30/12/2024-12/01/2025

Week 6-7

In this week’s developments, various outlier detection methods were combined to create a more comprehensive model. Specifically, the results from different outlier detection techniques were integrated into the CNN model. Below are the detailed changes and improvements made this week:

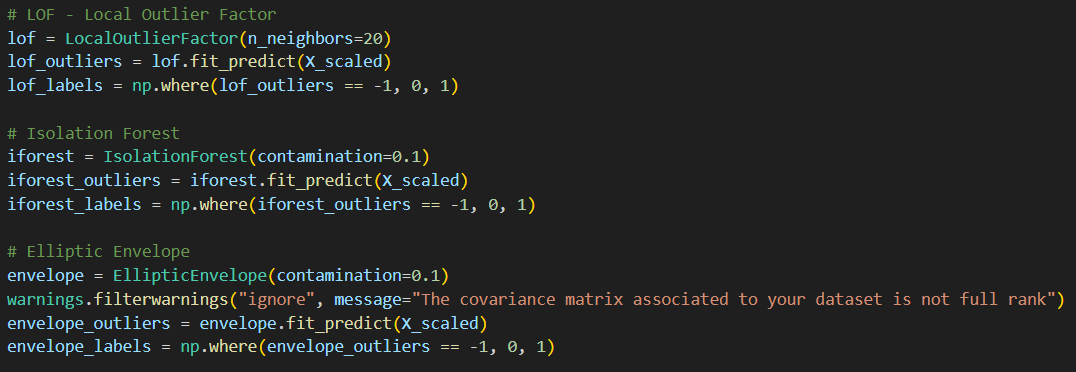
1. Combination of Outlier Detection Methods:

Initially, only the One-Class SVM (OCSVM) method was used in the code from Week 5. This week, new outlier detection methods were added:

- LOF (Local Outlier Factor): This method measures how “unusual” data points are compared to their neighbors.

- Isolation Forest: This method isolates data points to detect anomalies.

- Elliptic Envelope: This method models the distribution of the data and detects outliers based on that.



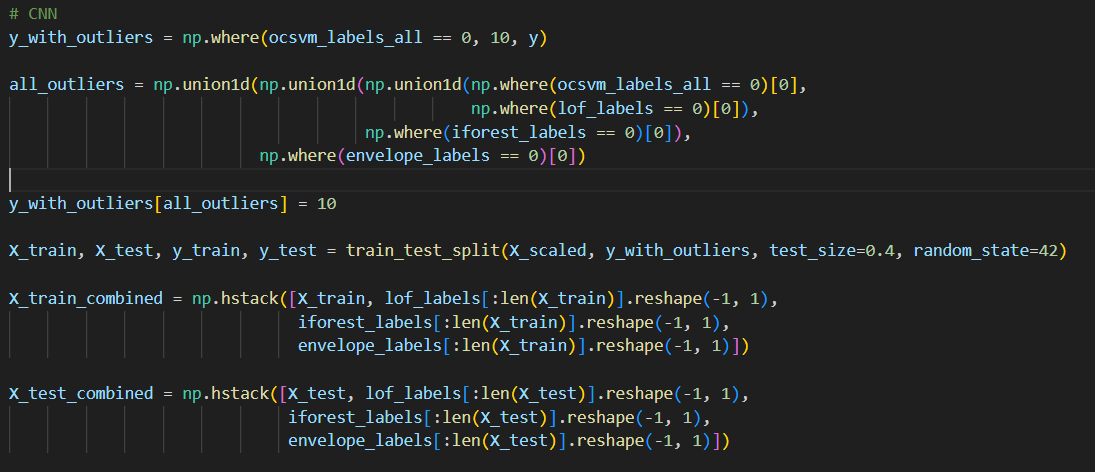
These methods were applied to detect outliers in the dataset separately, and their results were integrated into the CNN model for a combined assessment. This allowed for a more comprehensive evaluation by incorporating outliers detected by various methods into the model.

2. Integration of Outlier Labels into the CNN Model:

The outliers detected by each method were integrated into the CNN model to allow the model to better handle these anomalous data points. A new class (Class 10) was introduced to label outliers. This modification meant that outliers were both used as training data and included in the model’s predictions.

Expansion of the Training Data: The outlier labels were included as part of the training data, which enabled the model to train on a broader dataset. This helps the model generalize better.

Outlier Detection: In the CNN model’s output, the 10th class represents outliers. The accuracy of this class is enhanced by integrating outliers detected by the other methods into the model.



3. Performance Evaluation:

Initially, the CNN model had an accuracy of around 97% and could detect about 10 outliers. After integrating the results of other outlier detection methods into the CNN model, the accuracy dropped to around 94%. However, this change allowed the model to detect more outliers, with the current detection count at approximately 100 outliers. This is a positive development as it shows the model is able to detect a wider range of anomalies.

metin, elektronik donanım, ekran görüntüsü, bilgisayar içeren bir resim

Açıklama otomatik olarak oluşturuldu

4. Hyperparameter Tuning:

In addition to the changes made in combining outlier detection methods, hyperparameter tuning played a crucial role in improving the model’s performance. Specifically, the nu and kernel parameters of the One-Class SVM (OCSVM) were adjusted.

Nu Parameter: The nu parameter controls the fraction of outliers in the dataset. It determines the upper bound on the fraction of margin errors and the lower bound on the fraction of support vectors. The optimal selection of nu helps balance the detection of outliers without overfitting the model to noise. Four different values for nu were tested:

• 0.01

• 0.05

• 0.1

• 0.7

Each of these values was evaluated based on the number of outliers detected and the model’s overall performance. The selection of nu directly impacted the trade-off between false positives and false negatives in the outlier detection process.

Kernel Parameter: The kernel parameter specifies the kernel function to be used in the One-Class SVM. Four different kernel functions were tested to identify the best approach for handling the data’s underlying structure:

Linear Kernel: Suitable for data that is linearly separable.

Polynomial Kernel: Useful for data with non-linear relationships.

RBF Kernel (Radial Basis Function): Commonly used for high-dimensional data and non-linear problems.

Sigmoid Kernel: Less commonly used but can be effective in certain situations.

Each kernel was evaluated in terms of model accuracy, the ability to detect outliers, and the time taken for training. The results helped determine the best combination of nu and kernel for the task at hand.

After evaluating different combinations of the nu values and kernels, the optimal values were chosen to strike a balance between performance and detection of anomalies. This tuning contributed significantly to the overall effectiveness of the model, ensuring a better fit for the data and an improved outlier detection rate.

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

5. Graphical Enhancements and Visualizations:

Significant improvements were made in terms of graphical visualizations. The results of the PCA transformation of the data were plotted, showing the outliers detected by each method in different colors and labels. Some of the key graphical improvements include:

Added Jitter: To make the outlier points more distinct, a small random jitter was added to the data. This helped prevent overlapping points, especially in dense areas.

Color Coding: Each outlier detection method is now visually distinguished by different colors. OCSVM outliers are shown in red, LOF, Isolation Forest, and Elliptic Envelope outliers are in orange, and outliers detected by the CNN model are in blue.

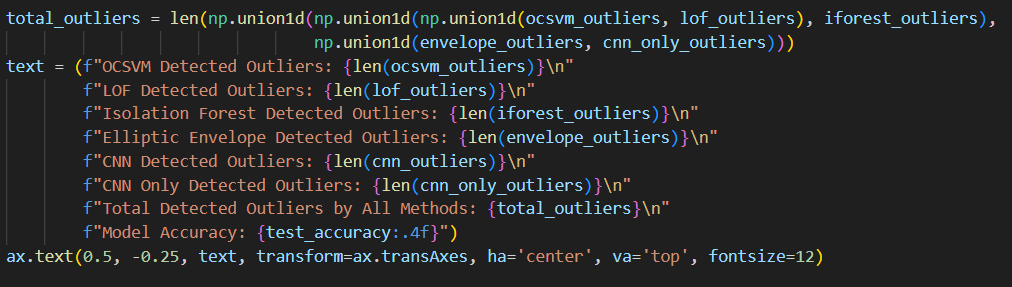
Text Annotations: Text was added to the graph to show the number of outliers detected and their accuracy. Additionally, a “gradient text” function was added to enhance the visual appeal of the annotations.

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

6. Calculation of Total Outlier Count:

The total number of outliers detected was calculated by combining the outliers detected by each method. This information was visualized on the graph, providing an easy-to-understand count of outliers detected by the combined methods. Currently, the CNN model detects around 100 outliers, many of which are the result of combining the detections from the other methods.



7. Results and Evaluation:

As a result of these changes, the model’s accuracy dropped slightly to accommodate the detection of more outliers. However, this trade-off has significantly improved the model’s ability to detect anomalies.

Performance metrics and test results are as follows:

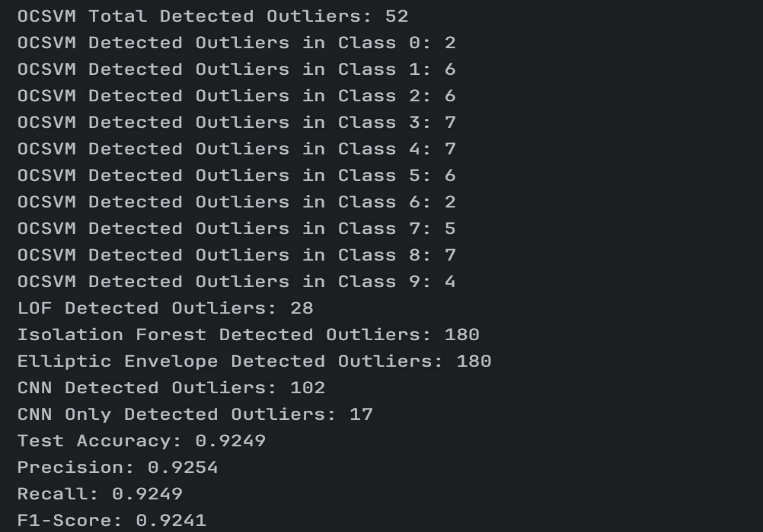
• Accuracy: 92.11%

• Precision: 92.12%

• Recall: 92.11%

• F1-Score: 92.18%

With these developments, the model now demonstrates improved overall performance and can detect a wider range of outliers. The integration of multiple outlier detection methods has expanded the model’s capabilities, allowing for more comprehensive anomaly detection.



Referencess:

- <https://scikit-learn.org/stable/auto_examples/svm/plot_oneclass.html>

- <https://www.analyticsvidhya.com/blog/2024/03/one-class-svm-for-anomaly-detection/?utm_source=chatgpt.com>

- <https://stats.stackexchange.com/questions/402493/how-to-tune-the-hyperparameters-for-oneclass-svm-while-doing-unsupervised-learni>

- <https://stats.stackexchange.com/questions/347185/what-is-the-parameter-nu-in-oneclass-svm>

- <https://www.geeksforgeeks.org/svm-hyperparameter-tuning-using-gridsearchcv-ml/>

- <https://www.geeksforgeeks.org/understanding-one-class-support-vector-machines/>

- <https://www.geeksforgeeks.org/understanding-one-class-support-vector-machines/>

- <https://stackoverflow.com/questions/44698928/is-there-a-way-to-perform-grid-search-hyper-parameter-optimization-on-one-class>

- <https://stackoverflow.com/questions/11230955/what-is-the-meaning-of-the-nu-parameter-in-scikit-learns-svm-class>

- <https://www.analyticsvidhya.com/blog/2024/03/one-class-svm-for-anomaly-detection/>

- <https://www.restack.io/p/hyperparameter-tuning-answer-one-class-svm-cat-ai>

- <https://medium.com/@prayushshrestha89/tuning-svm-hyperparameters-making-your-classifier-shine-like-a-pro-8673639ddb16>

- <https://medium.com/@prayushshrestha89/tuning-svm-hyperparameters-making-your-classifier-shine-like-a-pro-8673639ddb16>

- <https://sites.gatech.edu/omscs7641/2024/02/16/tutorial-on-hyperparameter-tuning-using-scikit-learn/>