Anomaly Detection Workshop – Reflection Sheet

Name(s): Carmel Gafa **Date:** 13.05.2025

Instructions

*Work individually or in pairs. Answer concisely (bullet points are fine). Save this file (or export to PDF) and email it to Charlie by **18:00 CET tomorrow**. Feel free to reference code, figures, or metrics from your notebook runs.*

Notebook 01 – Statistical Foundations

1. **Conceptual distinction**

Give one real-world example of a *contextual* anomaly and explain why a univariate Z-score could miss it.

A pedometer recording 13km run at 11:30 in August during a heat wave.

2. **Robust Z-score metric**

On your run, what PR-AUC did the Robust Z-score achieve? Briefly interpret its curve shape (early precision vs recall).

The Robust Z-score method achieved a PR-AUC of 0.0659 on this run.

3. **Mahalanobis insight**

Did Mahalanobis distance outperform Z-score for this dataset? If yes, what property of the data explains the gain? If no, why might Mahalanobis have struggled?

Yes, Mahalanobis distance outperformed the Robust Z-score on this dataset, achieving a PR-AUC of 0.1478.

This improvement is due to Mahalanobis distance's ability to consider the correlation between features, which the Z-score completely ignores.

The Robust Z-score method operates univariately, computing absolute deviations from the median and aggregating them across features. This approach treats each feature independently and struggles with detecting anomalies that arise from unusual combinations of otherwise normal values.

In contrast, Mahalanobis distance computes the distance of a point from the mean of the distribution, scaled by the inverse covariance matrix:

$$D(x) = \sqrt{(x - \mu)^T \Sigma^{-1} (x - \mu)}$$

This formulation assigns high scores to points in low-density regions of the multivariate distribution, even if their features appear normal individually.

4. **Threshold sensitivity**

Predict the effect of lowering the Z-score threshold from the 99.9-th percentile to the 99.5-th. Which error types rise?

Intuitively, lowering the Z-score threshold from the 99.9th percentile to the 99.5th percentile should means more data points will be classified as anomalies, making the model more permissive. But in this case the PR-AUC metric did not change.

Keeping in mind that

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

Lowering the threshold can include more samples as predicted anomalies, improving recall by capturing more true positives (TP).

However, in this case, it also increased the false positive (FP) proportionally, so the overall balance between precision and recall remained the same, resulting in the same PR-AUC.

Notebook 02 – Unsupervised Methods

5. **Model ranking**

Rank Isolation Forest, One-Class SVM, and Autoencoder by PR-AUC on your run. State one strength that justifies the top model's performance.

In this study, the following results were obtained:

Isolation Forest PR-AUC=0.708

Autoencoder PR-AUC=0.763

One-Class SVM PR-AUC=0.924

One-Class SVM constructed a tight boundary around the normal data distribution, using an RBF kernel, allowing it to capture non-linear structure and reject points outside high-density regions. Its strength lies in modelling the support of the normal class rather than separating it from anomalies, making it especially effective when anomalies are sparse and structurally distinct.

6. **Hyper-parameter impact**

Choose a hyper-parameter you tuned (e.g., `contamination`, ν, latent dimension). Describe how changing it affected the Precision-Recall curve.

The following results were obtained from varying the latent dimension and batch size:

Encoding Dim	Batch Size	PR-AUC
5	32	0.766
5	64	0.718
5	128	0.777
5	256	0.774
10	32	0.806
10	64	0.739
10	128	0.816
10	256	0.721
15	32	0.766

Encoding Dim	Batch Size	PR-AUC
15	64	0.783
15	128	0.808
15	256	**0.835**

This result confirms the intuition that larger latent spaces allow the model to preserve more information from the input, improving its ability to reconstruct normal data and isolate anomalies based on reconstruction error.

Additionally, larger batch sizes stabilise training, yielding more consistent performance gains, particularly at higher encoding dimensions. While encoding dimension 10 showed some promise, its performance was more sensitive to batch size changes, suggesting it may be less robust.

7. **Metric selection**

In a domain where false negatives are costlier than false positives, which evaluation metric would you optimise during tuning and why?

In this case, recall is the most appropriate evaluation metric to optimise during tuning, which measures the proportion of actual anomalies (positives) that are correctly detected. Optimising for recall ensures that as few anomalies as possible are missed, even if this means accepting a higher number of false positives.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

8. **Method suitability**

Name a scenario where OC-SVM might beat Isolation Forest, and justify based on data characteristics.

As OC-SVM with an RBF kernel can model complex, non-linear boundaries around the normal data, effectively capturing density contours, a scenario where One-Class SVM might outperform Isolation Forest is when the data lies on a smooth, non-linear manifold with a clear, compact normal region surrounded by sparse anomalies.

An example would be a sensor-based condition monitoring system.

Notebook 03 – Explainability

9. **IF SHAP reading**

In your per-instance SHAP force plot, which feature had the largest red contribution and what actionable insight does that provide?

The force plot generated in this study shows that the feature with the most significant red contribution is failed_logins_last_5min, with a value of 13.0.

This feature, therefore, contributes most to the model's decision to classify the event as anomalous.

10. **AE SHAP caveat**

Why can SHAP explanations of raw reconstruction error become noisy in very high-dimensional data, and how could you mitigate this?

Why SHAP Explanations Become Noisy in High Dimensions

- In high-dimensional spaces, the total reconstruction error is spread across many features. Each feature may contribute only a small amount, making individual SHAP values less distinguishable from noise.
- High-dimensional data often contains correlated or redundant features. SHAP assumes feature independence, and when this assumption is violated, it can lead to unstable and noisy attributions.
- As dimensionality increases, the number of possible feature combinations grows exponentially. This complexity can make SHAP's estimation of feature contributions less reliable.

To mitigate these issues, several strategies can be employed:

- Reducing the number of features by eliminating irrelevant or low-variance ones.
- Techniques like Principal Component Analysis or autoencoder bottlenecks can project high-dimensional data into lower-dimensional spaces, preserving essential information while simplifying the structure for SHAP analysis.

11. **Trust & regulation**

Draft a one-sentence, GDPR-compliant explanation you could provide to a user whose transaction was flagged anomalous by the IF model.

Your transaction was flagged by our automated system because it exhibited patterns that significantly deviated from your typical activity, indicating potential unauthorized access.

12. **Model debugging**

Describe one way SHAP helped you spot a potential model weakness or data issue during the workshop.

The model heavily relies on a single feature, failed_logins_last_5min, to drive anomaly scores upward, suggesting a lack of robustness or feature redundancy.

The model might overfit to obvious outliers and ignore more subtle anomalies.

Final reflection

13. **Key takeaway**

What was the most surprising or valuable thing you learned across all three notebooks? (≤3 sentences)

One of the most valuable things I learned was the use of SHAP and Mahalanobis distance for interpreting and detecting anomalies, both of which were new concepts to me. I also implemented Isolation Forests, One-Class SVMs, and Autoencoders for the first time, gaining hands-on experience with multiple unsupervised anomaly detection techniques. Additionally, I deepened my understanding of how imbalanced datasets affect evaluation metrics and discovered Youden's J statistic as a useful thresholding tool for ROC analysis.