IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

CREDIT CARD FRAUD DETECTION USING MACHINE LEARNING

[1]Harsha Vardhan, [2]G. Venkatesh, [3]A. Ram Charan, [4]CH. Navya, [5]A. Venkateswarllu

[1]AsstProfessor, CSE, JNTUK, GVR&S CET, Guntur, 522017, Andhra Pradesh, India,
[2,3,4,5] Research Scholars, JNTUK, GVR&S CET, Guntur, 522017, Andhra Pradesh, India

ABSTARCT

The goal of this project is to detect credit card theft in actual situations. The amount of credit card frauds that occur now days is far higher than it was in the past. Phony identities and other technologies are being used by criminals to trick people and extract money from them. Findi0ng a remedy for these kinds of frauds is therefore crucial. We created a model for this proposed project to identify fraudulent activities in credit card transactions. The majority of the crucial elements needed to identify unlawful and criminal transactions are available in this system. The ever-evolving nature of technology makes it more challenging to monitor the activities and trends of illicit transactions. One can use the growing array of technology available to them to come up with a solution.

I. INTRODUCTION

The use of credit cards has skyrocketed in the modern world, as more and more people use cashless payment methods and rely solely on internet commerce. The credit card has improved accessibility and ease of use for digital transactions. Fraud detection is the practice of keeping an eye on a cardholder's transaction activity to determine if an incoming transaction is legitimate and allowed or not—if it isn't, it will be considered unlawful. We are using the logistic regression algorithm, naïve bayes algorithm, and random forest algorithm in a planned system to categorize the credit card dataset. An associate in the algorithmic nursing program for regression and classification is called Random Forest.

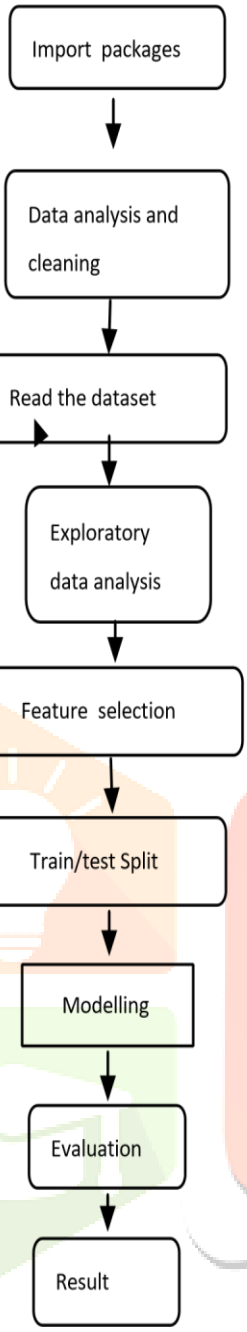
II. PROBLEM STATEMENT

In case of the existing system the fraud is detected after the fraud is done that is, the fraud is detected after the complaint of the holder. And so the card holder faced a lot of trouble before the investigation finish. And also as all the transaction is maintained in a log, we need to maintain a huge data, and also now a day's a lot of online purchase are made so we don't know the person how is using the card online, we just capture the ip address for verification purpose. So there need a help from the cyber crime to investigate the fraud. To avoid the entire above dis advantage we propose the system to detect the fraud in a best easy way.

III. LITERATURE SURVEY

1.authour by K .Ratna Sree Valli, P. Jyothi , G .Varun Sai ,R .Rohith Sai Subash, we have explained the concept of frauds related to credit cards. 2. Author by Lakshmi svs This paper investigates the performance of logistic. regression, decision tree and random forest for credit cvard fraud detection. 3. Author by Sushant agarwal It is possible to the behaviour of such fraudulent activities to minimise them and Avoid recurrence.4. Author by mr. Thirunavukkarasu it involves leveraging algorithms to identify fraudulent transactions based on patterens an anomalis in the data. 5.Author by O.John It aims to evaluate and compare various methods and detective fraudulent transactions.6. Author by Deep Prajapati it creats a model to achieve the breast possible outcomes in detective and impending fraudulent transaction..7.Author by prapyusharma monitoring the behaviour of transactions done by the card holder to detectively unauthorized for transactions.

IV. PROPOSED MODEL



AbbreviationsandAcronyms(Heading2)

Defineabbreviationsandacronymsthefirsttimetheyareusedinthetext,evenaftertheyhavebeendefinedintheabstract.AbbreviationssuchasIEEEandSIdonothavetobedefined.Donotuseabbreviationsinthetitleorheadsunlesstheyareunavoidable.

3.1methadology

Machine Learning is undeniably one of the most influential and powerful technologies in today‘s world. Machine learning is a tool for turning information into knowledge. In the past 50 years, there has been an explosion of data [10]. This mass of data is useless; we analyse it and find the patterns hidden within. Machine learning techniques are used to automatically find the valuable underlying patterns within complex data that we would otherwise struggle to discover. The hidden patterns and knowledge about a problem can be used to predict future events and perform all kinds of complex decision making. To learn the rules governing a phenomenon, machines have to go through a learning process**,** trying different rules and learning from how well they perform. Hence, why it‘s known as Machine Learning*“Traditionally, software engineering combined human created rules with data to create answers to a problem. Instead, machine learning uses data and answers to discover the rules behind a problem.”* - Chollet, 2017.

4.1.1 Basic Terminology

- Dataset: A set of data examples, which contain features important to solving the problem.
- Features: Important pieces of data that help us understand a problem. These are fed into a Machine Learning algorithm to help it learn.
- Model: The representation (internal model) of a phenomenon that a Machine Learning algorithm has learnt. It learns this from the data it is shown during training. The model is the output you get after training an algorithm. For example, a decision tree algorithm would be trained and produce a decision tree model.

4.1.2 Types of Machine Learning

There are multiple forms of Machine Learning; supervised, unsupervised, semi- supervised and reinforcement learning. Each form of Machine Learning has differing approaches, but they all follow the same underlying process and theory.

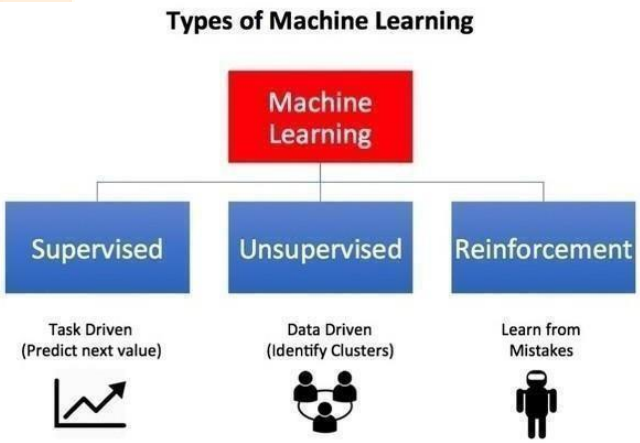


Fig 1.1 Types of Machine Learning

- **Supervised Learning:** It is the most popular paradigm for machine learning. Given data in the form of examples with labels, we can feed a learning algorithm these examplelabel pairs one by one, allowing the algorithm to predict the label for each example, and giving it feedback as to whether it predicted the right answer or not. Over time, the algorithm will learn to approximate the exact nature of the relationship between examples and their labels. When fullytrained, the supervised learning algorithm will be able to observe a new, never- before-seen example and predict a good label for it.

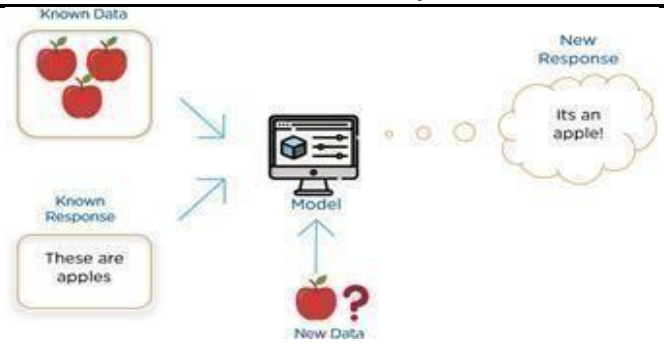


Fig 1.2 Supervised Learning

- **Unsupervised learning**:

  It is very much the opposite of supervised learning. It features no labels. Instead, the algorithm would be fed a lot of data and given the tools to understand the properties of the data. From there, it can learn to group, cluster, and organize the data in a way such that a human can come in and make sense of the newly organized data. Because unsupervised learning is based upon the data and its properties, we can say that unsupervised learning is data- driven. The outcomes from an unsupervised learning task are controlled by the data and the way it‘s formatted.

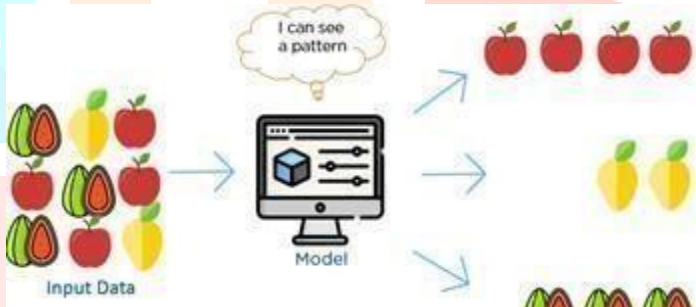


Fig 1.3 Unsupervised Learning

- **Reinforcement learning**: It is fairly different when compared to supervised and unsupervised learning. Reinforcement learning is very behaviour driven. It has influences from the fields of neuroscience and psychology. For any reinforcement learning problem, we need an agent and an environment as well as a way to connect the two through a feedback loop. To connect the agent to the environment, we give it a set of actions that it can take that affect the environment. To connect the environment to the agent, we have it continually issue two signals to the agent: an updated state and a reward (our reinforcement signal for behaviour).

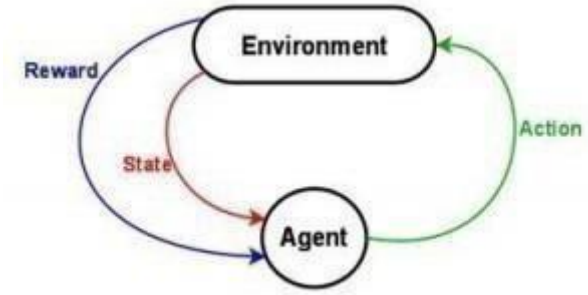


Fig 1.4 Reinforcement Learning

4.1.3 Basic Process

i. Data Collection**:** Collect the data that the algorithm will learn from.

ii. Data Preparation**:** Format and engineer the data into the optimal format, extracting important features and performing dimensionality reduction.

iii. Training**:** Also known as the fitting stage, this is where the Machine Learning algorithm actually learns by showing it the data that has been collected and prepared.

iv. Evaluation**:** Test the model to see how well it performs.

v. Tuning**:** Fine tune the model to maximize its performance.

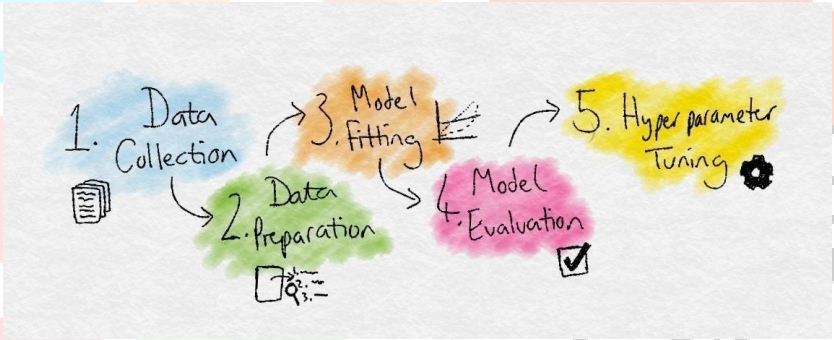


Fig 1.5 General Process

1.5.4 Algorithms Used

Machine Learning offers a wide range of algorithms to choose from. These are usually divided into classification, regression, clustering and association. Classification and regression algorithms come under supervised learning while clustering and association comes under unsupervised learning.

- Classification: A classification problem is when the output variable is a category, such as ―red‖ or ―blue‖ or ―disease‖ and ―no disease‖. Example: Decision Trees
- Regression: A regression problem is when the output variable is a real value, such as

  ―dollars‖ or ―weight‖. Example: Linear Regression

A few algorithms can come under multiple types. Considering the problem statement and the desired output of the project, the most suitable type of algorithm would come under regression. Before choosing an algorithm and working with it further, many algorithms were explored and the error rates and accuracy were checked for each. The table 1.1 summarizes the various algorithms that were explored.

Table 1.1 Summary of the Approaches

S.no.	Algorithm	Accuracy
1.	Random Forest	100%
2.	Naïve Bayes	100%
3.	Logistic Regression	100%
4.	KNN	100%
5.	Support vector machine (SVM)	100%
6.	Decision tree	100%

From the above table, we can conclude that the Logistic Regression Algorithm gives the best accuracy for our dataset.

Logistic Regression:

Logistic Regression is one of the classification algorithm, used to predict a binary values in a given set of independent variables (1 / 0, Yes / No, True / False). To represent binary / categorical values, dummy variables are used. For the purpose of special case in the logistic regression is a linear regression, when the resulting variable is categorical then the log of odds are used for dependent variable and also it predicts the probability of occurrence of an event by fitting data to a logistic function. Such as

$$O = e^{(I0 + I1*x)} / (1 + e^{(I0 + I1*x)})$$

Where, O is the predicted output I0 is the bias or intercept term I1 is the coefficient for the single input value (x). Each column in the input data has an associated I coefficient (a constant real value) that must be learned from the training data.

$$y = e^{(b0 + b1*x)} / (1 + e^{(b0 + b1*x)})$$

regression is started with the simple linear regression equation in which dependent

variable can be enclosed in a link function i.e.,to start with logistic regression, I'll first write the simple linear regression equation with dependent variable enclosed in a link function:

$$A(O) = \beta 0 + \beta(x)$$

Where A() : link function O : outcome variable x : dependent variable A function is established using two things:

1) Probability of Success(pr) and 2) Probability of Failure(1-pr). pr should meet following criteria:

a) probability must always be positive (since p >= 0) b) probability must always be less than equals to 1 (since pr <= 1). By applying exponential in the first criteria and the value is always greater than equals to 1. $pr = \exp(\beta o + \beta(x)) = e^{(\beta o + \beta(x))}$

b) For the second criteria, same exponential is divided by adding 1 to it so that the value will be less than equals to 1 $pr = e^{(\beta o + \beta(x))} / e^{(\beta o + \beta(x))} + 1$

Logistic function is used in the logistic regression in which cost function quantifies the error, as it models response is compared with the true value.

$X(\theta)=-1/m*(\sum yi\log(h\theta(xi))+(1-yi)\log(1-h\theta(xi)))$ Where $h\theta(xi)$: logistic function yi :

outcome variable Gradient descent is a learning algorithm.

Decision Tree Algorithm:

Decision tree is a type of supervised learning algorithm (having a pre-defined target variable) that is mostly used in classification problems. It works for both categorical and continuous input and output variables. In this technique, we split the population or sample into two or more homogeneous sets (or sub-populations) based on most significant splitter / differentiator in input variables.

TYPES OF DECISION TREE

1. Categorical Variable Decision Tree: Decision Tree which has categorical target variable then it called as categorical variable decision tree.
2. Continuous Variable Decision Tree: Decision Tree has continuous target variable then it is called as Continuous Variable Decision Tree

TERMINOLOGY OF DECISION TREE

Root Node: It represents entire population or sample and this further gets divided into two or more homogeneous sets.

1. Splitting: It is a process of dividing a node into two or more sub-nodes.
2. Decision Node: When a sub-node splits into further subnodes, then it is called decision node.
3. Leaf/ Terminal Node: Nodes do not split is called Leaf or Terminal node.
4. Pruning: When we remove sub-nodes of a decision node, this process is called pruning. You can say opposite process of splitting.
5. Branch / Sub-Tree: A sub section of entire tree is called branch or sub-tree.
6. Parent and Child Node: A node, which is divided into sub-nodes is called parent node of sub-nodes where as sub-nodes are the child of parent node.

WORKING OF DECISION TREE:

Decision trees use multiple algorithms to decide to split a node in two or more sub- nodes. The creation of sub-nodes increases the homogeneity of resultant sub-nodes. In other words, we can say that purity of the node increases with respect to the target variable. Decision tree splits the nodes on all available variablesand then selects the split which results in most homogeneous sub-nodes.

1. Gini Index
2. Information Gain
3. Chi Square

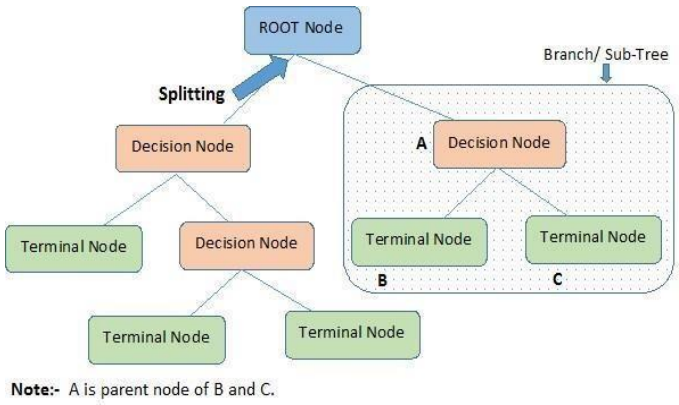


Fig 1.15 Work of Decision Tree

Random Forest:

Random forest is a tree based algorithm which involves building several trees and combining with the output to improve generalization ability of the model. This method of combining trees is known as an ensemble method. Ensembling is nothing but a combination of weak learners (individual trees) to produce a strong learner. Random Forest can be used to solve regression and classification problems. In regression problems, the dependent variable is continuous. In classification problems, the dependent variable is categorical.

WORKING OF RANDOM FOREST:

Bagging Algorithm is used to create random samples. Data set D1 is given for n rows and m columns and new data set D2 is created for sampling n cases at random with replacement from the original data. From dataset D1,1/3rd of rows are left out and is known as Out of Bag samples. Then, new dataset D2 is trained to this models and Out of Bag samples is used to determine unbiased estimate of the error. Out of m columns, M << m columns are selected at each node in the data set. The M columns are selected at random. Usually, the default choice of M, is m/3 for regression tree and M is sqrt(m) for classification tree. Unlike a tree, no pruning takes place in random forest i.e, each tree is grown fully. In decision trees, pruning is a method to avoid over fitting. Pruning means selecting a sub tree that leads to the lowest test error rate. Cross validation is used to determine the test error rate of a sub tree. Several trees are grown and the final prediction is obtained by averaging or voting.

Algorithm steps for finding the Best algorithm:

Step 1: Import the dataset
Step 2: Convert the data into data frames format
Step3: Do random oversampling using ROSE package

Step4: Decide the amount of data for training data and testing data
Step5: Give 70% data for training and remaining data for testing.
Step6: Assign train dataset to the mode
Step7: Choose the algorithm among 3 different algorithms and create the model
Step8: Make predictions for test dataset for each algorithm

Step9: Calculate accuracy for each algorithm

Step10: Apply confusion matrix for each variable

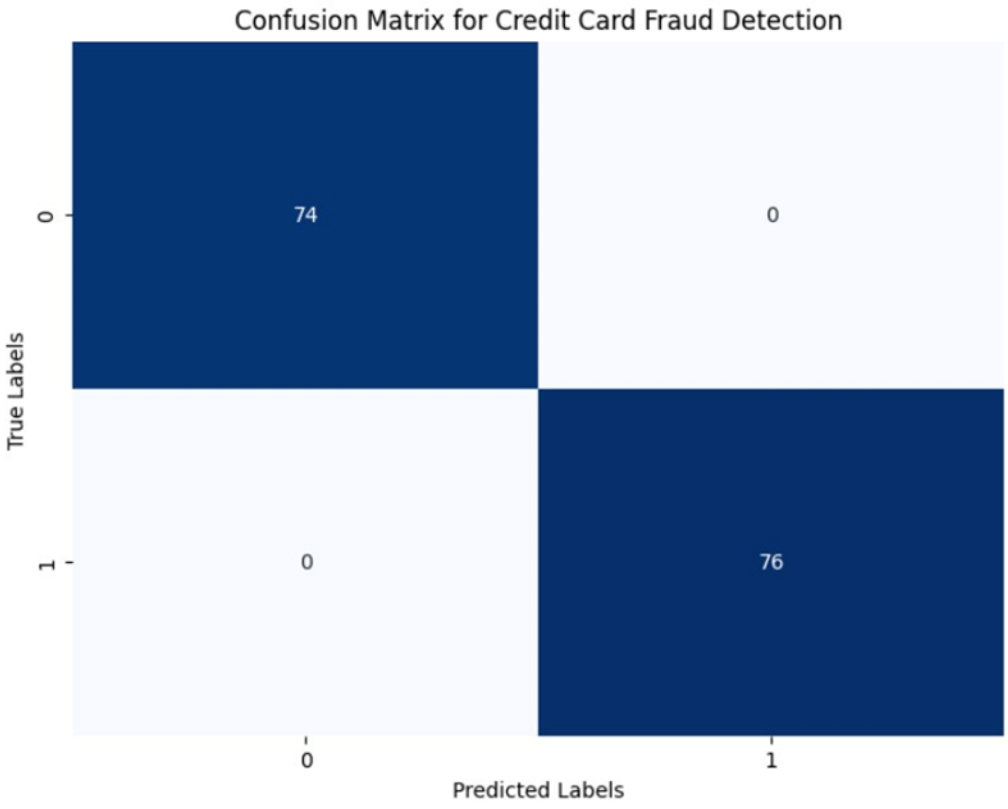
Step11: Compare the algorithms for all the variables and find out the b

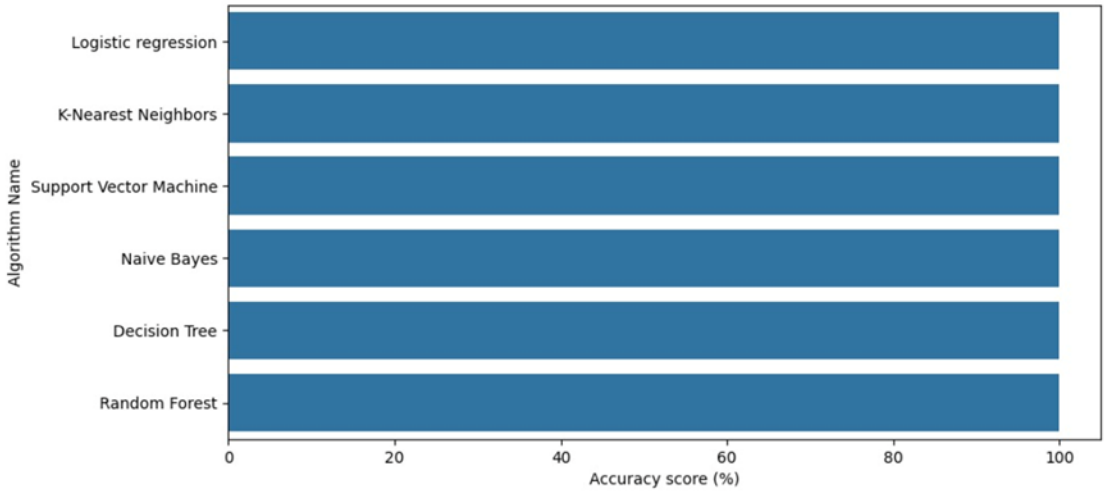
Result

```
[1]: import pandas as pd

[2]: dataset = pd.read_csv('credit.xls', index_col = 0)
     dataset = dataset.drop(columns=['category', 'state'])

[3]: dataset.head()
```

[3]:

trans_hour	trans_day	trans_month	trans_year	upi_number	age	trans_amount	zip	fraud_risk
22	16	9	2021	87	27	199.31	84946	1
12	13	7	2016	71990978	42	448.06	68524	1
13	10	8	2019	5601	70	127.79	34800	1
10	7	10	2023	671623009	61	967.70	7311	0
17	10	1	2022	666	40	911.30	60970	1





[54]:
```
# Now, let's use the trained model to predict the fraud risk of a new transaction
# Define the new transaction data
new_transaction = pd.DataFrame([[22, 16, 9, 2021, 'Retail', 87, 27, 199.31, 'Missouri', 84946]],
                               columns=['trans_hour', 'trans_day', 'trans_month', 'trans_year', 'category', 'upi_number', 'age', 'trans_amount', 'state'

# Make prediction for the new transaction
prediction = clf.predict(new_transaction)

# Interpret the prediction
if prediction[0] == 1:
    print("The Transaction is at High Risk of Fraud.")
else:
    print("The Transaction is Not at High risk of Fraud.")
```

```
The Transaction is at High Risk of Fraud.
```

[55]:
```
# Now, let's use the trained model to predict the fraud risk of a new transaction
# Define the new transaction data
new_transaction = pd.DataFrame([[10, 7, 10, 2023, 'Entertainment', 671623009, 61, 967.7, 'Tennessee', 7311]],
                               columns=['trans_hour', 'trans_day', 'trans_month', 'trans_year', 'category', 'upi_number', 'age', 'trans_amount', 'state'

# Make prediction for the new transaction
prediction = clf.predict(new_transaction)

# Interpret the prediction
if prediction[0] == 1:
    print("The Transaction is at High Risk of Fraud.")
else:
    print("The Transaction is Not at High risk of Fraud.")
```

```
The Transaction is Not at High risk of Fraud.
```

[46]:

	Algorithm Name	Accuracy score (%)
0	Logistic regression	100.0
1	K-Nearest Neighbors	100.0
2	Support Vector Machine	100.0
3	Naive Bayes	100.0
4	Decision Tree	100.0
5	Random Forest	100.0

Conclusion

In this study we use machine learning algorithms which play a crucial role in credit card fraud detection by leveraging patterns and anomalies within transaction data to identify potentially fraudulent activities. By employing techniques such as supervised learning, unsupervised learning, and anomaly detection, machine learning models can effectively distinguish between genuine and fraudulent transactions.In this we use only Supervised learning algorithms like logistic regression, decision trees, random forests,naïve bayes,kNN,support vector machine classifiers can be trained on labeled datasets to predict the likelihood of fraud for new transactions.

In order to evaluate performance of each model we have used metrics like Accuracy, Confusion Matrix, and Classification Report. Hence, we have acquired the result of an accurate value of credit card fraud detection i.e. 100% using all algorithms with new enhancements. Usage of more preprocessing techniques would also assist.

References

1.Bhattacharyya, S., Jha, S., Tharakunnel, K., & Westland, J. C. (2011). A survey of credit card fraud detection techniques: Data and technique-oriented perspective. Decision Support Systems, 50(3), 559-567.
2.Dal Pozzolo, A., Boracchi, G., Caelen, O., Alippi, C., & Bontempi, G. (2015). Credit card fraud detection: A realistic modeling and a novel learning strategy. IEEE Transactions on Neural Networks and Learning Systems, 29(8), 3784-3797.
3.Bhattacharyya, S., Jha, S., & Kalita, H. (2019). A survey on credit card fraud detection. Journal of Network and Computer Applications, 135, 62-82.
4.Phua, C., Lee, V., Smith, K., & Gayler, R. (2005). A comprehensive survey of data mining-based fraud detection research. arXiv preprint cs/0512041.
5.Zhang, Y., Li, Z., & Chen, X. (2019). An overview of credit card fraud detection methods. In International Conference on Security and Privacy in Communication Systems (pp. 467-478). Springer, Cham.
6.Coussement, K., & Van den Poel, D. (2008). Churn prediction in subscription services: An application of support vector machines while comparing two parameter-selection techniques. Expert Systems with Applications, 34(1), 313-327.
7.Rosales, J. E. B., & Finkelstein, L. (2014). Credit card fraud detection using autoencoders in high dimensional imbalanced datasets. In International Workshop on Machine Learning and Data Mining in Pattern Recognition (pp. 307-321). Springer, Cham.