Anomaly / Novelty Detection

MATH-412 - Project

École Polytechnique Fédérale de Lausanne (EPFL)

Kalil Bouhadra
kalil.bouhadra@epfl.ch

Gabriel Marival
gabriel.marival@epfl.ch

Georg Khella
georg.khella@epfl.ch

January 14, 2025

Contents

1 Introduction

In today's data-driven world, unusual patterns or outliers in data can signal critical events such as system failures or fraudulent activities. Being able to detect such anomalies is or primary importance in many contexts. This analysis delves into the domain of anomaly and novelty detection, exploring how machine learning techniques can identify rare, significant deviations from normal behavior, which may otherwise go unnoticed in large volumes of routine data. Both anomaly and novelty detection can be framed as binary classification tasks, more precisely as one-class classification problems (Khan and Madden 2014). In these problems, the goal is to assign each data point to one of two categories: normal or anomalous, while knowing that the normal class is far more represented than the anomalous one. The challenge lies in the nature of the training data. In novelty detection, all training data are labeled as normal, so the learned model must identify deviations from this normal behavior in new datasets. In anomaly detection, the data are unlabeled and contain only a few anomalous instances, requiring the model to detect significant departures from normal patterns without explicit guidance. While each algorithm has its own strengths and limitations, they share the same objective: to determine the most effective way to distinguish between normal and anomalous points, whether by finding an optimal decision boundary, threshold, or employing alternative strategies.

Specifically, the work aims to evaluate how effectively the One-Class Support Vector Machine (OSVM) can be applied under varying data constraints, performing well in novelty detection with entirely normal labeled data, and exploring its limitations in anomaly detection with unlabeled data. The study also examines the impact of key hyperparameters (e.g., kernel choice, ν, γ). In response to potential weaknesses in high-dimeensional context, the analysis investigates the use of dimensionality reduction techniques like Principal Component Analysis (PCA) to mitigate these issues, and tests alternative methods, such as Isolation Forest, for anomaly detection. In order to do this, Section 2 details the methodologies used, including the theoretical foundation of the OSVM, an explanation of relevant hyperparameters, and a discussion of supplementary techniques such as PCA and Isolation Forest. Section 3 presents the experimental setup, describing the datasets, evaluation protocols, and the process for tuning hyperparameters. It also provides an interpretation of the results, including comparative analyses of different approaches using suitable metrics. Finally, Section 4 summarizes the main findings, discusses conclusions drawn from the research, and suggests directions for future work.

2 Methods

Let us suppose for now that we are in a novelty detection context. Consider a dataset $D = \{(x_1, y_1), \ldots, (x_n, y_n)\}$, where $x_i \in \mathbb{R}^d$ and $y_i \in \{-1, 1\}$ and all labels y_i are 1 (normal points). The goal is to construct a decision function $f(x)$ (Equation 2.1) able to distinguish efficiently between normal and abnormal points in a new dataset.

$$f(x) = \begin{cases} 1 & \text{if } x \text{ is normal,} \\ -1 & \text{if } x \text{ is anomalous.} \end{cases} \tag{2.1}$$

While simple hyperplane or hypersphere boundaries may serve as optimal decision functions for certain data structures, they can be insufficient for separating anomalies from normal points in more complex spaces. To address this, a common approach is to introduce a mapping $\phi : \mathbb{R}^d \to \mathcal{F}$, which projects points x from the original space into a higher-dimensional feature space $\mathcal{F}$. In this transformed space, we aim to find a hyperplane defined by $w \cdot \phi(x) - \rho = 0$, where w is the normal vector to the hyperplane and ρ is the threshold (intercept). By introducing some additional parameters to increase flexibility, we get to the primal formulation of the One-Class SVM problem (Schölkopf et al. 1999):

$$\begin{aligned} \min_{w,\rho,\{\psi_i\}} \quad & \frac{1}{2}\|w\|^2 + \frac{1}{\nu n}\sum_{i=1}^{n}\psi_i - \rho, \\ \text{s.t.} \quad & w \cdot \phi(x_i) \geq \rho - \psi_i, \quad i = 1, \ldots, n, \\ & \psi_i \geq 0, \quad i = 1, \ldots, n. \end{aligned} \tag{2.2}$$

Here, ψ_i are slack variables that allow some points to violate the constraint (i.e., lie on the wrong side of the hyperplane), and $\nu \in (0, 1)$ is a hyperparameter that controls the trade-off between the volume of the region enclosing normal data and the number of allowed anomalies. A higher ν allows more points to be excluded as anomalies, whereas a lower ν forces the model to consider more points as normal. The threshold ρ adjusts the position of the hyperplane to control the tightness of the boundary around normal data. The objective function minimizes $\|w\|^2$ (maximizing the margin) while penalizing constraint violations through ψ_i and ν.

The decision function derived by the optimization Problem 2.2 is the following:

$$f(x) = \text{sign}(w \cdot \phi(x) - \rho), \tag{2.3}$$

which assigns 1 to points on one side of the decision boundary (considered normal) and -1 to points on the other side (considered anomalies).

To avoid explicitly computing the mapping $\phi(x)$, it can be useful to derive and solve the dual problem instead. Since the primal problem is convex and Slater's conditions are easily satisfied in the primal problem (Equation 2.2), strong duality holds, ensuring that solving the dual problem yields the same solution as the primal one. Moreover, by using the kernel trick $K(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle$, the dual problem (Equation 2.4) becomes independent of the explicit mapping $\phi(x)$ (Schölkopf et al. 1999). Therefore, a suitable kernel function can be chosen to capture the potential complex structure of the data.

$$\begin{aligned} \max_{\alpha} \quad & -\frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j\, K(x_i, x_j), \\ \text{s.t.} \quad & 0 \leq \alpha_i \leq \frac{1}{\nu n}, \quad i = 1, \ldots, n, \\ & \sum_{i=1}^{n}\alpha_i = 1, \end{aligned} \tag{2.4}$$

Once the dual problem is solved, the decision function for a new point x can be expressed as:

$$f(x) = \text{sign}\left(\sum_{i=1}^{n}\alpha_i\, K(x_i, x) - \rho\right). \tag{2.5}$$

We usually have two hyperparameters to tune in this context: ν, which governs the fraction of points allowed to violate the constraints, and kernel-related parameters, such as γ for a Gaussian (RBF) kernel, which determine the influence of nearby versus distant points. Other parameters are directly optimized by the dual convex optimization problem (Equation 2.4).

The OSVM algorithm is particularly well-suited for novelty detection. Indeed, due to its inherent structure, it can effectively learn a decision boundary that encloses the normal data distribution while maintaining enough flexibility to detect whether a new point is abnormal. However, in the context of anomaly detection, where the data is unlabeled and some anomalies are present, it may struggle to learn a suitable decision function. This issue is further exacerbated in high-dimensional spaces, where the complexity of the data and the curse of dimensionality can weaken the model's ability to accurately capture normal behavior. To mitigate the impact of high dimensionality on the efficiency of anomaly detection algorithms, PCA can be applied (Guillaume Obozinski 2024). This method addresses the curse of dimensionality by projecting the data onto a lower-dimensional subspace. It computes the eigenvalues and eigenvectors of the covariance matrix, selects the top eigenvectors corresponding to the largest eigenvalues,

and projects the data in these directions. This lead to reduced noise, and usually improved computational efficiency as it simplify the process of separation of the of data points. This is particularly beneficial for OSVM, which performs better in lower-dimensional spaces, especially in anomaly detection task. Other algorithms may outperform OSVM in the context of anomaly detection (such as Autoencoders, kNN, etc.), and we would like to highlight one in particular: Isolation Forest. This method isolates observations by recursively selecting features and random split values, partitioning the data until anomalies are isolated in fewer splits compared to normal points (Guillaume Obozinski 2024). Since anomalies are sparse and distinct, they are easier to isolate. An anomaly score is computed based on the number of splits required to isolate a point and the fewer the splits, the more likely the point is an anomaly. Isolation Forest is computationally efficient, scalable to large datasets, and robust in high-dimensional spaces, making it a strong candidate for anomaly detection.

3 Experiments

The dataset consists of breast cancer data, directly sourced from the Python scikit-learn library. It includes 212 normal instances and 20 anomalies. For the novelty detection task, only the 212 normal instances will be treated as labeled, while the full dataset will be considered unlabeled for the anomaly detection task. For evaluating our models in the novelty detection context, we rely on standard metrics from binary classification, such as precision, recall, F_1-score, and the ROC curve. These metrics are computed using confusion matrices, which provide detailed insight into classification performance. In the case of novelty detection, where labeled data is available, these metrics are directly applicable in the training part and highly informative. Consequently, in the subsequent analysis, we will primarily use confusion matrices to assess and compare model performance. A more detailed discussion of these metrics is provided in Appendix A.

First, we compared anomaly detection and novelty detection using OSVM, employing basic binary classification metrics which serve as a baseline for our analysis and allow us to compare the two tasks. As shown in Figure 1 in Appendix B, OSVM performs significantly better on novelty detection than on anomaly detection across all four metrics, particularly recall and F1-score. For anomaly detection, OSVM exhibits a very low recall and F1-score, indicating that many anomalies are missed. This result is expected, as the novelty detection scenario benefits from labeled training data, whereas anomaly detection must deal with unlabeled data, making it inherently more challenging. Nonetheless, the baseline comparison highlights the need for more advanced methods to improve anomaly detection performance.

Below, we discuss the performance of OSVM under varying hyperparameter settings. References to the confusion matrices are provided in Figures 2, 4, and the kernel comparison in Figure 4 (Appendix C). From these matrices, we can derive several qualitative insights into the impact of OSVM hyperparameters on novelty detection performance. In OSVM, some internal parameters, such as the Lagrange multipliers α_i and the threshold ρ, are directly optimized by the convex problem during training. In contrast, the parameters ν, γ, and the choice of kernel must be specified by the user: ν controls the trade-off between maximizing the margin and allowing errors in the training data. A higher ν tightens the decision boundary, reduces regularization, and thus makes it easier to classify points as anomalies. In the confusion matrices (Appendix C, Figure 2), a higher value of ν shows an increase in the number of predicted outliers. However, for very high values of ν, the precision on normal points drops significantly, as the model becomes too permissive. The parameter γ commes from the RBF kernel : $K(x_i, x_j) = \exp(-\gamma\|x_i - x_j\|^2)$, it controls the influence of distant points. A low γ results in a smoother decision boundary (potentially underfitting), while a high γ leads to a more complex boundary (potentially overfitting), as shown in Appendix C, Figure 4. The choice of the kernel also have an important impact on OSVM performances. In Appendix C, the confusion matrices in Figure 4 (last row) illustrate how using different kernels impacts performance.

One natural question is how to effectively tune hyperparameters. In novelty detection, since all training examples are normal, we can leverage classic model selection techniques using labeled data. With labels

available, grid search allows us to systematically explore hyperparameter combinations (e.g., various ν and γ values, or different kernels). By splitting the data into training and validation sets, we train models with different settings and select the hyperparameters that yield the best performance (measured by recall, precision, or F_1-score) on the validation set. However, in anomaly detection tasks, tuning via grid search is not feasible, as we are dealing with unlabeled data. Alternative approaches for parameter tuning include generating synthetic data, leveraging the statistical properties of the data (such as variance and distribution), or analyzing the stability of the resulting model. An example of a stability analysis is shown in Figure 5 in Appendix D. Intuitively, we seek models where the ratio of anomalies is close to 1, as this indicates that the predicted proportion of anomalies aligns well with the dataset's distribution. From the plot, this suggests selecting high values of ν, specifically greater than 0.36, which yields a ratio close to 0.9. However, it is important to note that such high values of ν may lead to reduced precision for normal points, as observed in previous experiments.

To enhance OSVM performance in anomaly detection, we applied Principal Component Analysis (PCA) for dimensionality reduction. PCA projects the data onto components that capture the most variance, helping reduce noise. As shown in Figure 6 in Appendix E, the first few components capture over 90% of the variance, indicating that most of the information is preserved with fewer dimensions. The confusion matrices in Figure 7 in Appendix F compare OSVM before and after PCA, as well as Isolation Forest. Without PCA, OSVM achieves 74.5% precision for normal points but only 14.33% recall for anomalies, suggesting that noise impacts its performance. After PCA, precision remains stable (73.67%), and recall improves slightly to 15.00%, indicating that dimensionality reduction helps reduce overfitting. Isolation Forest achieves 73.83% precision for normal points and 16.33% recall for anomalies, outperforming OSVM in detecting outliers. This method isolates anomalies more effectively due to its tree-based structure, making it robust against noise and complex data.

4 Conclusion

In this work, we have examined the One-Class SVM (OSVM) under two distinct but closely related frameworks: novelty detection (with labeled normal data) and anomaly detection (with largely unlabeled data). Our experiments show that OSVM excels in novelty detection but struggles in the anomaly detection scenario due to the absence of labeled anomalies. We explored the influence of key parameters such as ν and γ, as well as the choice of kernel, with the Gaussian kernel demonstrating the most adaptability. Furthermore, in high-dimensional settings where OSVM's performance declines, applying PCA often proved beneficial, mitigating noise and the curse of dimensionality. Nevertheless, while PCA-enhanced OSVM can help address some challenges in anomaly detection, Isolation Forest emerged as a stronger alternative, achieving higher recall and isolating anomalies more efficiently in fewer splits. Consequently, despite OSVM's solid performance in novelty detection, specialized methods like Isolation Forest may be more effective in real-world anomaly detection tasks. In future work, exploring more sophisticated deep-learning architectures such as autoencoders for feature extraction could further strengthen detection capabilities, particularly in complex, high-dimensional datasets. Combining OSVM with these advanced techniques, or integrating multiple algorithms into an ensemble, may offer enhanced robustness and flexibility for detecting rare yet critical outliers in various application domains.

A Detailed Discussion of Metrics

As both anomaly and novelty detection can be framed as binary classification tasks, a naive metric is the 0-1 loss:

$$L(y, f(x)) = \mathbf{1}_{\{f(x) \neq y\}},$$

which simply measures the overall misclassification rate. However, in imbalanced settings—such as detecting rare anomalies—this loss is not informative. For instance, a classifier always predicting the majority class ("normal") would achieve a low loss despite failing to detect anomalies.

More informative metrics from binary classification include precision, recall, and the F_1-score:

$$\text{Precision} = \frac{TP}{TP + FP}, \qquad \text{Recall} = \frac{TP}{TP + FN}, \qquad F_1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}},$$

where TP, FP, FN, and TN denote true positives, false positives, false negatives, and true negatives respectively. These can be summarized in the confusion matrix:

$$\begin{pmatrix} TN & FP \\ FN & TP \end{pmatrix}.$$

We will mainly use confusion matrices to evaluate our models in the novelty detection context, where labeled data allows computation of these metrics.

Additionally, the ROC curve—which plots the true positive rate against the false positive rate

$$TPR = \frac{TP}{TP + FN}, \qquad FPR = \frac{FP}{FP + TN},$$

for various thresholds—and its AUC offer a threshold-independent evaluation of classifier performance.

Note that these metrics are directly applicable in novelty detection due to the availability of labels, while anomaly detection without labels requires alternative evaluation approaches if we want to tune our parameters based on that.

B Comparison of Performance Between Novelty and Anomaly Detection

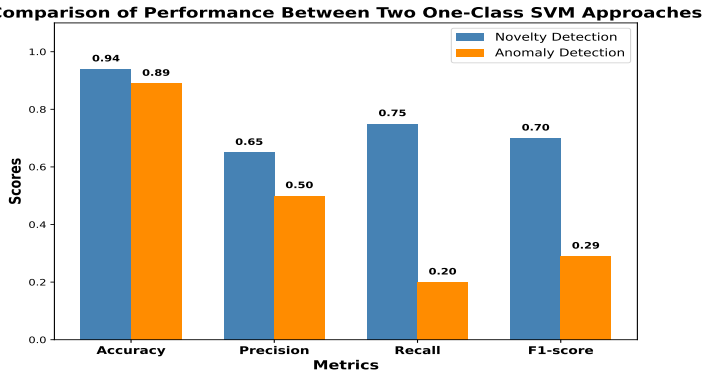
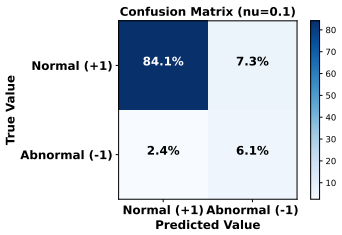


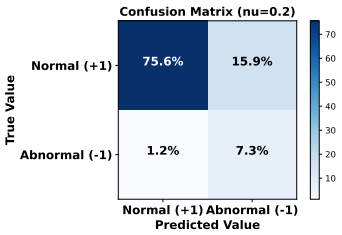
Figure 1: Comparison of OSVM performance on novelty detection versus anomaly detection across four metrics: accuracy, precision, recall, and F1-score. Novelty detection performs better for all the metrics.

C Confusion Matrices in novelty detection

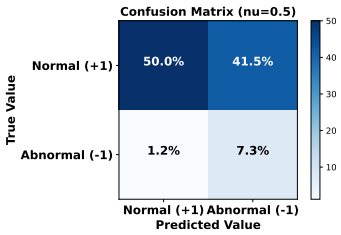
C.1 Impact of ν (with fixed $\gamma = 0.01$)



(a) Confusion Matrix with $\nu = 0.1$.



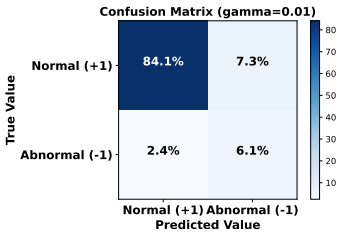
(b) Confusion Matrix with $\nu = 0.2$.



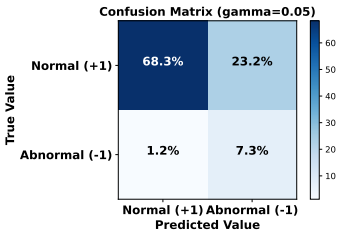
(c) Confusion Matrix with $\nu = 0.5$.

Figure 2: Confusion matrices of OSVM applied to novelty detection tasks. The plot shows the impact of the parameter ν on the performance of OSVM, with $\gamma = 0.01$.

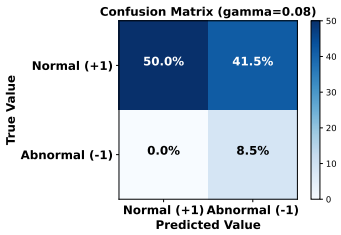
C.2 Impact of γ (with fixed $\nu = 0.1$)



(a) Confusion Matrix with $\gamma = 0.01$.



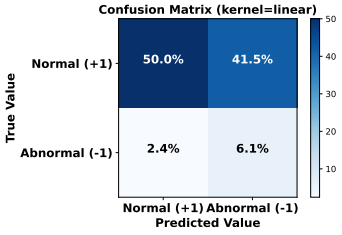
(b) Confusion Matrix with $\gamma = 0.05$.



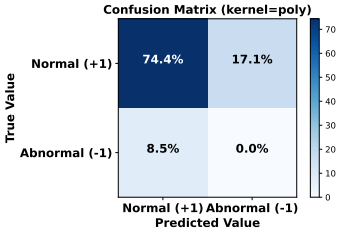
(c) Confusion Matrix with $\gamma = 0.08$.

Figure 3: Confusion matrices of OSVM applied to novelty detection tasks. The plot shows the impact of the parameter γ on the performance of OSVM, with $\nu = 0.1$.

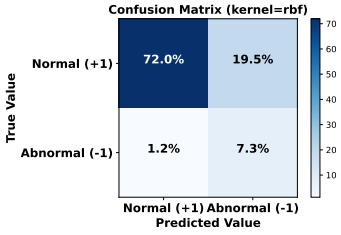
C.3 Impact of kernel (with fixed $\nu = 0.1$)



(a) Confusion Matrix for linear kernel



(b) Confusion Matrix for Polynomial kernel



(c) Confusion Matrix for Gaussian kernel

Figure 4: Confusion matrices of OSVM applied to novelty detection tasks. The plot shows the impact of the kernel on the performance of OSVM

D Impact of ν on the Proportion of Anomalies (Train vs Test)

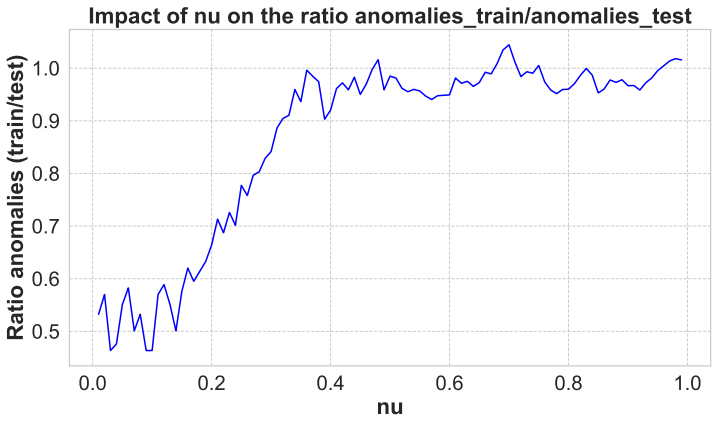


Figure 5: Impact of the parameter ν on the ratio of anomalies detected in the training and test sets. A ratio close to 1 indicates consistent anomaly detection performance across both sets.

E Cumulative Variance for PCA

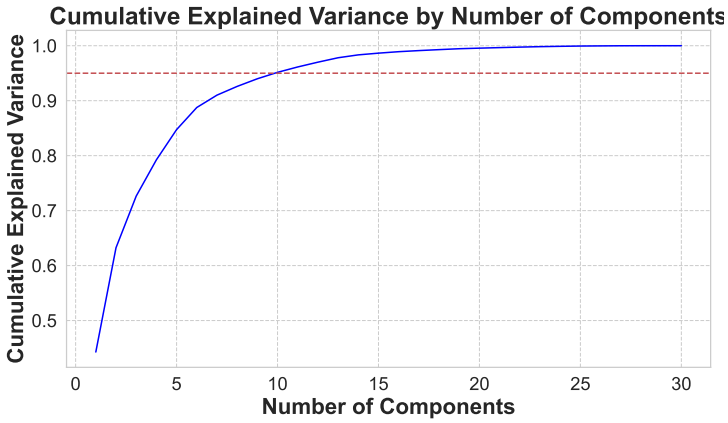
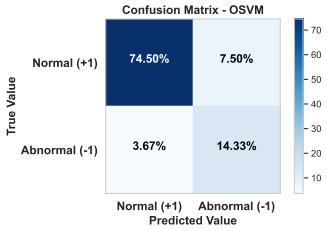
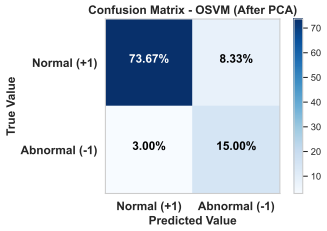


Figure 6: Cumulative explained variance by the number of components. The first few components capture more than 90% of the variance, demonstrating that dimensionality reduction preserves most of the information in the data.

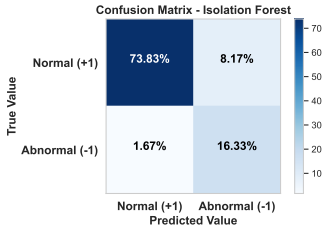
F Comparison of Confusion Matrices: OSVM with/without PCA, and Isolation Forest



(a) OSVM (Without PCA): 74.5% precision and 14.33% recall for anomalies.



(b) OSVM (After PCA): 73.67% precision and 15.00% recall for anomalies.



(c) Isolation Forest: 73.83% precision and 16.33% recall for anomalies.

Figure 7: Confusion matrices comparing OSVM with and without PCA, and Isolation Forest. PCA helps reduce noise and slightly improves recall. Isolation Forest provides higher recall for anomalies.

References

Guillaume Obozinski, Yoav Zemel (2024). *MATH-412 - Statistical Machine Learning*.

Khan, Shehroz S and Michael G Madden (2014). "One-class classification: taxonomy of study and review of techniques". In: *The Knowledge Engineering Review* 29.3, pp. 345–374.

Schölkopf, Bernhard et al. (1999). "Support vector method for novelty detection". In: *Advances in neural information processing systems* 12.