Group assignment: Is High-Dimensional Analysis Worth it?

Polina Revina Pei-Chi Hsieh

Abstract

In the context of semiconductor manufacturing, where data is collected from sensors across multiple production stages, detecting anomalies is crucial to prevent equipment damage and production delays. This research addresses the problem of identifying rare anomalies in a high-dimensional, imbalanced dataset. The primary research question focuses on determining which unsupervised anomaly detection method—K-Nearest Neighbors or Isolation Forest—performs more effectively under these conditions. Both methods are evaluated using cross-validation and compared based on their F1-scores on a labeled validation set.

1 Introduction

The semiconductor industry originated in the late 20th century and has become one of the most important components of the global economy. Nowadays, almost every electronic device has semiconductors, including mobile phones, cars and household appliances. In the semiconductor manufacturing industry, the occurrence of abnormal values can lead to premature damage to equipment parts, significantly increasing maintenance costs and potentially delaying the overall production schedule. A small abnormality can result in an entire batch of wafers being scrapped, while delayed detection of equipment failure may cause a complete production line shutdown, further driving up production costs, extending delivery times, and disrupting the overall production plan. These issues can lead to a decline in the performance of manufactured products or even non-compliance with quality standards, ultimately damaging the enterprise's reputation. Additionally, such disruptions can strongly impact the downstream supply chain, compounding the overall effect.

2 Data

We utilise a dataset gathered from one of India's leading semiconductor manufacturers, consisting of 1,763 rows and 1,559 columns, where each row represents an observation recorded every 10 milliseconds ((**machinehack2024wafer?**)). These features are derived from sensor readings across multiple manufacturing equipment, reflecting a range of operational parameters and conditions at different stages of production. The dataset is anonymised, with feature names hidden, requiring domain knowledge to interpret the

data. As this data comes from a competition, we have a validation set that has labels of anomalies that are marked by experts.

Anomalies are highly rare, as shown in Figure 1, so the outcome is imbalanced, with anomalies representing a tiny proportion of the samples. The increasing dimensionality of the data introduces additional challenges, as data points become sparse, and traditional distance-based metrics lose their effectiveness. To navigate this, we can use methods that are more efficient in these settings.

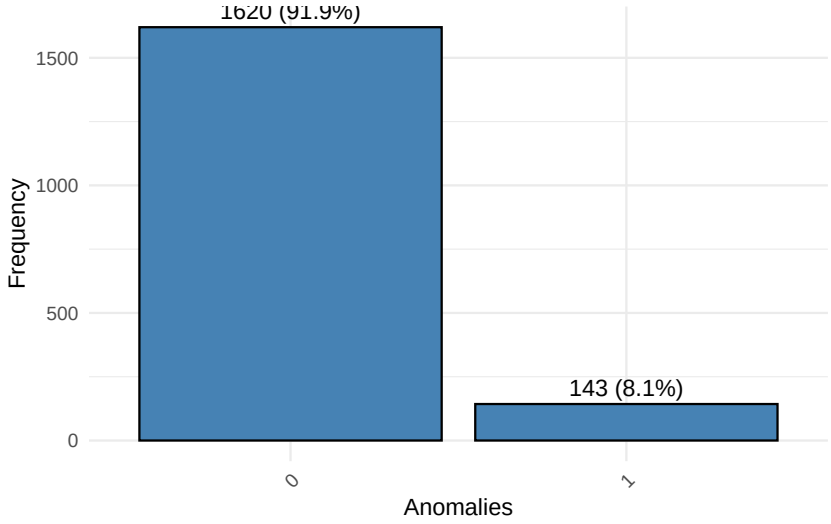

Figure 1: A bar-plot showing number of anomalies

Dataset contains 3 numeric features, others are binary indicators, which presents additional challenge for data preprocessing. Data don't consist any missing values but some features had zero variance so they were deleted. Distribution of numeric features is highly skewed with lots of outliers. However, as all features are anonimized, it's hard to determine the origin of anomalies, so they weren't deleted. There are also 40 duplicated rows but without any domain knowledge they can't be deleted.

To ensure that all features are on a comparable scale, which is essential for machine learning algorithms that rely on distance metrics or optimization, data standartization is applied. Without it, features with larger range can skew learning process and lead to biased models. As our data has lots of outliers, robust scaling is applied, t uses the median and interquartile range, which are resistant to extreme values.

As discussed previously, the target is highly imbalanced, which can lead to model favoring the majority class. To address this, we employed oversampling to increase the representation of the minority class. We chose SMOTE (Synthetic Minority Oversampling Technique) as it generate synthetic samples by interpolating existing samples, rather than duplicating them. On the other hand, random oversampling can lead to overfitting, as it only duplicates minority class and doesn't introduce new information. To handle a lot of categorical variables, SMOTE-NC is applied.

3 Methods

In our training set as in many real-world anomaly detection problems, labels for anomalies are unavailable. Thus, unsupervised methods will be used, as they allow the model to detect anomalies without needing any predefined labels for the data points. It is also important for further scalability for real-time monitoring of systems and generalization of our solution to unseen data. Several methods will be compared: KNN and isolation forest.

3.1 K-Nearest Neighbors (KNN)

KNN is a machine learning algorithm used for classification by determining the majority class of data points based on the similarity between a point and its k-nearest neighbours in feature space. The K here refers to the number of clusters, which is specified apriori. The algorithm computes the distance between the observation and its neighbors, typically using Euclidean distance. In the context of the task, the algorithm assumes that normal data points are clustered together, while anomalies are isolated and have fewer close neighbors.

As in high-dimensional spaces data points tend to become almost the same, making it difficult to distinguish observations effectively. Thus, the similarity of observations become less reliable, which could lead to poor performance. With high number of features, the computational cost of algorithms also grows, which can be inefficient in terms of resources and isn't suitable for scalable solution. It's also unable to handle noise and outliers due to assumption of spherical clusters (Xu et al. (2023))

3.2 Isolation Forest

The Isolation Forest is an unsupervised algorithm that identifies anomalies by recursively partitioning data using random splits and isolating anomalies ((**inproceedings?**);Sohil, Sohali, and Shabbir (2022)). Unlike distance-based methods, it uses the fact that anomalies are few and different compared to normal data points. So, the number of splits (or path, $h(x)$) required for normal observations is higher than for anomalies that tend to isolate quickly. The anomaly score than is calculated: $s(x) = 2^{-\frac{h(x)}{E(h(x))}}$, where $E(h(x))$ is the expected path of a balanced tree. An anomaly score close to 1 indicates a high change of anomaly, while a score near 0.5 suggests normale observation.

This approach is computationally efficient and suitable for handling large datasets while maintaining high performance. It also overcomes the challenge of high dimensional data by using random subspace selection during each split, it reduces the dimensionality in a local space.

3.3 Training

To evaluate the robustness of methods and assess how consistently the model identifies anomalies, cross-validation is used. This process helps ensure that the model is not overfitting to specific patterns and can generalize well to unseen data, improving its reliability.

3.4 Method comparison

Model performance will be evaluated and compared on test dataset. The key metric for comparison will be F1-score:

$$F1 = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

As anomalies are rare, focusing on a metric that balances both precision and recall is crucial, as we want to correctly identify anomalies but at the same time not to disrupt production process from having lots of false positive.

4 Results

Table below presents the performance metrics for KNN and Isolation Forest. KNN significantly outperforms Isolation Forest across all metrics, with an accuracy of 0.93, precision of 0.92, recall of 0.92, and an F1-score of 0.95. This demonstrates that KNN was highly effective in identifying anomalies while maintaining a low rate of false positives. The high F1-score indicates a well-balanced performance, with both precision and recall being high, which is crucial in an anomaly detection context where it is important to correctly identify rare events without generating too many false alarms.

In contrast, Isolation Forest achieved much lower performance with an accuracy of 0.42, precision of 0.3, recall of 0.77, and F1-score of 0.43. Although the recall is relatively high, indicating that the model detected a significant portion of the anomalies, the low precision suggests that many normal points were incorrectly classified as anomalies. This results in a high number of false positives, as reflected in the relatively low F1-score.

Table 1: Performance metrics for different models.

Method	Accuracy	Precision	Recall	F1-Score
KNN	0.93	0.92	0.92	0.95
Isolation Forest	0.42	0.3	0.77	0.43

5 Conclusion

In this report, we have explored the problem of anomaly detection in the semiconductor manufacturing industry using a dataset consisting of sensor features from manufacturing equipment. The objective was to identify rare anomalies that could potentially lead to equipment failures. We applied K-Nearest Neighbors (KNN) and Isolation Forest, addressed challenges such as class imbalance using techniques like SMOTE for oversampling and robust scaling for feature standardization.

For this dataset, we have found that high-dimensional analysis did not improve over simpler methods. The KNN algorithm outperformed the Isolation Forest, achieving significantly higher F1-scores and better precision and recall values. This indicates that KNN, despite its limitations in high-dimensional spaces, was more effective for detecting anomalies in this case.

The Isolation Forest method struggled with the sparsity and complexity of the data and anomalies weren't easily separable. As this method is based on the assumption that anomalies will be isolated first in the process and anomalies weren't clearly separated, it failed to identify many of the anomalies. The scatter plot Figure 2 of the first vs. second components shows that the anomalies are not easily separable from normal points with significant overlap between the two classes.

Despite this overlap, KNN's approach was able to detect anomalies more effectively by leveraging the relative distances between points. This finding shows KNN works better in this case, as it is more used at identifying outliers in data where the anomalies are not well-separated. It is important to note that KNN can become computationally expensive and less effective with even higher-dimensional data. While KNN performed well in this scenario, we need to imply optimizations for other scenarios.

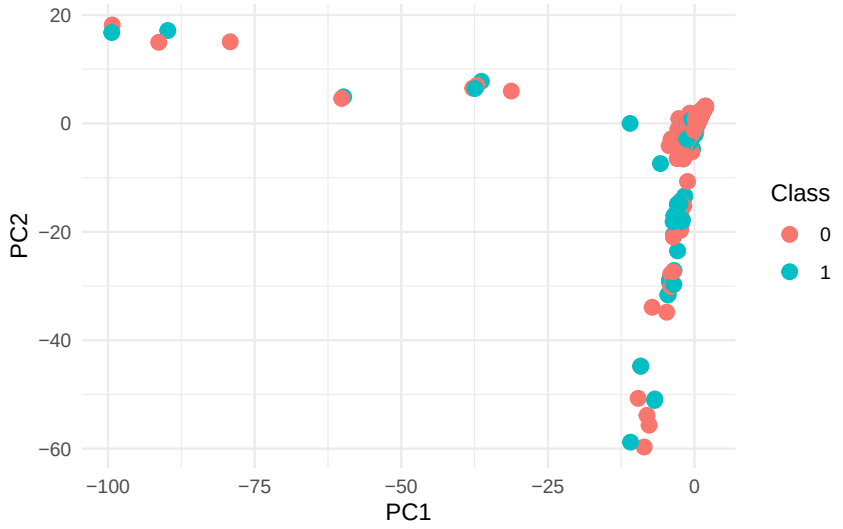

Figure 2: A bar-plot showing anomalies distribution by first and second principle components

6 References

Sohil, Fariha, Muhammad Umair Sohali, and Javid Shabbir. 2022. "An Introduction to Statistical Learning with Applications in R: By Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani, New York, Springer Science and Business Media, 2013, $41.98, eISBN: 978-1-4614-7137-7." *Statistical Theory and Related Fields* 6 (1): 87–87. https://doi.org/10.1080/24754269.2021.1980261.

Xu, Hongzuo, Guansong Pang, Yijie Wang, and Yongjun Wang. 2023. "Deep Isolation Forest for Anomaly Detection." *IEEE Transactions on Knowledge and Data Engineering* 35 (12): 12591–604. https://doi.org/10.1109/TKDE.2023.3270293.