OCSVM Anomaly Detection

[Problem 1] Run ‘OCSVM’ with the same dataset which used in practice and compare the result with the result of Isolation Forest and LOF.

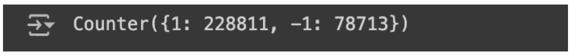
1. Show OCSVM result using ‘Matplotlib’

2. Describe whether there is the significant difference from the practice or not

Overview: This assignment involved implementing and analyzing an anomaly detection workflow using machine learning techniques to parse the log data, preprocess it, and apply two anomaly detection methods: Isolation Forest and Local Outlier Factor (LOF). The goal was to identify and visualize anomalous data points on a provided log file (`SotM30-anton.log`).

Steps:

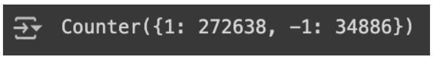
- Parse the log data using a function `parse_log` which was implemented to extract relevant attributes from each log entry. The parsed logs were then converted into a Pandas DataFrame, allowing for structured data manipulation.
- Split the df split into two parts:
  - `key`: extracting columns of`time`, `state`, `SRC`, `DST`, and `MAC`.
  - `data`: all other columns which were further categorized into non-numeric, hexadecimal numeric, and numeric columns.
- Non-Numeric Data: columns factorized to convert categorical values into numerical representations, facilitating their use in machine learning models.
- Hexadecimal Numeric Data: columns converted from hexadecimal to floating-point numbers.
- Numeric Data: columns converted to floating-point numbers for consistency.
- Preprocessing data:
  1. Isolation Forest: isolates observations by randomly selecting a feature and then randomly selecting a split value between the maximum and minimum values of the selected feature. The number of splits required to isolate a point is used as a measure of normality; anomalies require fewer splits.



```
Counter({1: 228811, -1: 78713})
```

means 78,713 anomalies out of 307,524 data points.

  2. Local Outlier Factor (LOF): measures the local density deviation of a given data point with respect to its neighbors. Points with substantially lower density than their neighbors are considered anomalies.

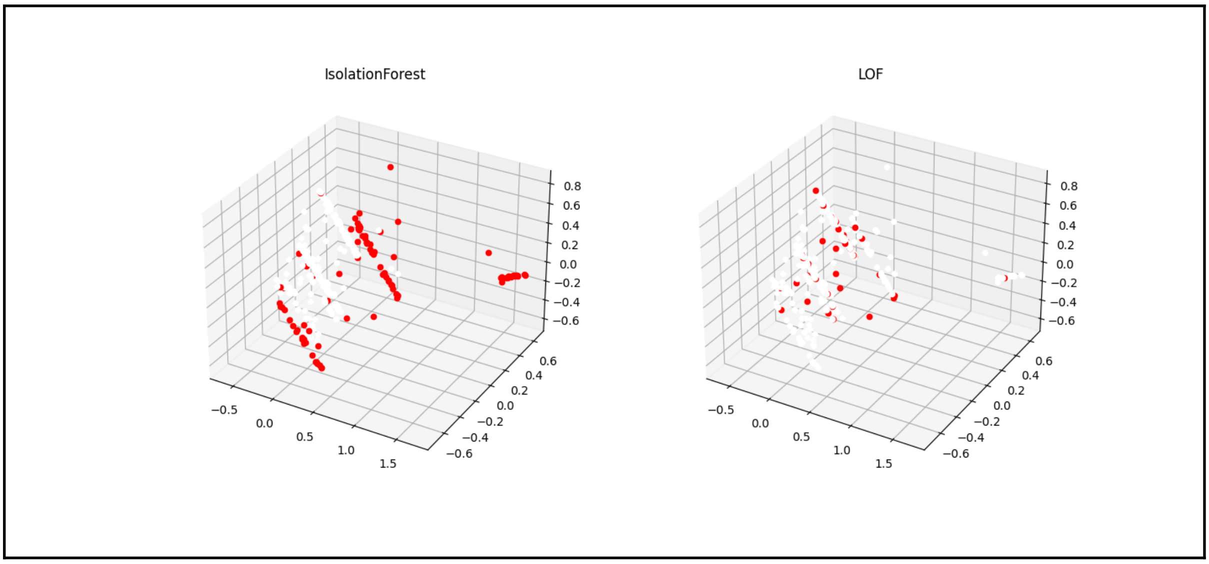


```
Counter({1: 272638, -1: 34886})
```

means 34,886 anomalies out of 307,524 data points.

- Visualization:
    - PCA was applied to reduce the data to three components for 3D visualization.
    - Plots highlighted normal points in white and anomalies in red.

Result:



Conclusion:

LOF identified fewer anomalies compared to Isolation Forest. Isolation Forest's approach is more global, focusing on how easily points can be isolated and tends to identify more anomalies because it considers points that are far from the majority as anomalous, regardless of density, while LOF's approach is more local, relying on density-based comparisons with neighboring points which tends to be more conservative as it requires points to have significantly lower density compared to their neighbors.

IF is more sensitive which might include more false positives making it useful for initial broad scans but might require further investigation to filter out the false positives.

LOF is more specific focusing on points that are truly isolated based on local density, being more likely to have fewer false positives but might miss some global anomalies that are not locally sparse which is better suited for environments where precision is more important, and the cost of investigating false positives is high.

The significant difference in the number of anomalies detected by the two methods highlights the importance of understanding the underlying mechanics of each anomaly detection algorithm. The choice of method can significantly impact the results and their interpretation. The high number of detected anomalies suggests that the log data contains many points that significantly differ from the majority. This could indicate potential security breaches, network issues, or other irregularities in the system generating the logs.