCE), 2012, 2(1): 231-237.

[24] Panigrahi S, Kundu A, Sural S, et al. Credit card fraud detection: A fusion approach using Dempster–Shafer theory and Bayesian learning[J]. Information Fusion, 2009, 10(4): 354-363.

[25] Bhattacharyya S, Jha S, Tharakunnel K, et al. Data mining for credit card fraud: A comparative study[J]. Decision Support Systems, 2011, 50(3): 602-613.

[26] Parsa A B, Taghipour H, Derrible S, et al. Real-time accident detection: coping with imbalanced data[J]. Accident Analysis & Prevention, 2019, 129: 202-210.

[27] Khalilia M, Chakraborty S, Popescu M. Predicting disease risks from highly imbalanced data using random forest[J]. BMC medical informatics and decision making, 2011, 11(1): 51.

[28] Freedman D A. Bootstrapping regression models[J]. The Annals of Statistics, 1981, 9(6): 1218-1228.

[29] Chen T, Guestrin C. Xgboost: A scalable tree boosting system[C]//Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining. 2016: 785-794.