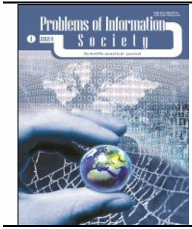


www.jpis.az

15 (1)
2024

Analyzing credit card fraud cases with supervised machine learning methods: logistic regression and Naive Bayes

Naila Habibullayeva[1], *Behnam Kiani Kalejahi*[2]

[1,2]Computer Science Department, Khazar University, Mahsati str. 41, AZ1096, Baku, Azerbaijan

[1]naila.habibullayeva@khazar.org, [2]bkiani@khazar.org

orcid.org/0000-0002-7118-0382

ARTICLE INFO

http://doi.org/10.25045/jpis.v15.i1.06

Article history:
Received 25 August 2023
Received in revised form
27 October 2023
Accepted 29 December 2023

Keywords:

Credit Card Fraud
Supervised Machine Learning
Logistic Regression modeling
Knime Software
Naive Bayes modelling
Imbalanced Classification

ABSTRACT

Frauds involving credit cards are simple and effortless to target. With the rise of online payment credit cards have had a huge role in our daily life and economy for the past two decades and it is an important task for companies to identify fraud and non-fraud transactions. As the number of credit cards grows every day and the volume of transactions increases quickly in tandem, fraudsters who wish to exploit this market for illegitimate gains have come to light. Nowadays, it's quite easy to access anyone's credit card information, which makes it simpler for card fraudsters to do their crimes. Thanks to advances in technology, it is now possible to determine whether information gained with malicious intent has been used by looking at the costs and time involved in altering account transactions. The Credit Card Fraud analysis data set, which is obtained from the Kaggle database, is used in the modeling process together with The Logistic regression method and Naive Bayes algorithms. Using the Knime platform, we are going to apply machine learning techniques to practical data in this study. The goal of this study is to identify who performed the transaction by examining the periods when people use their credit cards. The Logistic regression approach and the Naive Bayes method both had success rates of 99.83%, which is the highest. The two methods' results are based on Cohen's kappa, accuracy, precision, recall, and other metrics.

1. Introduction

Payments can be made using credit cards and POST devices used at shopping points, provided by banks to the people they serve. You can also withdraw cash from ATMs. Credit cards also make people's lives easier when it comes to paying their expenses in installments.

In this way, people reduce their monthly expenses by dividing them into a certain number of months instead of paying all at once. Thanks to its prevalence and strong infrastructure around the world, credit cards have become a payment tool that people can use easily and frequently in a very short time. In today's society, fraud on credit cards has considered a significant worry, with increased fraud in political agencies, corporate sectors, financial commerce, as well as other associations. The credit card is indeed an efficient and easy target for fraudsters since a significant volume of money may be stolen swiftly and without risk. Criminals perpetrate fraud on credit cards by stealing personal statistics including credit account values, banking information, and passwords. Fraudulent individuals attempt to constitute their malicious attack seem legal, making fraud reporting difficult. Credit card fraud has risen as a result of our society's growing reliance on the internet; yet, theft has grown not just internet but also offline. In 2022, global cybercrime expenses were $408.50 billion. To combat the problem, several corporations, such as VISA, are resorting to Machine learning solutions. Using machine learning to identify credit card fraud has several advantages, such as:

- Pattern classification
- Data processing efficiency
- Prediction accuracy

Although the use of certain data mining methods, the results in identifying credit card fraud are not particularly accurate. These expenses can be reduced only by detecting fraud with advanced algorithms, which is a promising mechanism for minimizing credit card fraud. As the use of the internet expands, the financial business issues credit cards.

In addition to this situation, many problems have arisen as the usage areas of credit cards have increased, and the reasons why people prefer them have increased. The most important problem that occurs when people use credit cards so much is that their information falls into the hands of other people and is misused. Credit card fraud can occur by copying an existing card exactly to a new card, or by stealing the information on the existing card from e-commerce sites and using it as the owner of the card or transferring money from it. Fraud with credit cards causes enormous financial losses for every nation on the planet. For this reason, certain analyses are made using data obtained from credit card transactions in the study, and as a result this analysis, it is aimed to prevent credit card fraud.

2. Literature Review

A plethora of traditional machine learning methods including Decision Tree, K-Nearest Neighbour (KNN), SVM, Logistic Regression, Random Forest, XGBoost, and other deep learning methods are applied to the process of the detection of credit card fraud. Including ANN and Logistic Regression, tree-based cooperative methods proved to be effective. From past work on this topic, it is revealed that it is important to balance the data as there is a large imbalance in the data set between fraud and non-fraud transactions. In this section, a significant number of works are presented.

Rimpal R. Popat et al. (2018) try an interesting approach. This team uses the end clustering technique to divide the data into three different groups according to the transaction amount. They use range partitioning for it. In the next step, they use the sliding window method by aggregating transactions into groups and then extracting patterns in cardholders' behavior. Minimum, maximum, and average transaction amounts made by cardholders are calculated. And whenever there is a new transaction made the new transactions are fed to the window while the old one is removed from it.

Xuan Shiyang et al. (2018) use supervised machine learning methods such as Random Forest, Stacking Classifier, and Logistic Regression and compared them with different metrics like Recall, Accuracy, Precision, etc. They eventually find out that Logistic Regression is the most accurate when it is picked as the base estimate of the r of Stacking classified followed by Random Forest and XGB classifier.

In another study, Tince Etlin Tallo et al. (2018) compare the advantages and drawbacks of fraud detection methods. For instance, they figure out that although the Hidden Markov Model is fast at detection, its accuracy is low, and it is not scalable for large data sets. On the other hand, Bayesian networks are good at accuracy while being expensive. Moreover, when it comes to artificial neural networks, they are portable and, effective in dealing with noisy data while being difficult to set up and having bad explanation capabilities. Another interesting point from this study is that they mentioned that there are no suitable metrics to evaluate the results of these prediction models as well as a lack of adaptive fraudulent insident of credit card detection systems.

The research on SVM, random forests, decision trees, and logistic regression by Navanushu Khare et al. (2020) is described. They experiment with a significantly unbalanced dataset. The effectiveness criteria include specificity, accuracy, sensitivity, and precision. According to the statistics, a logistic regression model is 97.7% accurate, Decision Trees are 95.5% true the random forest method is 98.6% accurate, and the classifier using SVM is 97.5% accurate. They determine that the Random Forest method is a highly efficient and precise method for detecting fraud. They also determine that, owing to the data imbalance problem, the SVM method do not execute any better in detecting fraud with credit cards.

To identify outliers, Vaishnavi Nath Dornadula et al. (2019) employ novel machine learning methods. That team use Local Outlier Factor and Isolation Forest algorithm which at the moment are considered the most popular outlier detection methods in the industry. Their accuracy is 99.6% while they have lower precision at 33%. The reason for the low precision in the data is a huge imbalance.

This study offers an approach with a different perspective in the literature. Considering that many existing studies focus on different objectives, we analyze two different types of supervised machine learning algorithms in detail. To perform our analyses, we chose a very simple and open-source platform, which provides a significant advantage in terms of accessibility and reusability of our work. This platform used makes the analysis process more understandable and applicable.

The main focus of our study shares a goal frequently encountered in other studies: to compare methods and determine which works better than the other. In this context, the algorithms and methodology we choose incorporate modern techniques and offer a unique analytical approach.

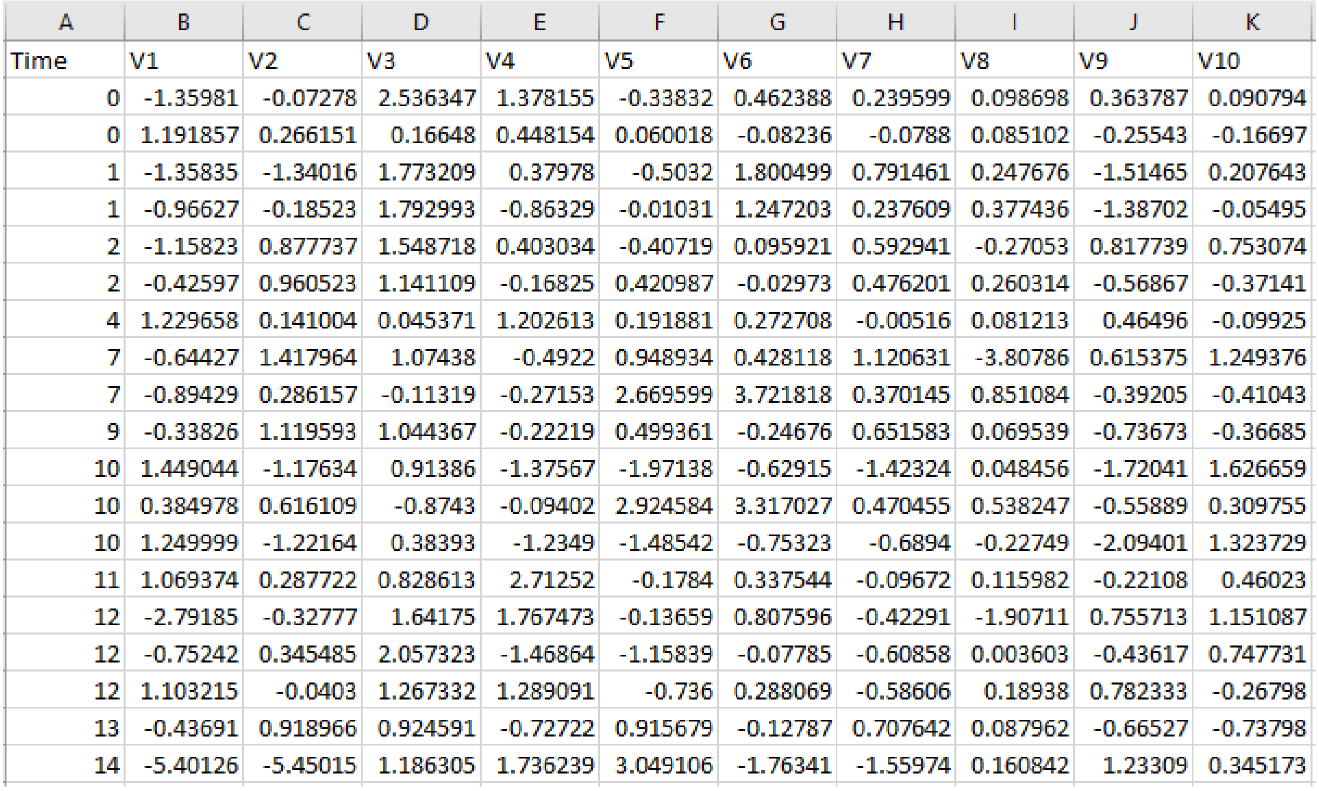
In particular, the up-to-dateness of the algorithms we use and the innovative features of the data set we examine make our study distinctive in the literature. Our results demonstrate the impressive potential of this modern approach and constitute an important reference source for future research.

3. Methods and Materials

The data set and methods employing to help detect fraud of credit card are explained in this section

3.1. The Dataset

The dataset used in this study is obtained from the Kaggle database and contains a total of 284,807 transactions, of which 492 are incorrect (error) (https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud). The data set has to be handled since it is so severely unbalanced before a model can be built Credit card companies need to be able to spot fraud financing card transactions to stop charging customers for goods they haven't purchased.



A	B	C	D	E	F	G	H	I	J	K
Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10
0	-1.35981	-0.07278	2.536347	1.378155	-0.33832	0.462388	0.239599	0.098698	0.363787	0.090794
0	1.191857	0.266151	0.16648	0.448154	0.060018	-0.08236	-0.0788	0.085102	-0.25543	-0.16697
1	-1.35835	-1.34016	1.773209	0.37978	-0.5032	1.800499	0.791461	0.247676	-1.51465	0.207643
1	-0.96627	-0.18523	1.792993	-0.86329	-0.01031	1.247203	0.237609	0.377436	-1.38702	-0.05495
2	-1.15823	0.877737	1.548718	0.403034	-0.40719	0.095921	0.592941	-0.27053	0.817739	0.753074
2	-0.42597	0.960523	1.141109	-0.16825	0.420987	-0.02973	0.476201	0.260314	-0.56867	-0.37141
4	1.229658	0.141004	0.045371	1.202613	0.191881	0.272708	-0.00516	0.081213	0.46496	-0.09925
7	-0.64427	1.417964	1.07438	-0.4922	0.948934	0.428118	1.120631	-3.80786	0.615375	1.249376
7	-0.89429	0.286157	-0.11319	-0.27153	2.669599	3.721818	0.370145	0.851084	-0.39205	-0.41043
9	-0.33826	1.119593	1.044367	-0.22219	0.499361	-0.24676	0.651583	0.069539	-0.73673	-0.36685
10	1.449044	-1.17634	0.91386	-1.37567	-1.97138	-0.62915	-1.42324	0.048456	-1.72041	1.626659
10	0.384978	0.616109	-0.8743	-0.09402	2.924584	3.317027	0.470455	0.538247	-0.55889	0.309755
10	1.249999	-1.22164	0.38393	-1.2349	-1.48542	-0.75323	-0.6894	-0.22749	-2.09401	1.323729
11	1.069374	0.287722	0.828613	2.71252	-0.1784	0.337544	-0.09672	0.115982	-0.22108	0.46023
12	-2.79185	-0.32777	1.64175	1.767473	-0.13659	0.807596	-0.42291	-1.90711	0.755713	1.151087
12	-0.75242	0.345485	2.057323	-1.46864	-1.15839	-0.07785	-0.60858	0.003603	-0.43617	0.747731
12	1.103215	-0.0403	1.267332	1.289091	-0.736	0.288069	-0.58606	0.18938	0.782333	-0.26798
13	-0.43691	0.918966	0.924591	-0.72722	0.915679	-0.12787	0.707642	0.087962	-0.66527	-0.73798
14	-5.40126	-5.45015	1.186305	1.736239	3.049106	-1.76341	-1.55974	0.160842	1.23309	0.345173

Fig.1. A part of Credit Card Fraud dataset

The dataset consists of September 2013 payment card operations made by users across Europe. In our data of operations that occurred throughout two days, we found 492 errors out of 284,807 operations. The sample is heavily biased with criminal activity accounting for 0.172% of all positive activities. All of the quantitative data parameters in the collection of data have completed PCA treatment. Regrettably, the disclosure of the initial characteristics and additional contextual details of the data is precluded by confidentiality concerns. The characteristics denoted as V1, V2, and so forth. The principal components derived from PCA are represented by V28, while the features 'Time' and 'Amount' remain untransformed.

3.2. The Performance Metrics

The Confusion matrix displays the node's particular output along with the amount of similarities in every single cell. Correctness facts are displayed in a separate column. The results include the average accuracy, Cohen's kappa, recall, precision, sensitivity, preciseness, the F-value, and the following: true, false, positive, and negative.

Accuracy: The ratio of accurate forecasts to all alternative guesses is used to compute accuracy, which is one of the most straightforward classification variables.

$$Accuracy = \frac{Number\ of\ correct\ prediction}{Total\ number\ of\ prediction} \quad (1)$$

Precision: A metric that quantifies the degree of correctness of a classification or prediction model. The term "precision" refers to the proportion of properly predicted positive cases, to the overall amount of anticipated positive instances, including comprises both correct and incorrect positives, in the model's output. Put another way, accuracy is a measurement of the ratio of actual positive situations to all of the scenarios that are projected to be positive.

A high level of precision denotes that the layout exhibits a superior competence to properly determine true cases despite the fact minimizing the occurrence of false positives in its output. Conversely, a diminished level of precision implies that the model exhibits an elevated frequency of false positives, thereby resulting in erroneous or deceptive outcomes.

Recall: The concept of recall pertains to the degree of comprehensiveness exhibited by a classification or prediction model. The term "precision" refers to a statistical metric that calculates the percentage of correctly anticipated positive situations, or "true positives" associated to the entirely number of positive instances that were either correctly identified or missed by the model, which includes both true positives and "false negatives." Stated differently, recall is a performance metric that quantifies the ratio of true cases that are accurately detected by the method.

F1 score: The F1 score is a measure of a test's accuracy—the harmonic mean of precision and sensitivity. It can have a maximum of 1 (perfect precision and sensitivity) and a minimum of 0. In general, it is a measure of the accuracy and robustness of your model. The formula for the F1 score is as follows.

$$F1\ score = \frac{2\ x\ recall\ x\ precision}{recall+precision} \quad (2)$$

Cohen's Kappa: Cohen's Kappa is a widely used measure for quantifying the agreement between two raters on the same nominal or ordinal construct. The measure performs better than routine methods such as percentage of agreement or linear regression, providing a more reliable insight into the consistency of relative ratings or diagnoses. It offers a robust approach for evaluating inter-rater agreement, using the range of 0 (Chance agreement) to 1 (Perfect agreement) wherein higher values closer to 1 indicate higher levels of agreement. This metric proves invaluable for researchers seeking to access systematic agreement between raters and consequently is used ubiquitously in evaluation studies.

Confusion matrix: It is a performance measure for a machine learning classification problem. It is a table containing 4 different combinations of predicted and actual values. These 4 combinations in the confusion matrix are True Positive, False Positive, True Negative and False Negative.

3.3. *The Logistic Regression method*

One of the methods used in the model in the study is the Logistic regression model. Thirty percent of the material set is utilized for testing, while seventy percent is applied for training. Logistic regression method is a popular and simple machine learning approach that works well for classifying data into two groups. It is easy to use and might be the beginning point for any sort of linear problem. Machine learning may benefit from its basic notions as well. A logistic regression model is a statistical technique used to forecast the probability of a discrete occurrence. Features of Logistic Regression:

- In this method, the reliant parameter has a Bernoulli distribution.
- The most remarkable, likelihood approach is used for assessment.
- In fact, there is no coefficient squared for determining demonstrating efficiency; instead, Congruence and KS-Statistics are used.

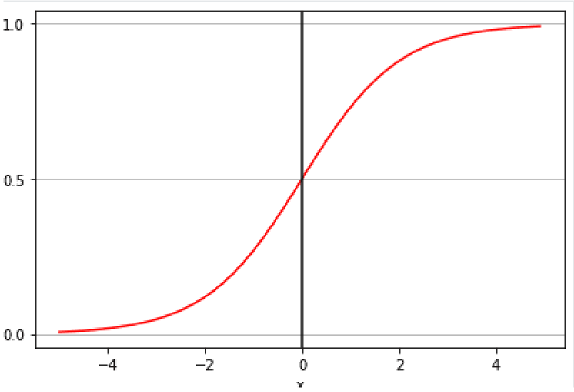


Fig.2. Logistic Regression model graph

3.4. *The Naive Bayes method*

The Naive Bayes algorithm aims to detect the new category of the class given to the system through a classification calculation determined according to probability calculations. The Naive Bayes method is a classification method that adapts to estimate the relationship between the target label to be achieved and the input parameters applied in the problem. This method uses these probabilities for prediction by calculating the frequency of the combination of independent parameters and dependent variables.

$$P(A|B) = \frac{P(B|A)\,P(A)}{P(B)} \qquad (3)$$

Formula for Bayesian statistical is calculated as above and here: P(A|B) is posterior, the above of the equation is equal to prior x likelihood and P(B) is evidence.

Naive Bayes method is used in the model in the study and 70% of the data is used for training in the model. An attempt is made to predict which class the data will be in by using the probability calculations made with the data in the training set and the 30% of the test data given to the system allocated for prediction.

4. Experimental Implementation

In this project, the Knime platform os used for the simulation of both prevent models (https://www.knime.com/). The interface of Knime is displayed in the figure below. Knime is an easy, user-friendly, and open-source platform where the parts needed can be dragged and dropped for modeling into the workspace, the main interface of the model can be created and rendered visually.

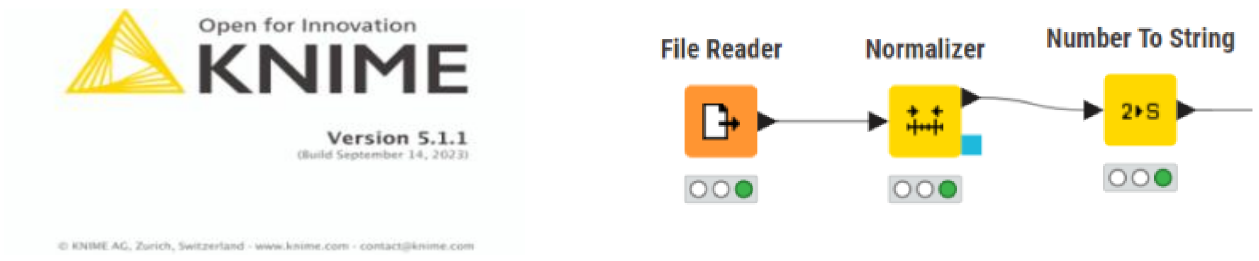
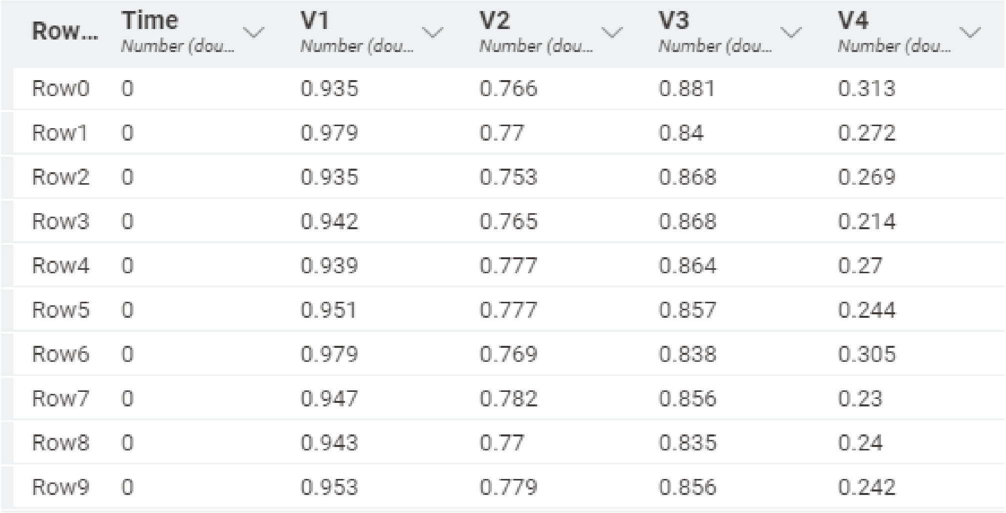

Open for Innovation
KNIME
Version 5.1.1
(Build September 14, 2023)
© KNIME AG, Zurich, Switzerland · www.knime.com · contact@knime.com
File Reader
Normalizer
Number To String
2▸S

Fig.3. Knime software visual

The data are first digitized to be used in the linear regression method and Naive Bayes method modeled for the problem. Afterward, all data are normalized to obtain a more efficient running time for both models, as shown in Fig.4.



Row...	Time Number (dou...	V1 Number (dou...	V2 Number (dou...	V3 Number (dou...	V4 Number (dou...
Row0	0	0.935	0.766	0.881	0.313
Row1	0	0.979	0.77	0.84	0.272
Row2	0	0.935	0.753	0.868	0.269
Row3	0	0.942	0.765	0.868	0.214
Row4	0	0.939	0.777	0.864	0.27
Row5	0	0.951	0.777	0.857	0.244
Row6	0	0.979	0.769	0.838	0.305
Row7	0	0.947	0.782	0.856	0.23
Row8	0	0.943	0.77	0.835	0.24
Row9	0	0.953	0.779	0.856	0.242

Fig.4. A part of normalizer dataset

In Fig.5 we see the modeling of the Logistic Regression technique. Here, 70 % of the data is used, while the remaining 30 % is reserved for testing. After the training process is completed, system reliability is tested by applying it to the test set using the determined parameters.

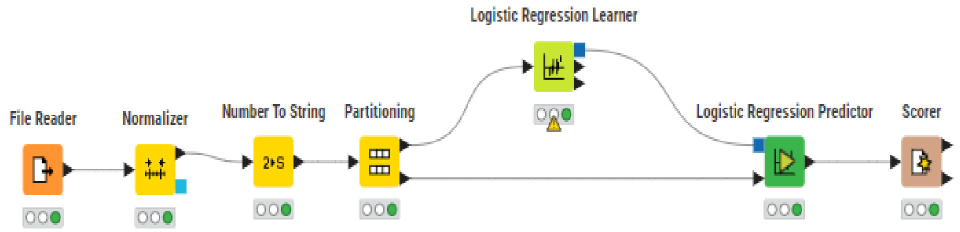


Fig.5. Logistic Regression prediction model

For the Naive Bayes model shown in Fig.6, 30 percent of the data set is utilized for testing, while seventy percent are employed for exercising. For Naive Bayes learning, the default probability is 0.0001 and the minimum standard deviation is 0.0001. Then, the Naive Bayes model is tested on the test set.

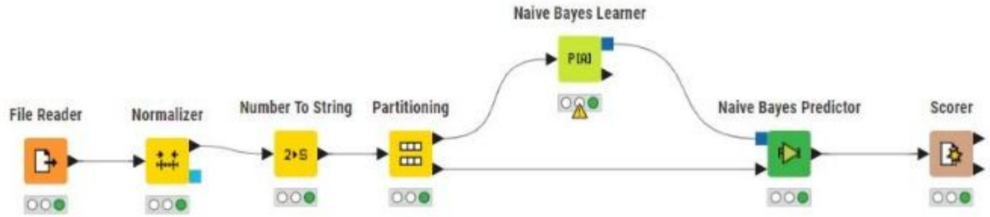
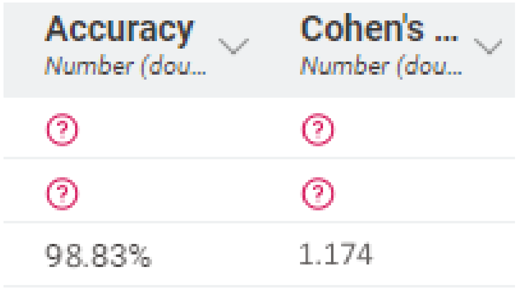


Fig.6. Naive Bayes prediction model

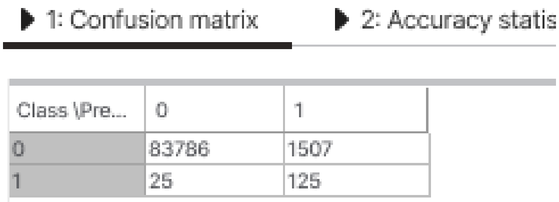
5. Results

In Logistic Regression modeling, the accuracy rate is 98.83% and the error rate is 1.174%. And this result is shown in detail in Fig.7.



Accuracy Number (dou...	Cohen's ... Number (dou...
?	?
?	?
98.83%	1.174

Fig.7. The Logistic Regression technique: Accuracy and Cohen's kappa result



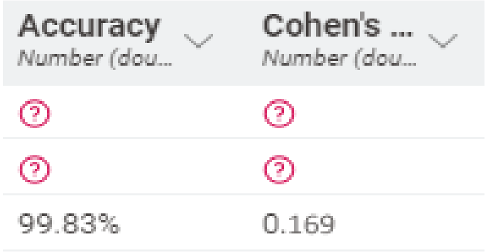
▶ 1: Confusion matrix ▶ 2: Accuracy statis

Class \Pre...	0	1
0	83786	1507
1	25	125

Fig.8. Logistic Regression Confusion matrix result

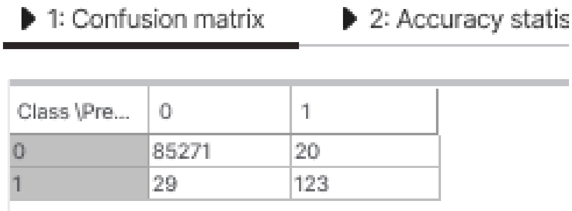
As seen in Fig.8 above, in the Logistic regression method model, the quantity of true positives is 125 and the amount of false positives is 25. While the amount of true negatives is 83786, the amount of errors is 1507 in storage and connection tubes.

In Fig.9, the Naive Bayes modeling, the accuracy rate is 99.83% and the error rate is 0.169% in this model, the quantity of true positives is 123 and the amount of false positives is 29. However, there are 85271 real negatives, and twenty (20) false negatives are reported in Fig.10.



Accuracy Number (dou...	Cohen's ... Number (dou...
?	?
?	?
99.83%	0.169

Fig.9. The Naive Bayes Accuracy and Cohen's kappa result



▶ 1: Confusion matrix ▶ 2: Accuracy statis

Class \Pre...	0	1
0	85271	20
1	29	123

Fig.10. The Naive Bayes Confusion matrix result

As a result, an analysis ois conducted on the Knime platform using the fraudulent circumstances on credit card data set obtained from the Kaggle database. We analyze two machine learning methods, Logistic regression and Naive Bayes algorithms are used in the analysis. The overall statistics of the two approaches are revealed in Table 1.

Table 1. Results of proposed solution. The Logistic regression and Naive bayes algorithms are trained on the train data set. Cohen's kappa value, that is, 1.174% and 0.169% of customers with fraud probability are detected correctly. These are positioned as general error rates. The overall accuracy is almost the same in both models (99.83% vs 98.83%).

Prediction models	Accuracy	Error
Logistic Regression	98.83 %	1.174%
Naive Bayes	99.83 %	0.169 %

The accuracy and error values of both models are shown the Table 1. In light of the results obtained, although the two algorithms produced very good results, it is determined that the Logistic regression algorithm is less successful than the Naive Bayes algorithm for this study.

6. Conclusion

In conclusion, we cannot claim that our algorithm entirely identifies fraud even though there are other fraud detection methods. We conclude from the results of our evaluation that the precision of both Naive Bayes and Logistic Regression was roughly comparable. When it comes to accuracy, recall, F1, and error scores, the Naive Bayes approach performed better than the Logistic regression algorithm. Consequently, we deduced that the Naive Bayes method outperformed the Logistic Regression approach in detecting credit card fraud.

The data above made it evident that various machine learning algorithms were utilized to recognizing fraud, however, the outcomes were not good enough. Therefore, by using machine learning algorithms to precisely identify credit card fraud, subsequent research may provide more accurate findings.

References

Awoyemi, John O. et al. (2017). "Credit Card Fraud Detection Using Machine Learning Techniques: A Comparative Analysis." 2017 International Conference on Computing Networking and Informatics (ICCNI) doi:10.1109/iccni.2017.8123782

Behrouz Far et al. (2018). "Supervised Machine Learning Algorithms for Credit Card Fraudulent Transaction Detection: A Comparative Study." Annals of the History of Computing, doi.ieeecomputersociety.org/10.1109/IRI.2018.00025

Jiang, Changjun et al. (2018). "Credit Card Fraud Detection: A Novel Approach Using Aggregation Strategy and Feedback Mechanism." IEEE Internet of Things Journal 5: 3637-3647

Melo-Acosta, German E., et al. "Fraud Detection in Big Data Using Supervised and Semi-Supervised Learning Techniques." 2017 IEEE Colombian Conference on Communications and Computing (COLCOM) doi:10.1109/colcomcon.2017.8088206

Navanushu Khare et al. (2020). " An Efficient Study of Fraud Detection System Using Ml Techniques." Intelligent Computing and Innovation on Data Science, doi: 10.1007/978-981-15-3284-9_8

Pumsirirat et al. (2018). "Credit Card Fraud Detection using Deep Learning based on Auto-Encoder and Restricted Boltzmann Machine." International Journal of Advanced Computer Science and Applications, 9(1)

Rimpal R. Popat et al. (2018). "A Survey on Credit Card Fraud Detection Using Machine Learning." 2018 2nd International Conference on Trends in Electronics and Informatics (ICOEI), doi: 10.1109/ICOEI.2018.8553963

Randhawa, Kuldeep et al. (2018). "Credit Card Fraud Detection Using AdaBoost and Majority Voting." IEEE Access, vol. 6, 2018, pp. 14277–14284 doi:10.1109/access.2018.2806420

Roy, Abhimanyu, et al. (2018). "Deep Learning Detecting Fraud in Credit Card Transactions." 2018 Systems and Information Engineering Design Symposium (SIEDS) doi:10.1109/sieds.2018.8374722.

Tince Etlin Tallo et al. (2018). "The Implementation of Genetic Algorithm in Smote (Synthetic Minority Oversampling Technique) for Handling Imbalanced Dataset Problem." 2018 4th International Conference on Science and Technology (ICST), doi: 10.1109/ICSTC.2018.8528591

Xuan Shiyang, et al. (2018). "Random Forest for Credit Card Fraud Detection." 2018 IEEE 15th International Conference on Networking, Sensing and Control (ICNSC) doi:10.1109/icnsc.2018.8361343

Vaishnavi Nath Dornadula et al. (2019). "Credit Card Fraud Detection using Machine Learning Algorithms." International conference on recent trends in advanced computing 2019, doi.org/10.1016/j.procs.2020.01.057