**Table of Contents**

|  |  |
| --- | --- |
| Content | Page no |
| Abstract | 1 |
| Introduction   1. Background 2. Objective | 1 |
| Dataset   1. Description of given dataset 2. Data Preprocessing | 2 |
| Implementation   1. KNN Algorithm Overview 2. Working Principle 3. Workflow of KNN 4. Advantages and Disadvantages 5. Isolation Forest Overview 6. Working Principle 7. Isolation Trees 8. Advantages and Disadvantages 9. Comparison between KNN vs Isolation Forest | 2-6 |
| Step-by-Step Implementation   1. Import required libraries 2. Load and explore dataset 3. Preprocess the Data 4. Split the Data 5. Feature scaling 6. K-Nearest Neighbor for Anomaly Detection 7. Isolation Forest for Anomaly Detection | 6-10 |
| Evaluation of K-Nearest Neighbors   1. Classification Report 2. Confusion Matrix: KNN Model | 10-12 |
| Evaluation of Isolation Forest   1. Classification Report 2. Confusion Matrix: Isolation Forest Model | 12-13 |
| Results & Discussions   1. KNN vs Isolation Forest 2. Interpretation | 14 |
| Real Time Applications | 14-15 |
| Conclusion   1. Summary 2. Future work | 15-16 |
| References | 16 |

**Anomaly Detection in Network Traffic Using K-Nearest Neighbor and Isolation Forest Algorithms**

1. **Abstract**

As per the given problem statement in the assignment, we are likely to study, implement and analyze anomalies of the given dataset using K-Nearest Neighbors and Isolation Forest approaches (*Liu, F. T., Ting, K. M., & Zhou, Z. H, 2008*), p. 1). As we know that Anomaly Detection is a very important technique, which is used in various industries or technological aspects such as cybersecurity, fraud detection, and system monitoring which mainly deals with identifying the susceptive patterns that different from the expected patterns. To implement the Anomaly detection technique, we are going to use NETWORK ANOMALY DATA dataset, an optimal dataset for network intrusion detection, which is taken for the training, evaluation and testing the machine learning models what we have implemented. In our report, we are implementing the certain steps like data preprocessing to handle the unwanted data or missing data, encoding categorical data, and apply scaling feature. By using KNN and Isolation Forest we are trying to generate the results from the models in using certain metrics such as precision, recall. F1-score, and Confusion Matrix. This assignment contributes to the study and development of detection algorithms that detect anomalies and that can enhance the network authentication or security by early identifying and detecting the potential threats.

1. **Introduction**
2. **Background**

In this technological world, data security plays a very important role because currently world is facing potential cyber scams, potential threats to the organization systems, financial scams related to banking etc. (*Qiu, T., et al, 2020, p. 1*) To put a stop to these suspicious activities we have Anomaly Detection Algorithm, that plays a very important role to identify and detect the data patterns by suspecting any potential threats to the data systems. This detection technique is used in lot of domains such as system monitoring, network security, cybersecurity, and fraud detection activities etc. To implement this detection approach, we are using NETWORK ANOMALY DATA dataset to train, test and evaluate the ML model accuracy.

1. **Objective**

The main objective of this algorithm we are going to develop a detection system that can accurately identify the anomalies in the data related to the network traffic. To achieve this, we are going to use two machine learning algorithms: K-Nearest Neighbor (KNN) and Isolation Forest. To develop the mentioned algorithms, we have to follow certain steps:

1. Data Preprocessing: First step, prepare the NETWORK ANOMALY DATA dataset for model training by cleaning the dataset such as null or missing values, encoding categorial variables, scaling features.
2. Model Implementation: Here we are implementing KNN and Isolation Forest algorithms.
3. Model Evaluation: In this step, we are evaluating performance of both models using some performance or analysis metrics such as f1-score, precision, recall, and confusion matrix.
4. Visualization: The results of model’s evaluation are plotted as graphs or confusion matrices to get clear insights visually.
5. Comparison: Here we are comparing the both algorithms effectiveness and accuracy.

1. **Dataset**
2. **Description of Given Dataset**

In this assignment, we are using NETWORK ANOMALY DATA dataset which is very similar to KDD Cup 1999 dataset, which is one of the popular datasets for (*Chandola, V., Banerjee, A., & Kumar, V., 2009, p. 6*) network intrusion detection. This dataset was actually developed by This dataset is suitable for both normal and suspicious network traffic data, which makes suitable for training and evaluating anomaly detection models.

As we found, this dataset contains Approx. “125975” data records, each with 41 features which contains both normal and suspicious data. The suspicious activities fall under 4 types: Denial of service (DoS), Remote to Local(R2L), User to Root(U2R), and Probing.

1. **Data Preprocessing**

We already know that the data preprocessing plays a very important role for preparing the dataset to train and evaluate our implemented machine learning model or algorithm. The key steps involving this data preprocessing for Network anomaly data dataset include:

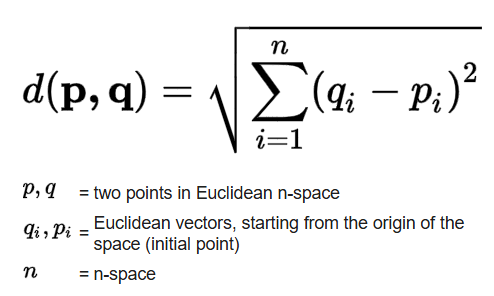
1. Handling Missing Values: Here, NETWORK\_ANOMALY\_DATA dataset does not contain any missing or null values, so this step is not actually required. But we have to follow the rules to defined to develop the machine learning models. We have some common approaches for handling missing data or values include removing unwanted records or rows from the dataset, inputting missing values using statistical methods such as mean or mode or any other we can use.
2. Encoding Categorical Variables: Here in this dataset contains several categorical features such as protocol\_type, service, and flag. As we know that machine learning algorithms require numerical data to train so we need to convert to numerical values. We have one common method to do that i.e. One-Hot-Encoding technique, which creates binary columns for each category.
3. Feature Scaling: It is the normalization technique to normalize the range of features to contribute to the model. Here Standardization is the common scaling technique that transforms the features with mean value 0 and standard deviation value 1.
4. **Implementation**
5. **KNN Algorithm Overview**

K-Nearest Neighbor (KNN) is a algorithm mainly it comes under the (*Chandola, V., Banerjee, A., & Kumar, V., 2009, p. 25)* supervised machine learning algorithm majorly used for the to solve the problems of classification as well as regression problems, which stores all the available data and finds the new data points on similarity.

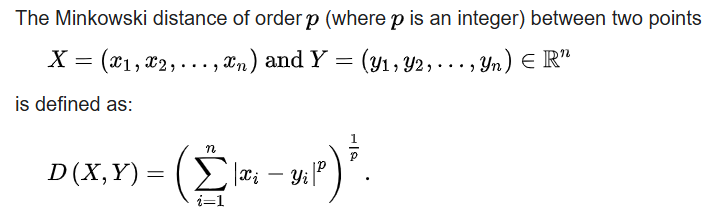
1. **Working Principle**

The steps involved in implementing the KNN Algorithm as follows:

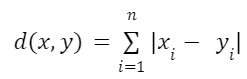
1. Choose the number of neighbors(k): As the algorithm rule the value of – k determines the how many neighbors will be considered to classify the data points.
2. Calculate the distance: To calculate the distance we have some common distance measure techniques such as Euclidean, Manhattan etc. for given data point we can calculate the accurate distance to all other points.
3. Euclidean Distance: It calculates the distance between the two points in the plane i.e. 2D plane or hyperplane. This distance helps as the metric to find the net displacement between two points.



1. Minkowski Distance: It is a distance calculation metric that is used to find the similarity and mis similarity between the two or more points by computing the sum of the absolute differences between them.

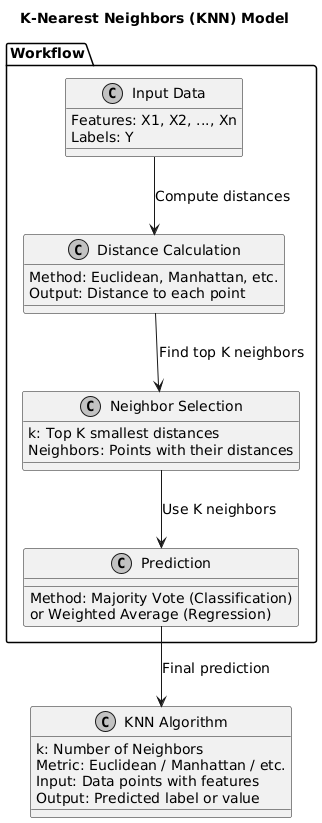


1. Manhattan Distance: This also used to calculate or determine the distance between the two points in the 2D space i.e. this is a metric that finds the absolute differences of their cartesian coordinates.



1. Identify the K-th nearest neighbor: Here we can identify the data points which are very close to the given data point to find most accurate result.
2. Classify the data points: Here in this step we can classify data points based on labels with majority of the data points among k-th nearest neighbors.
3. **Workflow of KNN Algorithm**

Here we are representing the workflow of the K-Nearest Neighbors (KNN) to show the flow of operations, such as calculating the distances, finding the nearest neighbors, and finding the prediction result.



1. **Advantage and Disadvantage**
2. **Advantages:**

* Kth-Nearest-Neighbor algorithm is very easy to understand and implement.
* One important thing we have to note is the algorithm k-nearest neighbor is a type of algorithm which is actually a lazy learner, that defines that the KNN is actually performs the training phase only it is required. It stores all the preprocessed dataset and performs classification during prediction state.
* And, KNN is used for both classification and regression models.

1. **Disadvantages:**

* This algorithm is computationally expensive because it needs to calculate the distances to all the data points which makes the algorithm a bit time taking process.
* It needs the more memory because it stores the entire dataset at a time.
* KNN may perform not with high accuracy only if the KNN features contains the redundant or irrelevant features.

1. **Isolation Forest Overview**

Isolation forest is a type of machine learning model that is mainly a type of unsupervised machine learning approach, which mainly deals with detection of anomalies in the given data that identifies the anomalies by isolating outliers in the given data. This Isolation Forest algorithm approach is supported by Decision Tree technique, with shortest path in the tree structure.

1. **Working Principle**

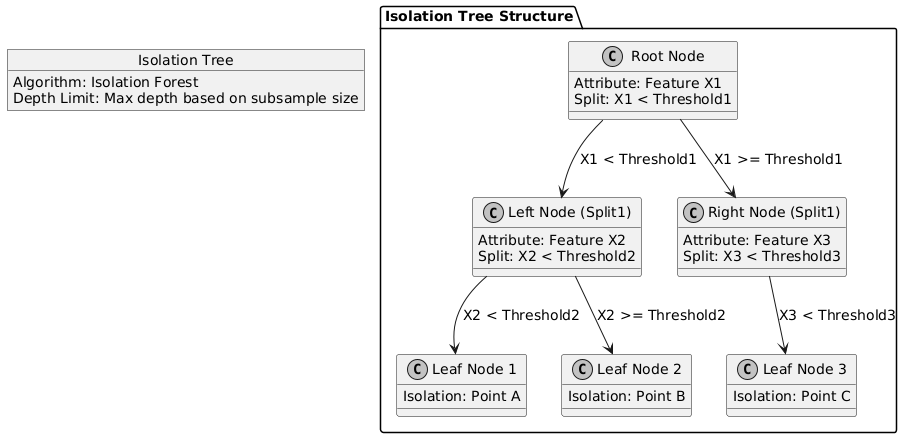
The main working principle of the Isolation Forest as follows:

1. Random Subsampling: In this step, we are going to randomly selecting the subset of the data from the dataset.
2. Tree Construction: This is the core step, here we are constructing the isolation tree by recursively partitioning the dataset. Now, for each node in the tree structure:

* We need to select data randomly.
* We need to select the split the value in the range between the minimum and maximum values for the current selected features.

1. Length of Isolation Path: In this step, the path length is calculated from the root to the current given point. We can find the anomalies if the data points are having shorter lengths.
2. Anomaly score: In this step, we need to calculate the score is selected as average path length taken from multiple trees, a higher score shows us or an indication of possibility of anomaly.
3. **Isolation Trees**

As we know, Isolation Forest based on tree-based method, it is very similar to Random forests. Here we are representing Isolation Tree as part of the Isolation Forest Model Algorithm. We have created a sample hierarchical tree structure where the nodes represent decision splits, and leaves are representing isolated data points. Here is the sample hierarchical structure of the Isolation tree.



**Fig: Isolation Tree**

1. **Advantages and Disadvantages:**
2. **Advantages:**

* Isolation forest is very efficient in respect of computation and have ability to deal with more complex datasets.
* One point to be noted is it do not form any assumptions about distribution of data.
* It is one of the most optimal algorithms to identify suspicious patterns.

1. **Disadvantages:**

* The performance of the Isolation Forest might impact because of choice of parameters and subsampling size.

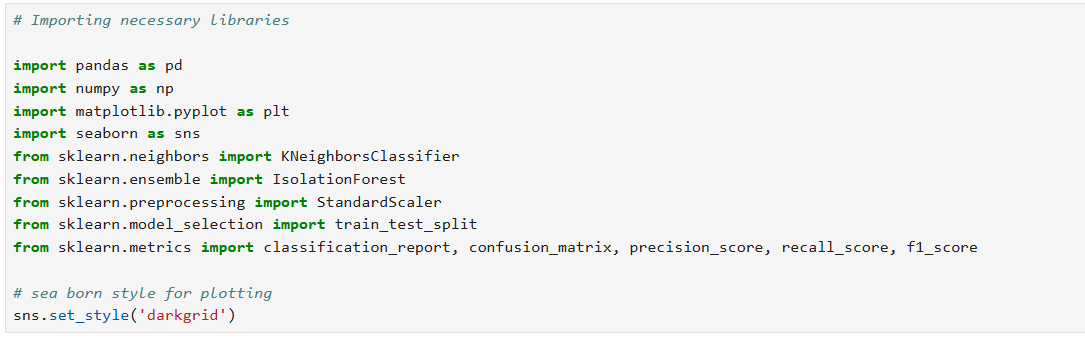
1. **Comparison between KNN vs Isolation Forest**

The comparison between the K-Nearest Neighbors and Isolation Forest model can be provided with respect of the table.

|  |  |  |
| --- | --- | --- |
| Feature | Nearest Neighbour(K) | Isolation Forest Model |
| Type of Data | It mainly focuses on labelled dataset. | It mainly deals with the un-labelled dataset. |
| Complexity | The complexity exponentially increases with size of the dataset. | It is efficient with large datasets also by using random samplings. |
| Use Cases | It uses the density-based methods to detect anomalies. | It uses the isolating points to detect anomalies. |
| Interpretability | This is transparent through neighbour relationships. | It is more abstract approach by using Isolating tree structures. |
| Scalability | It might struggle with the large datasets. | It handles with more complex high-dimensional datasets as well. |

1. **Step-by-Step Implementation**
2. **Import Required Libraries**

Importing the required libraries include pandas, NumPy, matplotlib, seaborn, and scikit-learn.



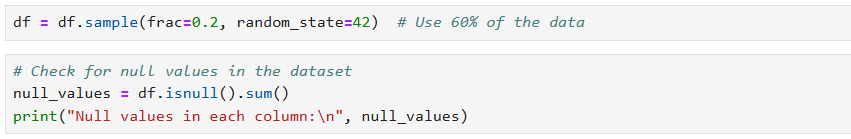
1. **Load and Explore the Dataset**

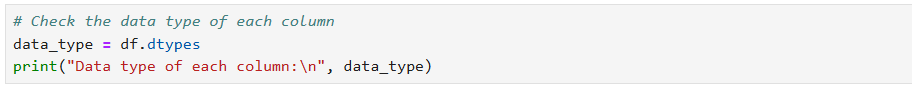
Load the NETWORK\_ANOMALY\_DATA dataset and explore its features to understand the data.

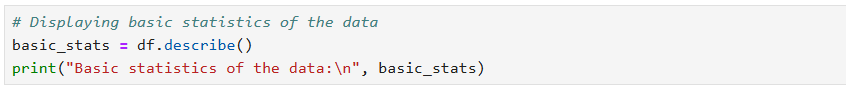


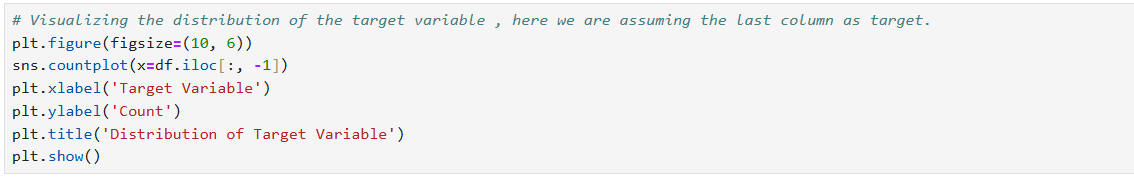
1. **Preprocess the Data**

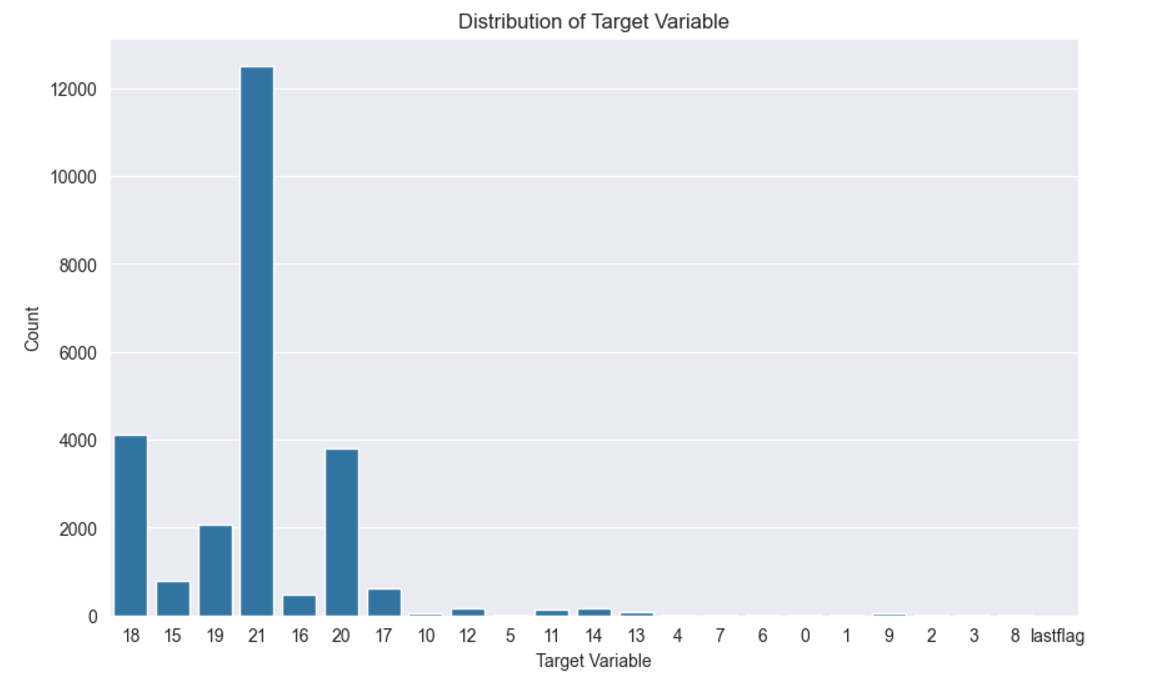
In this step we are converting the categorical features to numeric data using one-hot encoding technique. Here we are handling the missing data or values, encoding categorical data or values, and split the data into features and labels.

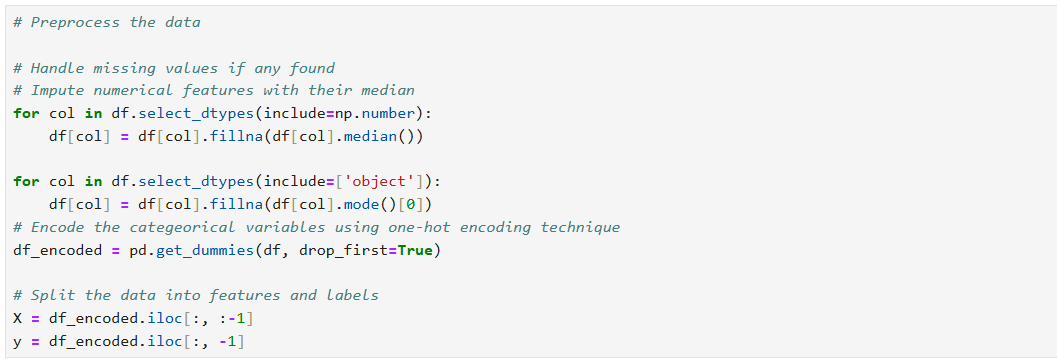








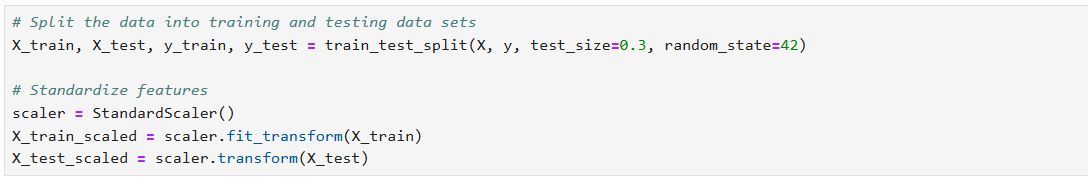


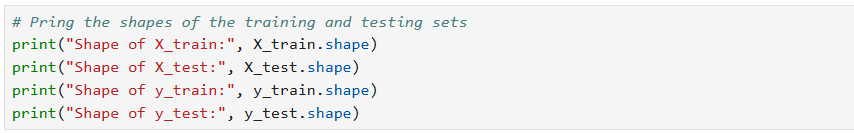


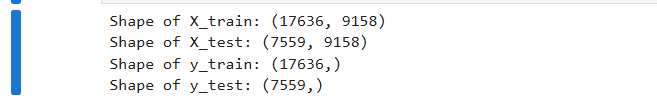
1. **Split the data**

In this phase the train\_test\_split method is used to slice the overall data into two slices called train data and test data. Here the data is split into features (X) and labels (y).

* Train set: Training dataset is used to train the machine learning model, the dataset on which model was trained on.
* Test set: Test data is kept separately that is taken as subset of the dataset, which is utilized to evaluate the model accuracy.

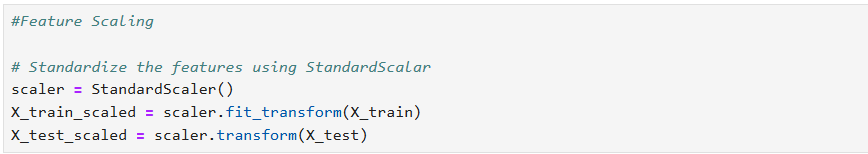






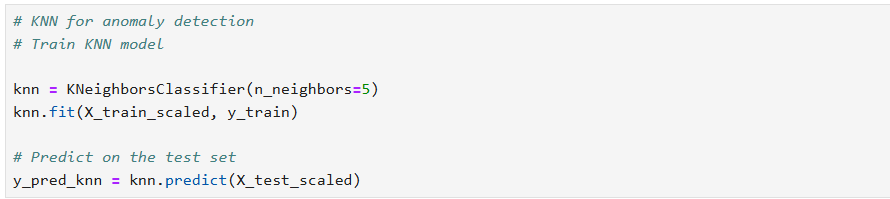
1. **Feature Scaling**

As we know that feature scaling is process of normalizing the all the preprocessed features in a given dataset. It is the approach to implement preprocessing that transforms all the feature values to similar scale, by ensuring the features equally contribute to the machine learning model.



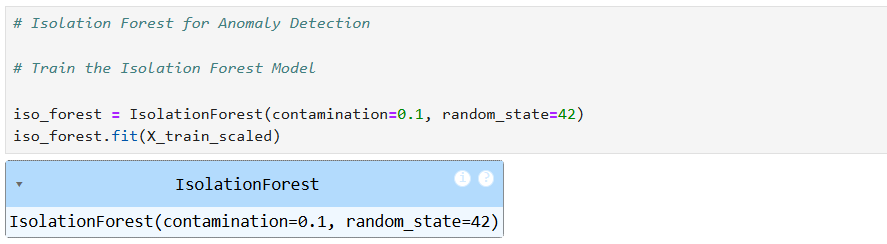
1. **K-Nearest Neighbor for Anomaly Detection**

KNN is a lazy learning machine learning model, it classifies the data points based on the neighboring datapoints. This algorithm is effective for anomaly detection as it an identify outliers based on its distance from its neighbors.



1. **Isolation Forest for Anomaly Detection**

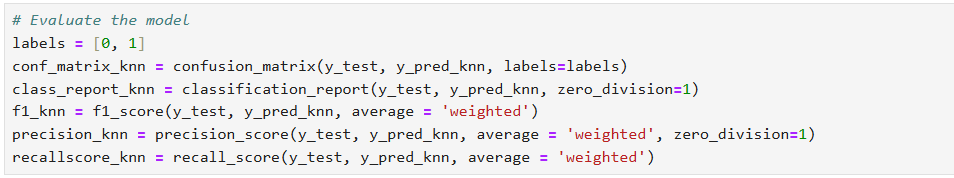
This algorithm also contributes for the anomaly detection method, it is an ensemble learning method specifically designed for detecting anomalies. It follows the tree-based structure anomalies are detected in the algorithm can be detected as they have shorter paths in the tree structure.

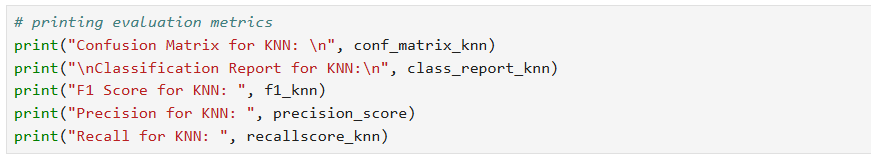


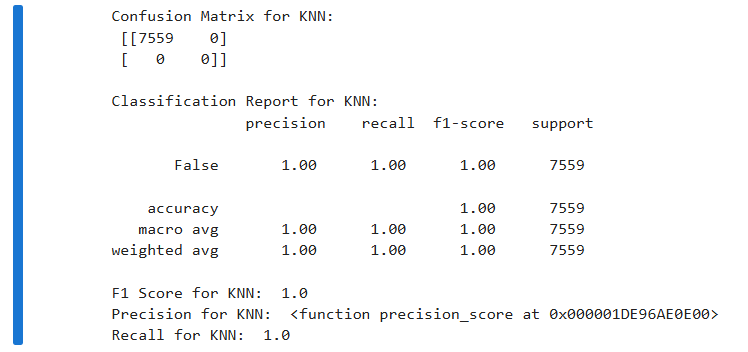
1. **Evaluation of K-Nearest Neighbors (KNN)**
2. **Classification Report**

The classification report provides the detail report for the KNN model based on the test data prediction. The report includes the measurable metrics such as (Liu, C., et al. ,2022, p.729) recall, F1-Score, precision for each class (normal and anomaly).

* Precision: It is a metric which shows the ratio of the true positive predictions with the total predicted positive results. It gives the accuracy of the positive predictions.
* Recall: It is a metric mainly gives the ratio of the true positives with the total actual positives. It measures the relevance or similarity score between the points.
* F1-Score: It provides the metric that balances both precision and recall. In other words, we can say that hormonic mean of precision and recall.





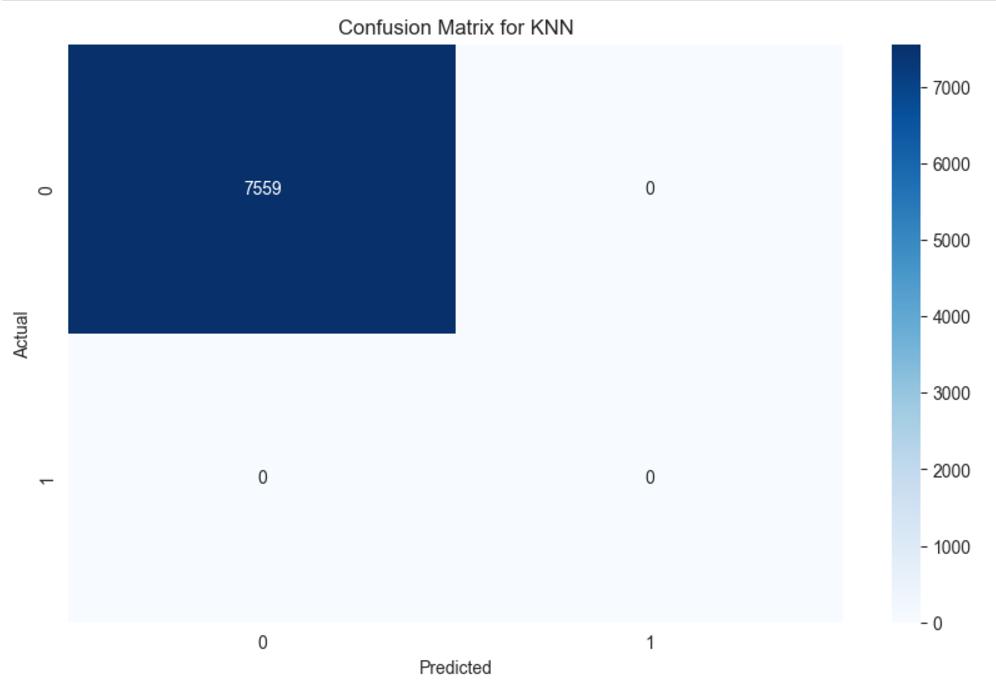


1. **Confusion Matrix: KNN Model**

This confusion matrix provides the overall summary of the KNN model. It provides overall summary of the KNN model prediction on test data. It shows the total number of true positives, false positives, true negatives, false negatives.

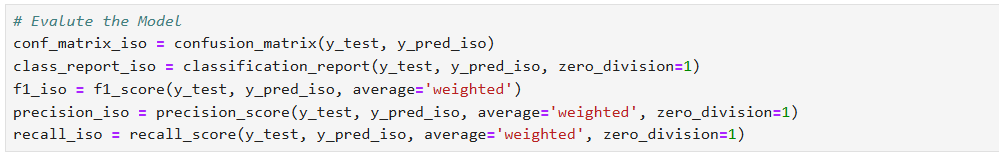
* True Positives (TP): If it correctly predicts only positive data points.
* True Negatives (TN): If it correctly predicts only negative data points.
* False Positives (FP): If it correctly predicts the positive instances.
* False Negatives (FN): If it correctly predicts only negative data points.

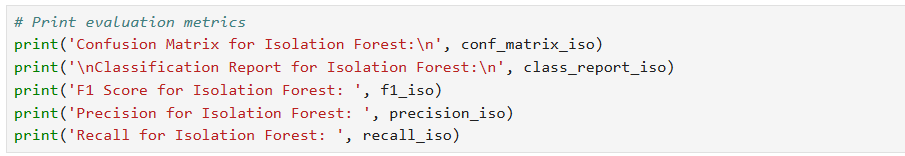


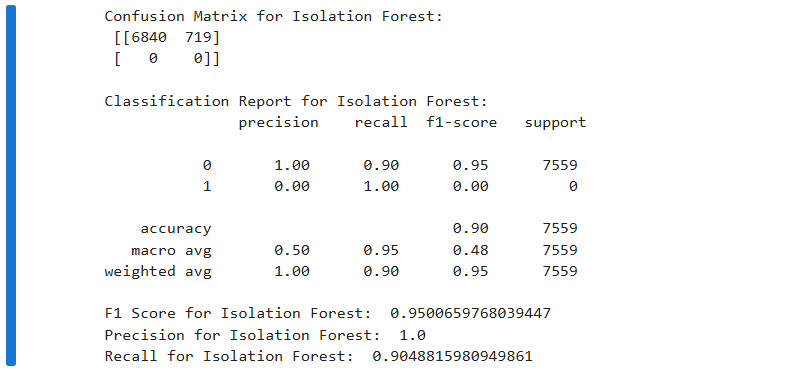


1. **Evaluation of Isolation Forest**
2. **Classification Report: Isolation Forest Model**

Here, the classification report for the Isolation Forest model provides detail metrics to understand the performance of how Isolation Forest Model is performing, (Liu, C., et al. ,2022, p.729) here the metrics include: precision, f1-score, recall.

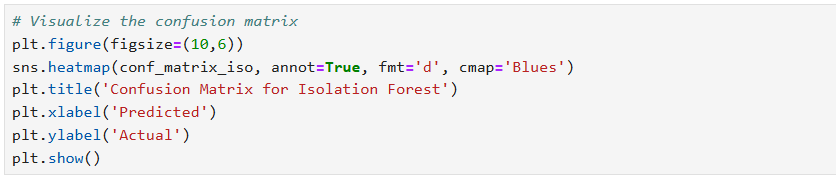


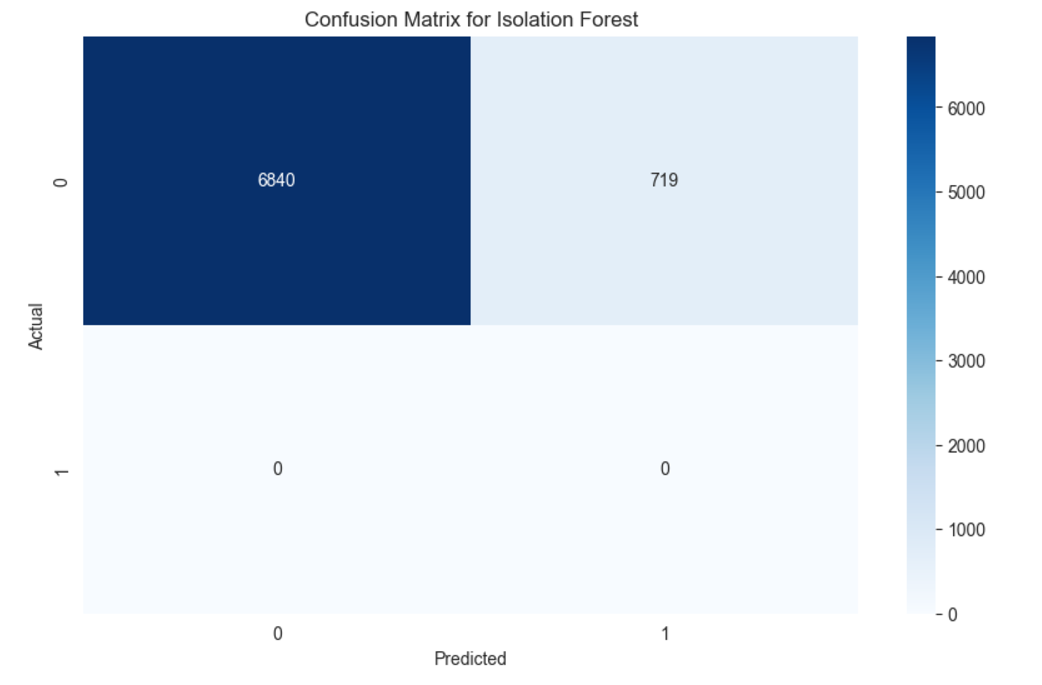




1. **Confusion Matrix: Isolation Forest Model**

Here, as we know the confusion matrix provides the overall summary of the Isolation Forest Model. It provides overall summary of the Isolation Forest prediction on test data.





1. **Results & Discussion**

|  |  |  |  |
| --- | --- | --- | --- |
| **Model** | **F1 Score** | **Precision** | **Recall** |
| KNN | 1.000000 | 1.0 | 1.000000 |
| Isolation Forest | 0.950066 | 1.0 | 0.904882 |

1. **KNN vs Isolation Forest Model**
2. **KNN Model**

* F1-Score: 1.000000
* Precision: 1.0
* Recall: 1.000000

As we can see the that KNN model achieved the perfect scores in all evaluation metrics that means the algorithm is able to detect all the anomalies without any false positives or false negatives. This evaluation metrics suggest that KNN is performing better for this given dataset.

1. **Isolation Forest Model**

* F1-Score: 0.950066
* Precision: 1.0
* Recall: 0.904882

This Isolation Forest Model is also performed well but less accurate than KNN. Here as we can see the precision scores is 1.0 indicating it predicted anomalies with high accuracy. But it had lower recall at 0.904882, and f1-score of 0.9500 which shows it misses some anomalies.

1. **Interpretation**
2. **KNN Model:** As we saw in the evaluation result KNN model demonstrated the perfect performance on the test and given a solution for our dataset is excellent choice for the KNN Model.
3. **Isolation Forest Model:** As per the results the Isolation Forest also performs well in terms of precision particularly. But we have seen that the results of the recall and f1-score bit missed some detection of anomalies. Overall, the performance was good and strong as per the evaluation results.
4. **Real Time Applications**

Here we are listing (*Chandola, V., Banerjee, A., & Kumar, V., 2009, p. 11*) real-time applications of the detecting anomalies using different algorithms just like KNN and Isolation Forest:

1. **Retail:** These algorithms can be used in determining the customer behaviors in purchasing or shopping scenarios to optimize the marketing strategies, mostly companies using these strategies to fetch the customer behavior analysis.
2. **Social media:** These algorithms can be integrated with social media platforms like X (Before Twitter), Instagram and other platforms to detect the fake news or fraud activities.
3. **Educational scenarios:** These types of algorithms used for the educational purposes to find the plagiarism or anomalies in writing styles of students or writing patterns of students.
4. **Weather patterns:** To detect anomalies in temperatures, pollution, seismic patterns, and rainfall anomalies we can predict the all the scenarios to determine unusual patterns.
5. **Telecommunications:** We can detect the spam or fraud calls from the scammers by collecting all the data of the customer and we can send a anomaly report to the network user.
6. **Stock Marketing:** We can find the irregular trading activities that can manipulate the stocks and mispresent the information of the stakeholders or stock holders.
7. **Human Resources:** Most of the companies are using these type of algorithms to optimize the recruitment process by filtering resumes, and for background verification and application processes.
8. **Aerospace and Defense:** We can detect the unusual patterns in flight data or any anomalies in the airplane status so that we can avoid risky scenarios for passengers’ safety.
9. **Others:** We can implement these algorithms in other scenarios such as “*Gaming and Entertainment, Logistics, E-commerce and other industrial applications.”*
10. **Conclusion**
11. **Future work**

As per our analysis and study the future enhancements of the anomaly detection algorithm can include:

1. Integrating Algorithms in Organization Infrastructures: We can integrate this type of anomaly detection algorithms with the organization infrastructures or any other government institutions then it is very helpful by early detection of intrusion or any other threats to the data systems.
2. Real time processing: By developing the algorithms like this in fraction of seconds the algorithms process the data in real time and they can provide immediate alerts if it found any anomalies or suspicious activities.
3. Integrating with AI: By Implementing the more sophisticated techniques such as deep learning we can develop more advanced and complex anomaly detection techniques then we can integrate with Robotics or Artificial Intelligence.
4. Privacy and Security: This type of algorithms secures the data of user or any other data like banking systems, government foreign affairs data and other data which are very important in many industries.
5. Self-Automated Devices Integration and Robotics: As we can observe many AI devices or self-driving automated devices such as Tesla cars, Optimus robots, we can find this type of algorithms integrated with those AI systems.
6. **Summary**

In this study, we have used or implemented two different algorithms including the K-Nearest Neighbors and Isolation Forest models to find or detecting the anomalies in the taken dataset NETWORK\_AMOMALY\_DATASET which is very similar to the KDD Cup 1999 dataset, both are related to the network intrusion data anomalies. To implement the Anomaly detection algorithm, we have followed certain steps to develop a optimized and more accurate model to detect the anomalies with more accuracy. The steps we have followed include: Data cleaning and preprocessing, model implementation, model evaluation, and visualization and comparison of two models we have implemented. As we have implemented both algorithms K-nearest-neighbors and Isolation Forest model, we observed that KNN performed with more accurate and reliable when compared to Isolation Forest model. As we can see the both model performance metrics such as F1-score, recall, and precision we have observed that KNN = {f1-score: 1.000000, precision: 1.0, recall: 1.000000} which are very accurate and precise i.e. KNN is able to detect the anomalies on the test data with more accuracy. On the other side, when we observe the Isolation Forest model performance metrics Isolation Forest = {f1-score: 0.950066, precision: 1.0, recall: 0.904882}, this model also performed very well in-terms of precision i.e. 1.0 which is very accurate to detect the anomalies on the test data but it disappointed with the metrics of f1-score and recall with values 0.950066 and 0.904882 respectively.

1. **References**
2. Liu, F. T., Ting, K. M., & Zhou, Z. H. (2008). Isolation Forest. IEEE International Conference on Data Mining. Focuses on the foundational principles of the Isolation Forest algorithm and its application to anomaly detection. [Reference-1](https://www.scirp.org/reference/referencespapers?referenceid=3054470)
3. Wang, L., Li, J., Bhatti, U. A., & Liu, Y. (2019). Anomaly detection in wireless sensor networks based on KNN. Artificial Intelligence and Security Conference. Discusses using KNN in sensor networks for anomaly detection. [Reference-2](https://link.springer.com/chapter/10.1007/978-3-030-24271-8_56)
4. Ahmed, S., Lee, Y., Hyun, S. H., & Koo, I. (2019). Unsupervised machine learning-based detection of covert data integrity assault in smart grid networks using Isolation Forest. IEEE Transactions on Information Forensics and Security. [Reference-3](https://dl.acm.org/doi/10.1109/tifs.2019.2902822)
5. Susto, G. A., Beghi, A., & McLoone, S. (2017). Anomaly detection through online isolation forest: Application to plasma etching. IEEE SEMI Advanced Semiconductor Manufacturing Conference. [Reference-4](https://www.researchgate.net/publication/318335190_Anomaly_Detection_through_on-line_Isolation_Forest_An_application_to_plasma_etching)
6. Spiekermann, D., & Keller, J. (2021). Unsupervised packet-based anomaly detection in virtual networks. Computer Networks. [Reference-5](https://www.sciencedirect.com/science/article/pii/S1389128621001286)
7. Hoang, D. H., & Nguyen, H. D. (2018). A PCA-based method for IoT network traffic anomaly detection. International Conference on Advanced Communication Technology. [Reference-6](https://www.researchgate.net/publication/324464302_A_PCA-based_method_for_IoT_network_traffic_anomaly_detection)
8. Gan, Z., & Zhou, X. (2018). Abnormal network traffic detection based on improved LOF algorithm. International Conference on Intelligent Human-Machine Systems. [Reference-7](https://www.researchgate.net/publication/328908273_Abnormal_Network_Traffic_Detection_Based_on_Improved_LOF_Algorithm)
9. Paulauskas, N., & Baskys, A. (2019). Application of histogram-based outlier scores to detect network anomalies. Electronics. [Reference-8](https://www.mdpi.com/2079-9292/8/11/1251)
10. An improved X-means and Isolation Forest methodology for network traffic anomaly detection. PLOS ONE (2023). Proposes a combination of X-means and Isolation Forest for detecting anomalies in network traffic. [Reference-9](https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0263423)
11. Zong, B., et al. (2018). Deep autoencoding Gaussian mixture model for unsupervised anomaly detection. International Conference on Learning Representations. [Reference-10](https://openreview.net/forum?id=BJJLHbb0-)
12. Erfani, S. M., et al. (2016). High-dimensional and large-scale anomaly detection using a linear one-class SVM with KNN. Pattern Recognition. [Reference-11](https://pdf.sciencedirectassets.com/272206/1-s2.0-S0031320316X00076/1-s2.0-S0031320316300267/am.pdf?X-Amz-Security-Token=IQoJb3JpZ2luX2VjEAUaCXVzLWVhc3QtMSJIMEYCIQCPussr4E%2Fb3a8ED7RC1973w2Er7rI5ltxWPknh1s9itgIhAJnBhnVRWMh2aF8PSqn%2BoWJGLQlEY4LIVayHM%2F3Fv7hPKrwFCL7%2F%2F%2F%2F%2F%2F%2F%2F%2F%2FwEQBRoMMDU5MDAzNTQ2ODY1Igwji0DbG5wgXkxZIcYqkAVGwe72KWb19B9c3yQNwLmZAVLHHT8YaRWu1xRTbKyrA66MZBHoMm5I4te2Q5hNL9tMlxLJ4KYewVpe8Q4noCP%2FsTWlwf52NrQBpHrQpuXHv%2FhJCScTaXSHnA4n%2Bstla8SYNj%2FQxzJrNJNYyt17iGcI9Ar8iO2GNLXvOHCzdVAo1WuP8ie2B6om8RxZ5AO7b1it8JEygsM13%2Bx3lqjcb0ZtL1IKThUq39vW2APuKwpmmsqOR6SmQnZwKqzgf2eB%2B16petmR3E1OyAJ%2FJgO1CQzge0D1zTgTqTOLyCl1GnVmy0s7M53VSgIq0i%2FTK%2FrKV8D13ZVpOzx69UA1M05AhP9FZnQJXP9Dil8e8QijBzCPiaVoEQvF1wq2hHxZq1IpWvv7cDKzgdVlUA%2ByFncBjk6W698RmYXXmmacL%2Bvc1SfaUguA2y09jqhn8KHEw1%2BunKcIjUsrCZsv%2FGAF5R4eJtLB23qSU46MN0vPyCIDDbiojM3NJZBQQlUYRUdyt%2FyAgYRgAApxX8emSE81F5W0r8sckN0CPjZ4p7ViG9Z2bwv3KUyw0g25CYvjudAy0g9DHLOW8MOFj9LlMyZYDfdPkcLmJvGQospDb0ipuzw1%2F1qrlyXr%2BRT5UB%2Bs%2Fe6ObU1%2ByWVene0FRcVs78PN412jT6rCB6FYJUpNjkRxwIkOAbgfzl6V5QlphY0cTF95u9XQUUN1Q8AC8lFjsRzLNSZj4zGkoPz%2BCLazN4%2B%2BfnzsIgOPGy6eosvN5NZcZ51G%2Bh%2FYnyNkPHh0t2DZJ%2FHaYv8HoLeIQ%2Fsq6lDToNqGERK1pDw9JjMAtJAOLoPALF5yXZLQRs4%2FirssZKXrsiBv2IxXoo795rK1OCO%2FcIYfvBAnAYntZzCCteu6BjqwAZ6xqnAIDcTXTZTChOMaP2byiTp5em05VDIzM3lBYFy6yIrxT6hwpLAjrlHRlN1SIpzl%2BayBtKBBOI0LWRi2VocLXJjjJXl4ZS0Id%2BYUTBOSSfULJkmmFfbHCiDHfir1IeKAkIKKj3eofi9vEmGXD9lFaY6FTksGFgBgxAJTxxEI9pbL9jVQbus1W%2BkYadQd%2FjUmN3ikaMWi8%2FIEp2THrfN7n4dG4rfhsS8Pmhl5%2BbWt&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Date=20241212T125942Z&X-Amz-SignedHeaders=host&X-Amz-Expires=300&X-Amz-Credential=ASIAQ3PHCVTYSA7MZHDN%2F20241212%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Signature=0d5503e1d71b0dc0e647fbdbf6e29acc7d7b9ebca3adbd8905c3f9cbe33e2ecf&hash=2a2a80a2ba386b54fea097e55d336b75d3dc7f4ef7558bedcecce0a97e6fa7b7&host=68042c943591013ac2b2430a89b270f6af2c76d8dfd086a07176afe7c76c2c61&pii=S0031320316300267&tid=pdf-63a1b8ad-8e75-45f4-a51b-f57c44556fd0&sid=b2cdd674295fb348b54ac1e8547057340c5cgxrqb&type=client)
13. Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. ACM Computing Surveys. Comprehensive overview of anomaly detection methods, including KNN and Isolation Forest. [Reference-12](https://www.researchgate.net/publication/220565847_Anomaly_Detection_A_Survey)
14. Liu, C., et al. (2022). Comparative analysis of KNN and Isolation Forest in detecting frauds in financial datasets. Springer AI Conference Proceedings. [Reference-13](https://link.springer.com/chapter/10.1007/978-981-97-4581-4_4)
15. Rahman, S., & Fung, B. C. (2020). Outlier detection using KNN over dynamic time warping distances. Data Mining and Knowledge Discovery. [Reference-14](https://www.researchgate.net/publication/232654743_KNN_Based_Outlier_Detection_Algorithm_in_Large_Dataset)
16. Zhao, Y., et al. (2019). PyOD: A Python Toolbox for Scalable Outlier Detection. Journal of Machine Learning Research. Implements algorithms such as Isolation Forest and KNN. [Reference-15](https://www.jmlr.org/papers/volume20/19-011/19-011.pdf)
17. Ding, C., et al. (2021). Hybrid machine learning techniques for anomaly detection in power grids. Renewable and Sustainable Energy Reviews. [Reference-16](https://www.sciencedirect.com/science/article/pii/S0306261921001409)
18. Zhou, Y., et al. (2020). Multi-strategy anomaly detection using a combination of Isolation Forest and clustering methods. Journal of Big Data.
19. Shen, S., et al. (2018). Adaptive Isolation Forest for anomaly detection in imbalanced datasets. Knowledge-Based Systems.
20. Jiang, Y., et al. (2017). Combining KNN and DBSCAN for robust anomaly detection in complex data distributions. Expert Systems with Applications. [Reference-19](https://www.sciencedirect.com/science/article/abs/pii/S0957417422000707)
21. Qiu, T., et al. (2020). A comparative study of machine learning techniques for anomaly detection in network traffic. IEEE Access. [References-20](https://www.researchgate.net/publication/377734681_A_Comparative_Study_of_Anomaly_Detection_Techniques_for_IoT_Security_using_AMoT_Adaptive_Machine_Learning_for_IoT_Threats)