Fintech course - A.Y. 2024/2025

Group 10

EARLY WARNING SYSTEM

Final Project - Business Case 4

Accoto Davide, Allegrini Francesco, Augenti Donato, Busci Alberto, Catina Andrea

AGENDA

INTRODUCTION

An Early Warning System (EWS) is a powerful framework that helps identify **emerging risks** or **hazards** before they escalate into full-blown crises. By continuously monitoring key indicators and analyzing data in **real time**, EWS provides timely and actionable alerts that empower decision-makers to act swiftly and effectively. Whether applied in disaster management, finance, health, or security, these systems play a critical role in minimizing damage, saving lives, and safeguarding assets. Ultimately, Early Warning Systems transform uncertainty into preparedness, turning potential threats into opportunities for resilience and **strategic response**.

OUR GOALS

- Build Machine Learning (ML) models to classify **risk-on** (normal) vs. **risk-off** (abnormal) market conditions in real time.

- Develop a framework that **integrates unsupervised and supervised learning** approaches to enhance financial risk management: by anticipating abnormal market conditions, we can implement appropriate **hedging strategies**.

- Utilize anomaly detection techniques, particularly **novelty detection**, to identify emerging risk-off regimes, including those not observed in the training data. When accurately predicted, these regimes can be leveraged for speculative purposes by investing in **assets** that typically appreciate during abnormal market conditions.

OUR SOLUTIONS

SUPERVISED LEARNING

- Optimized Random Forest
- Optimized Support Vector Machine (SVM)
- XGBoost

DEEP LEARNING

- Autoencoder
- Variational Autoencoder

UNSUPERVISED LEARNING

- Optimized Isolation Forest
- One Class Support Vector Machine
- Local Outlier Factor
- Gaussian Mixture (GMM)
- COPOD

ENSEMBLED METHODS

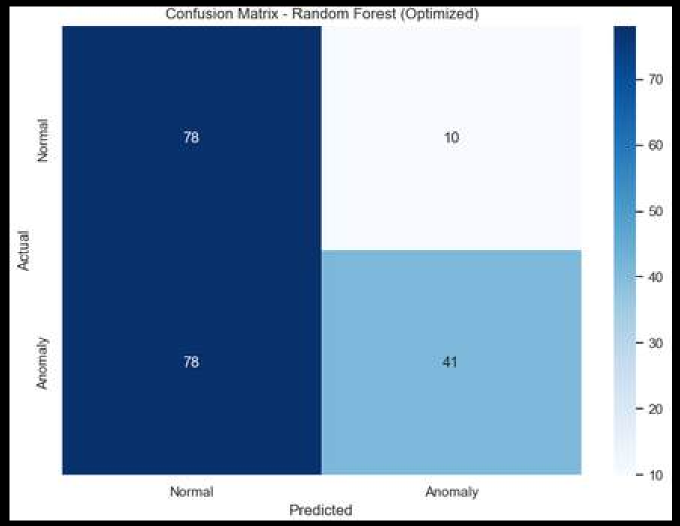
- GMM + XGBoost
- GMM + SVM + XGBoost
- GMM + SVM + COPOD + XGBoost

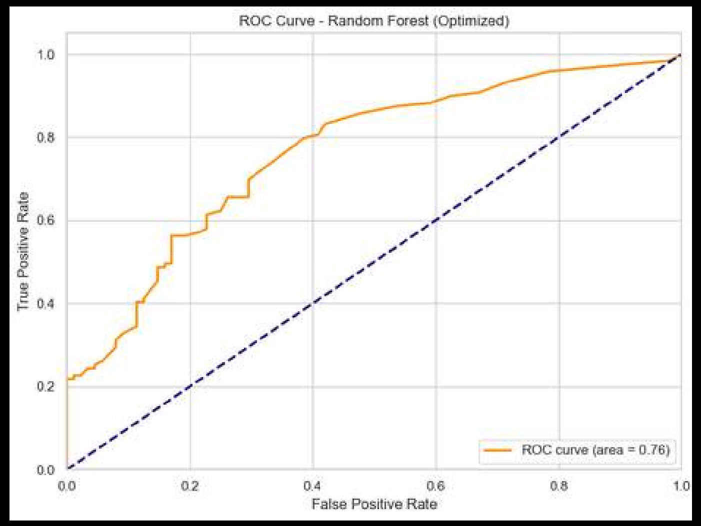
OPTIMIZED RANDOM FOREST

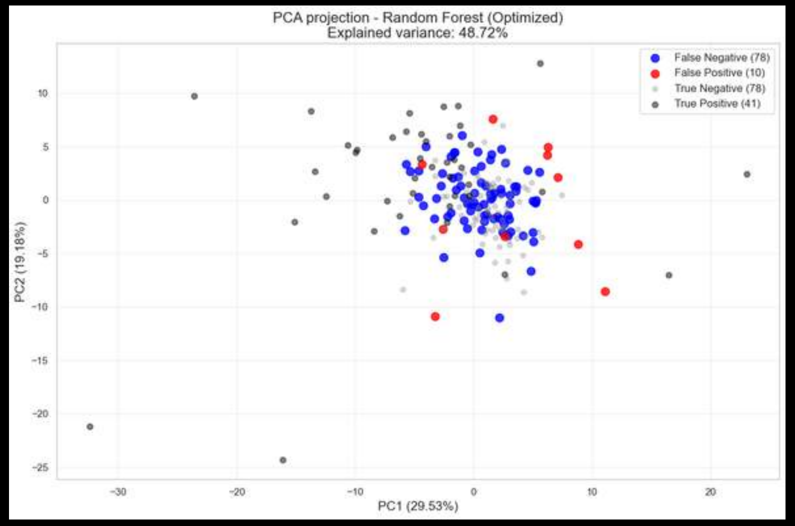
Main metrics

- **Precision: 0.8039**
- **Recall: 0.3445**
- **F1 Score: 0.4824**

Confusion Matrix, ROC Curve and PCA Projection







Learning from Anomalies in an Imbalanced Setting

We train a supervised Random Forest classifier by combining **non-anomalous training data** with **mixed cross-validation data** containing both normal and anomalous instances. This enables the model to learn from anomalies despite their absence in the original training set. The trained model predicts on unseen test data, providing both **binary classifications** and **anomaly probability scores** for comprehensive evaluation.

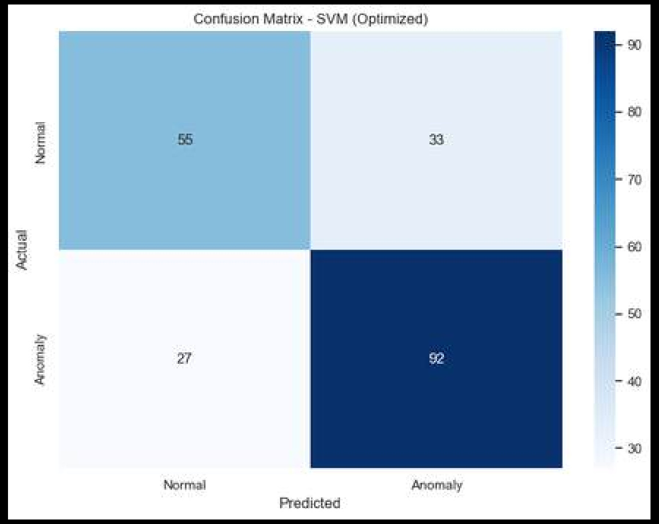
To improve performance in an imbalanced setting, we tune the Random Forest by **adjusting class weights** to maximize the **F1 score**. Assigning higher penalties to the majority class enhances recall without severely compromising precision. The best result is achieved with a class weight ratio of **1:6**, reaching an F1 score of 0.4824 — making the model more effective for early warning scenarios.

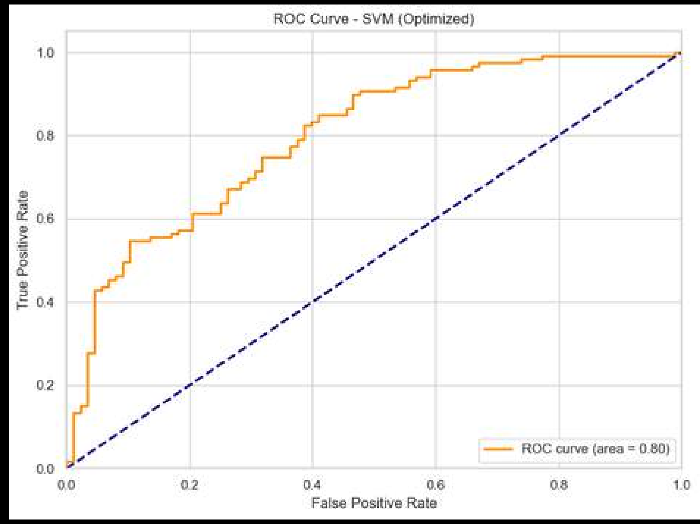
OPTIMIZED SVM

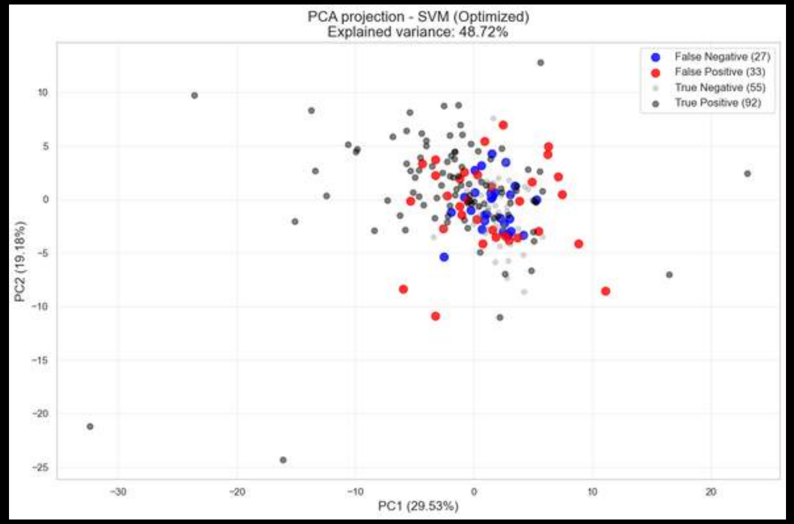
Main metrics

- **Precision: 0.7360**
- **Recall: 0.7731**
- **F1 Score: 0.7541**

Confusion Matrix, ROC Curve and PCA Projection







SVM Performance Boost through F1 Optimization

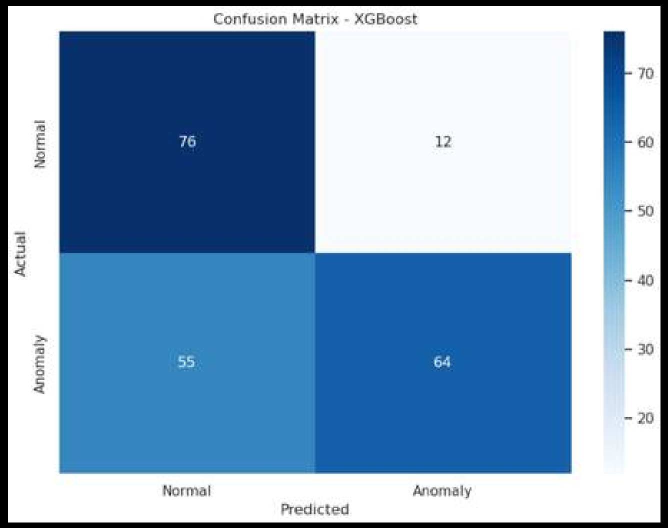
We train a Support Vector Machine (SVM) with an **RBF kernel**, using balanced class weights to address class imbalance. The model is **fine-tuned through hyperparameter** (*C* and *gamma*) and threshold optimization to maximize the **F1 score**. This process significantly **improves recall and F1**, while **precision decreases slightly** but remains within an acceptable range. Final predictions are evaluated with standard metrics and visualized using **PCA** to assess model separation and effectiveness.

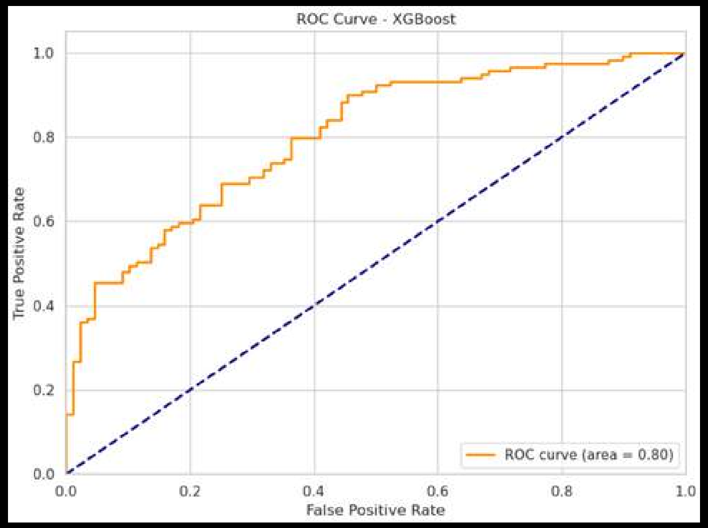
XGBOOST

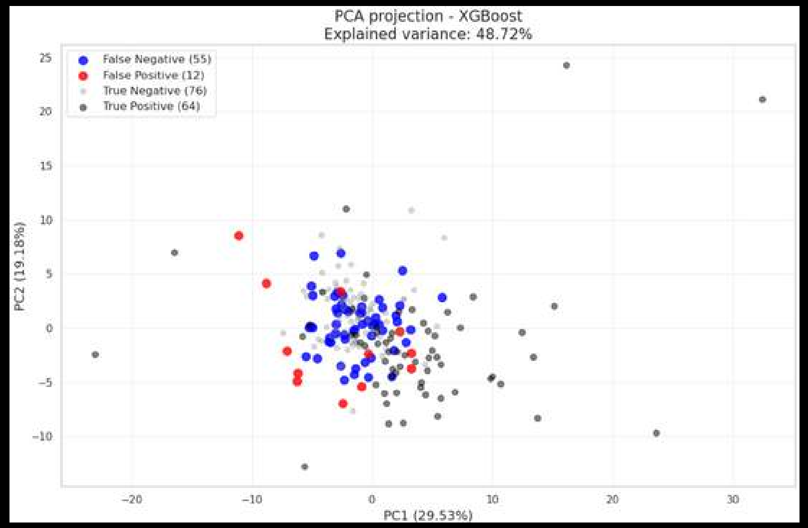
Main metrics

- **Precision: 0.8421**
- **Recall: 0.5378**
- **F1 Score: 0.6564**

Confusion Matrix, ROC Curve and PCA Projection







XGBOOST

We train an XGBoost classifier to detect anomalies in a time-series setting, using a temporal train-test split to preserve data chronology. The model is **optimized with a log-loss objective** and handles class imbalance effectively. After training, it predicts anomalies on the test set, and its performance is evaluated using precision, recall, F1-score, and a confusion matrix. Feature importance analysis is also conducted to assess the model's decision basis and detect potential data leakage

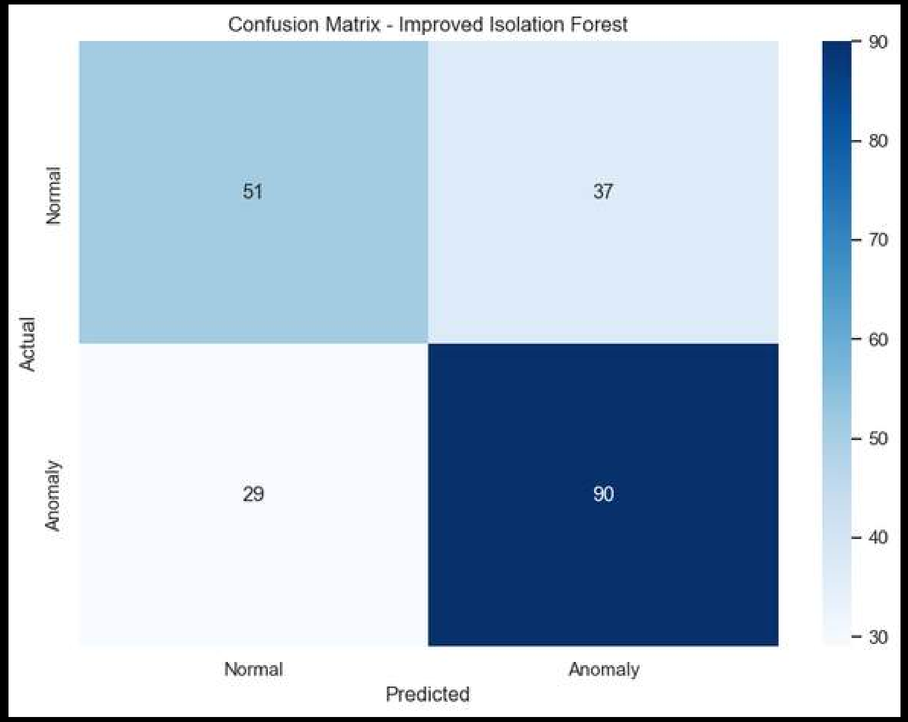
OPTIMIZED ISOLATION FOREST

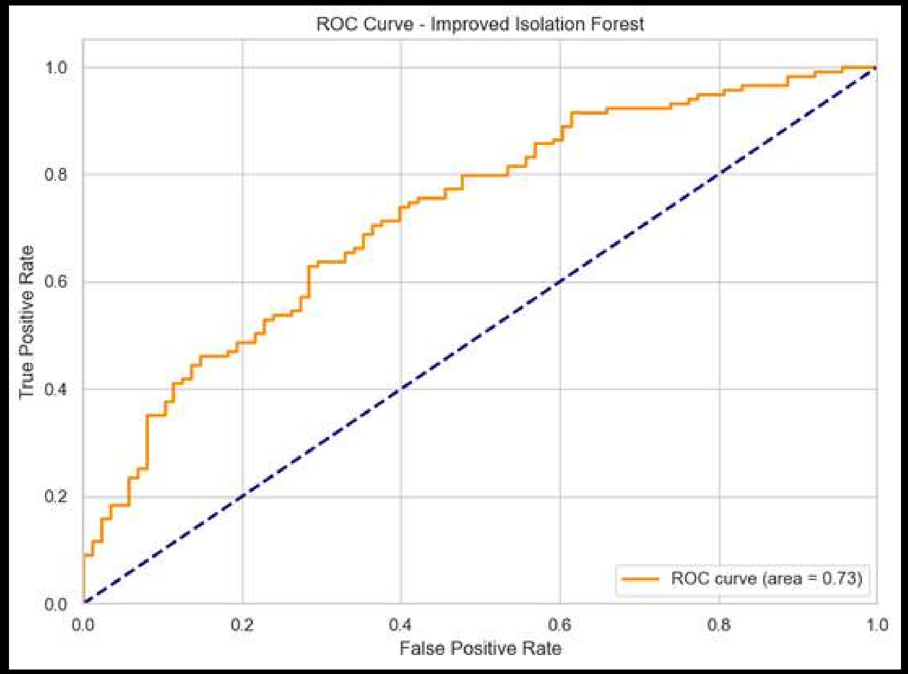
We train an Isolation Forest model, specifically designed for unsupervised anomaly detection, which isolates anomalies based on the **number of splits** required to separate observations. The model is fitted on the training data and assigns anomaly scores to the test set, where lower scores indicate higher likelihood of being an outlier. Predictions are obtained by applying a decision threshold to these scores. The model's performance is then evaluated using **precision, recall, F1 score, and ROC analysis**, followed by a PCA-based visualization to illustrate the distribution and separability of predicted anomalies in a reduced feature space.

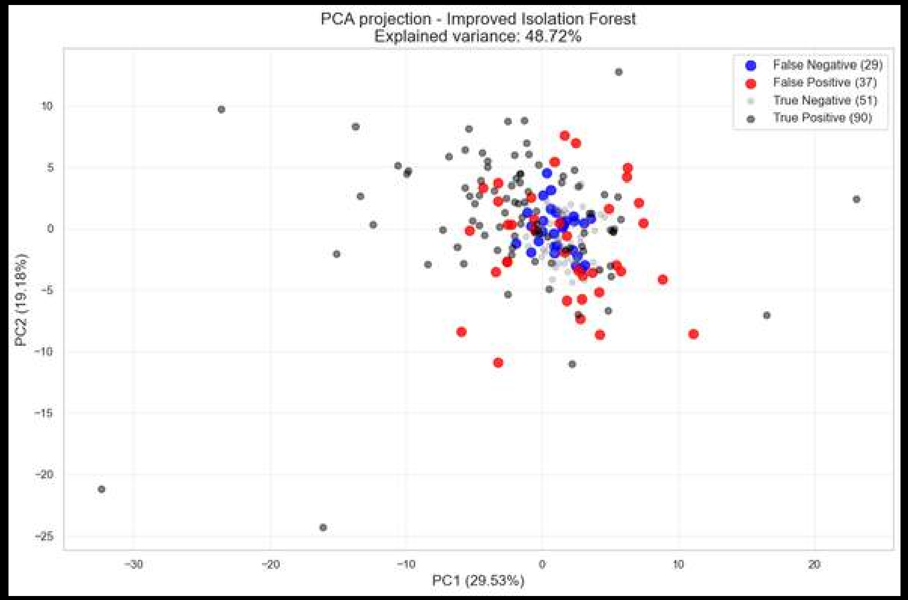
Precision: 0.7087

Recall: 0.7563

F1 Score: 0.7317







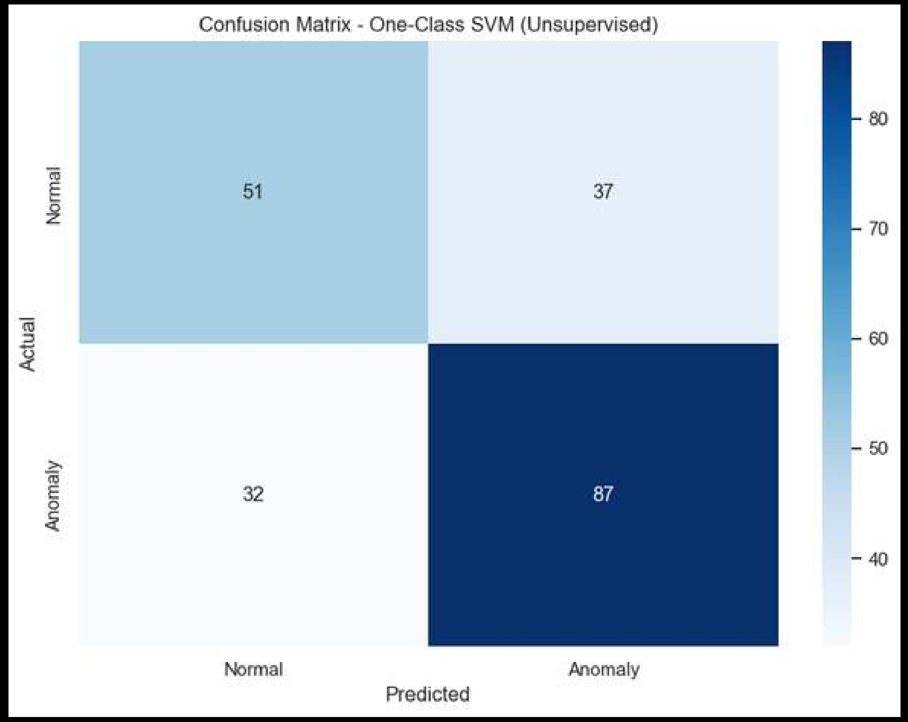
ONE CLASS SVM

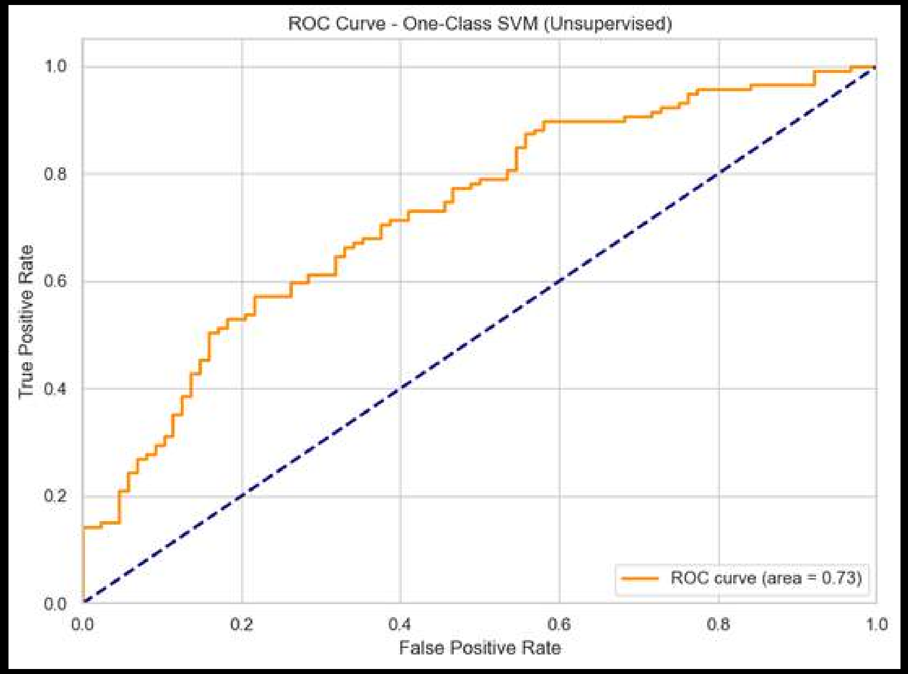
We train a One-Class SVM with an **RBF kernel** to detect anomalies, assuming that normal instances dominate the training data. The model learns the boundary of the normal class and is then applied to a mixed test set to identify deviations. Anomaly scores are used for evaluation through **classification metrics and ROC curves**, supported by PCA visualizations to illustrate the separation between normal and anomalous instances.

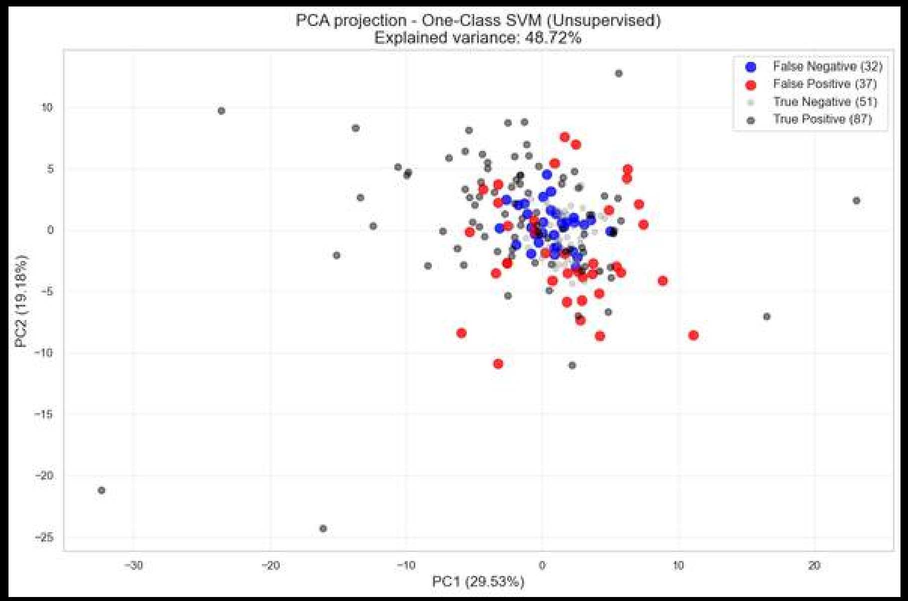
Precision: 0.7016

Recall: 0.7311

F1 Score: 0.7160







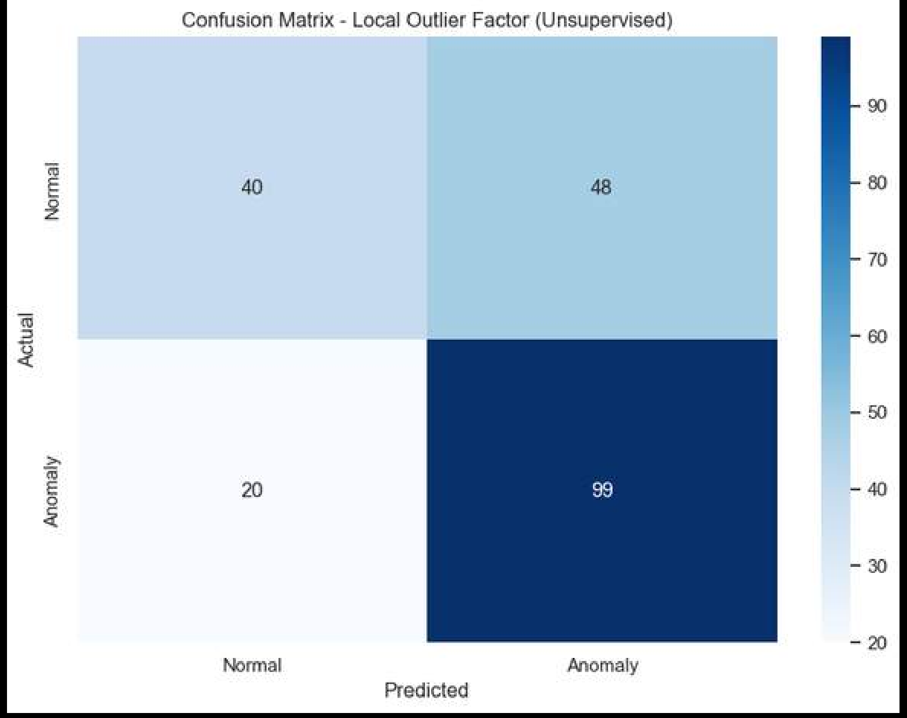
LOCAL OUTLIER FACTOR

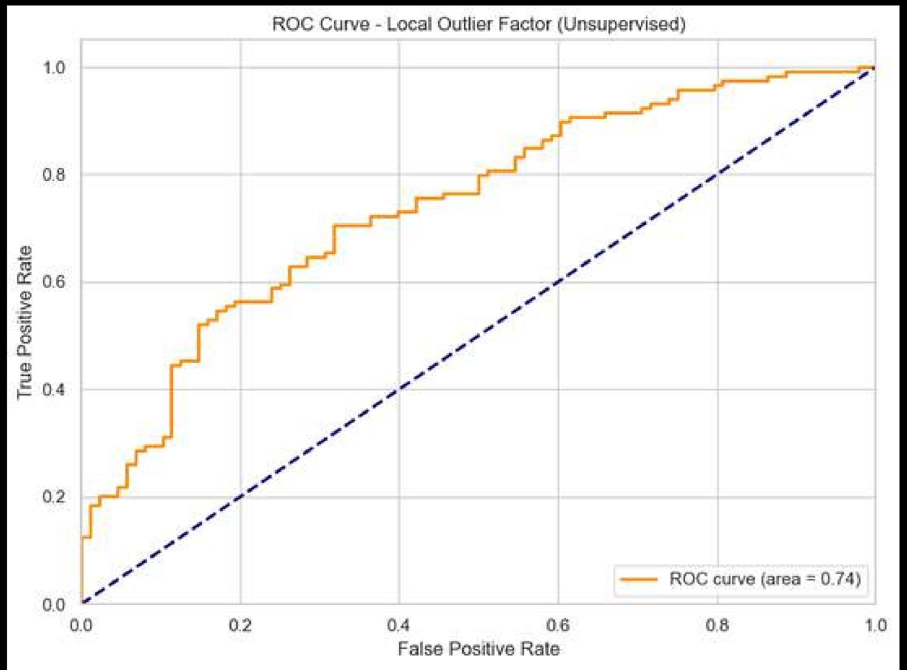
We apply the Local Outlier Factor (LOF) algorithm to identify anomalies based on local density deviations. The model compares each data point's density with that of its neighbors, flagging points with **significantly lower density** as anomalies. Anomaly scores are computed and used for performance evaluation using **classification metrics and ROC curves**, while PCA visualizations help illustrate the spatial separation of detected anomalies from normal observations.

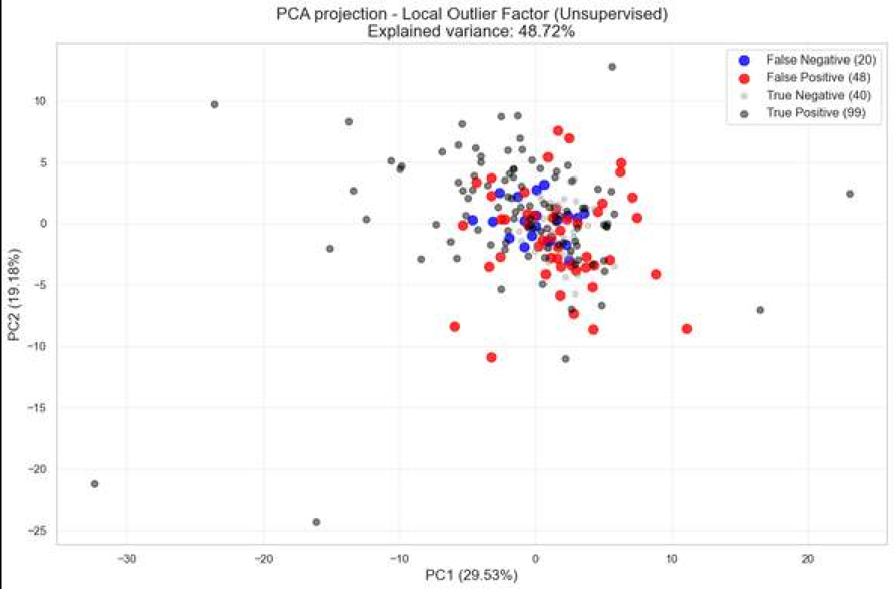
Precision: 0.6735

Recall: 0.8319

F1 Score: 0.7444







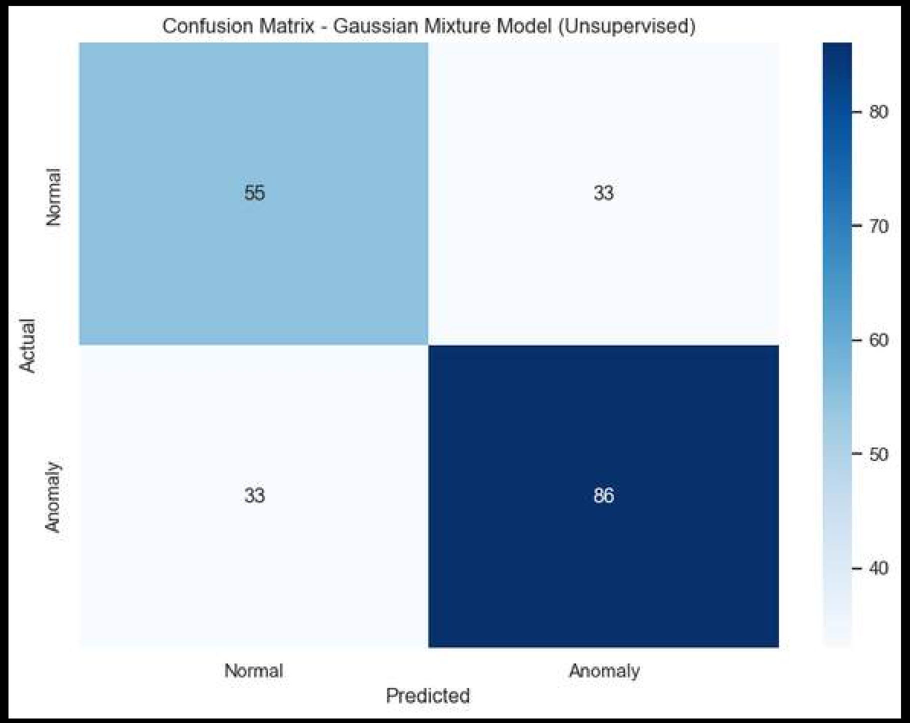
GAUSSIAN MIXTURE

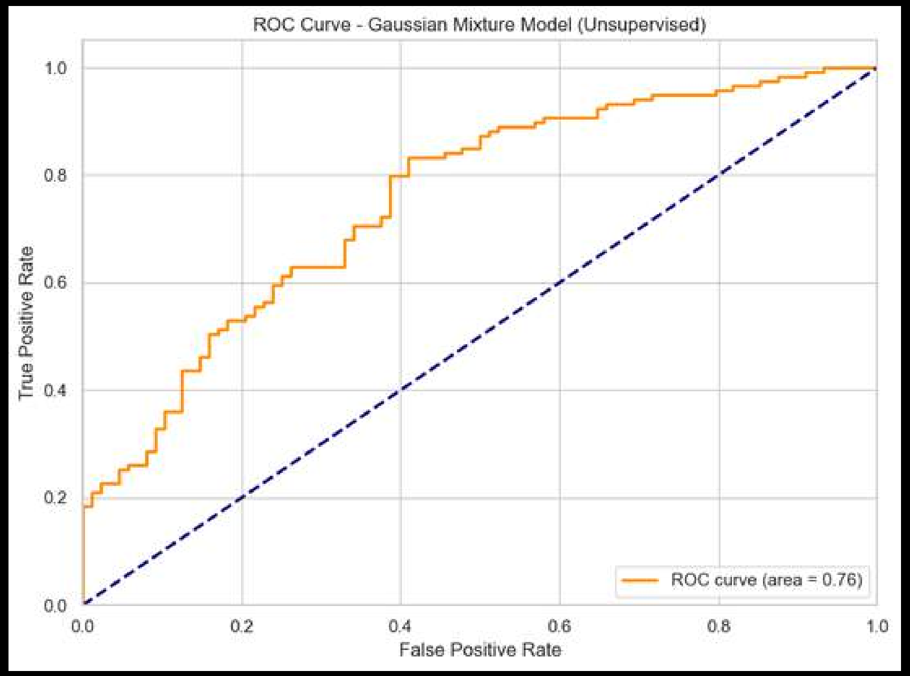
We use a Gaussian (Normal) model for anomaly detection by assuming the data follows a **multivariate normal distribution**. The model estimates the mean and covariance from the training data and computes the likelihood of each test sample under this distribution. Points with low likelihood values are flagged as anomalies. Performance is evaluated through standard **classification metrics and ROC analysis**, while PCA-based visualizations reveal the spatial distribution of outliers relative to the Gaussian contours.

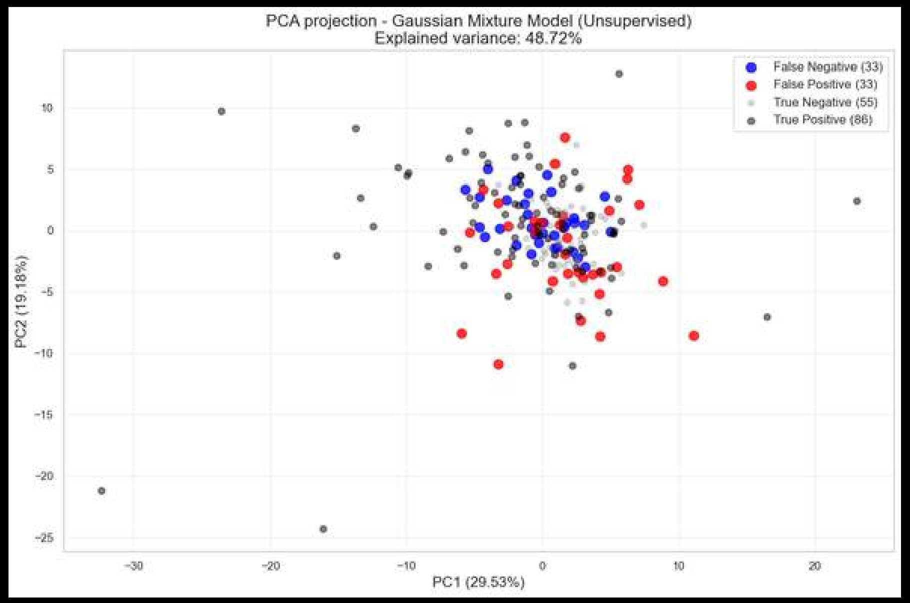
Precision: 0.7227

Recall: 0.7227

F1 Score: 0.7227







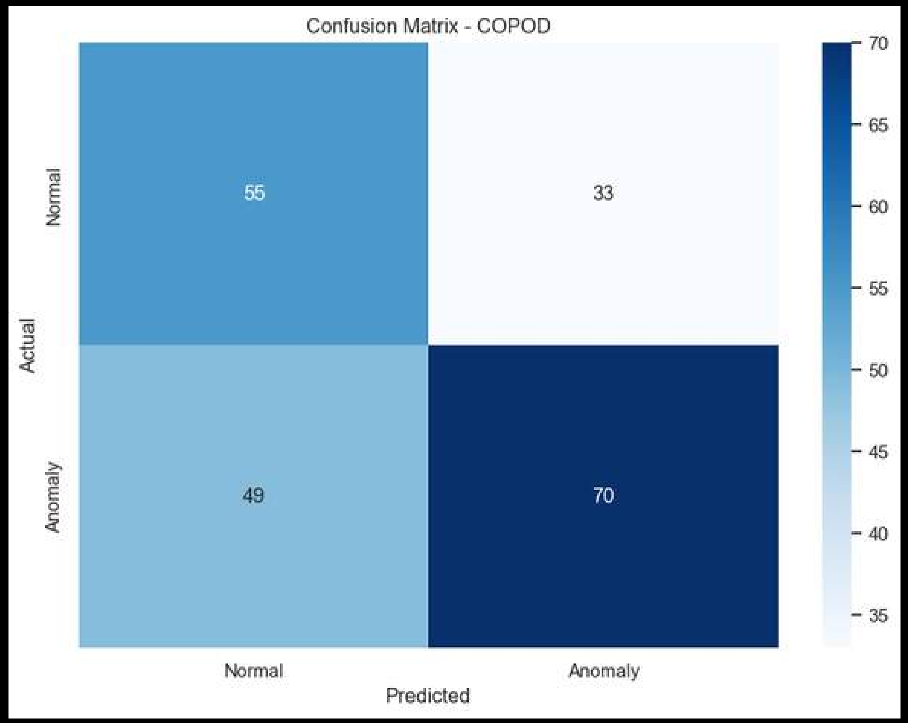
COPOD

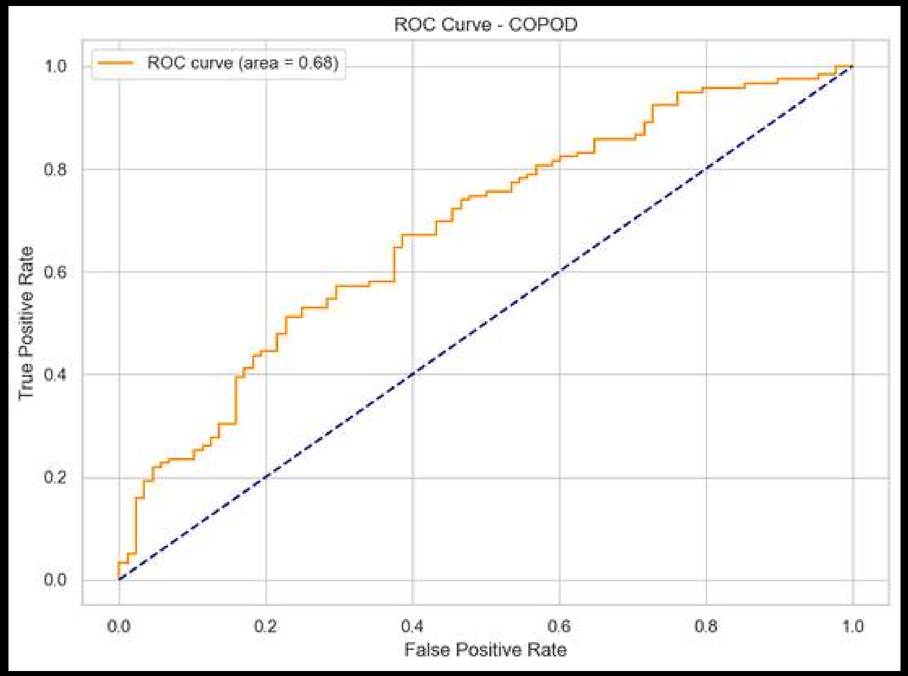
We apply the Copula-Based Outlier Detection (COPOD) algorithm, an unsupervised statistical method that does not require labeled data. COPOD estimates tail probabilities using **empirical copulas** to identify extreme deviations in the feature distribution. It computes outlier scores directly from the data without training, making it fast and interpretable. We evaluate its performance on the test set and visualize the results using **confusion matrices, ROC curves, anomaly score distributions**, and PCA projections to highlight separation between normal and anomalous instances.

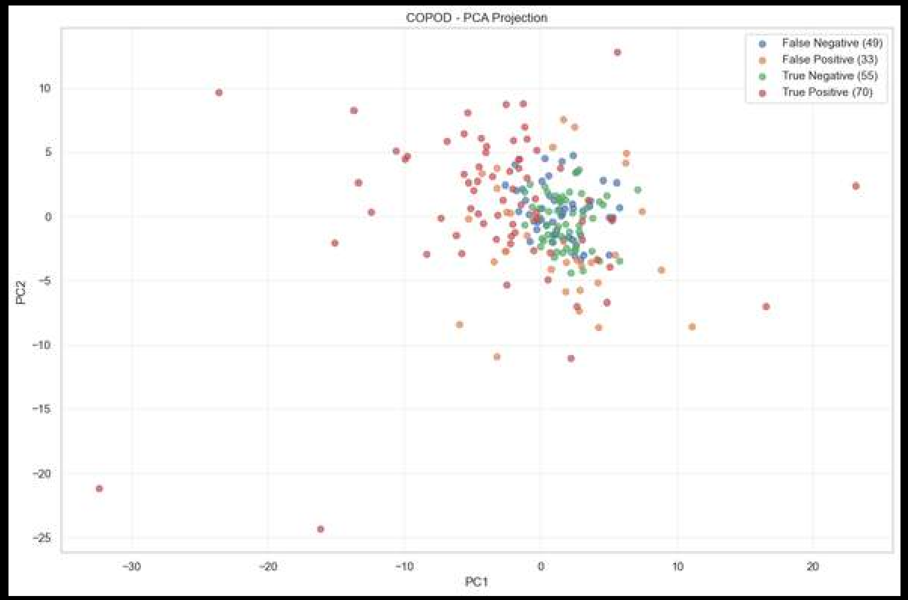
Precision: 0.6796

Recall: 0.5882

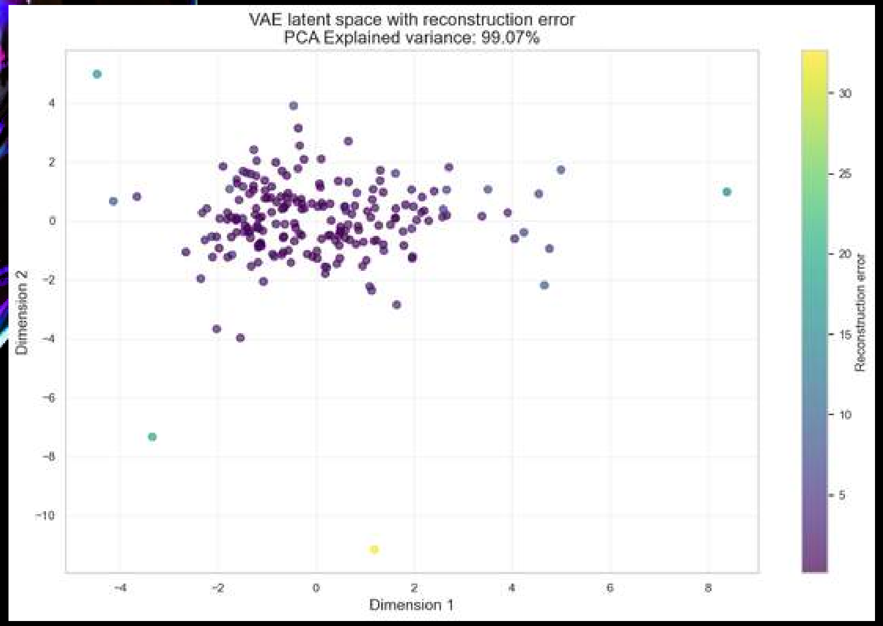
F1 Score: 0.6306



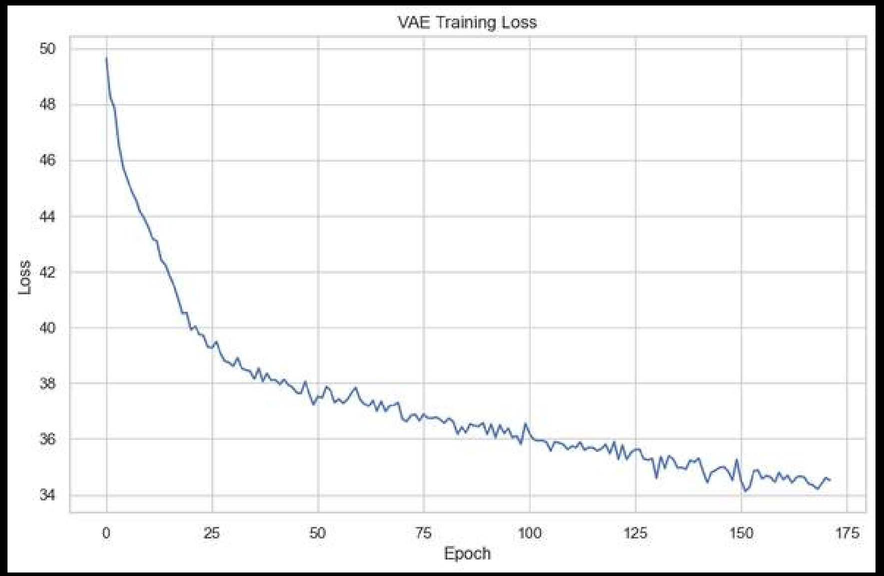




VARIATIONAL AUTOENCODER



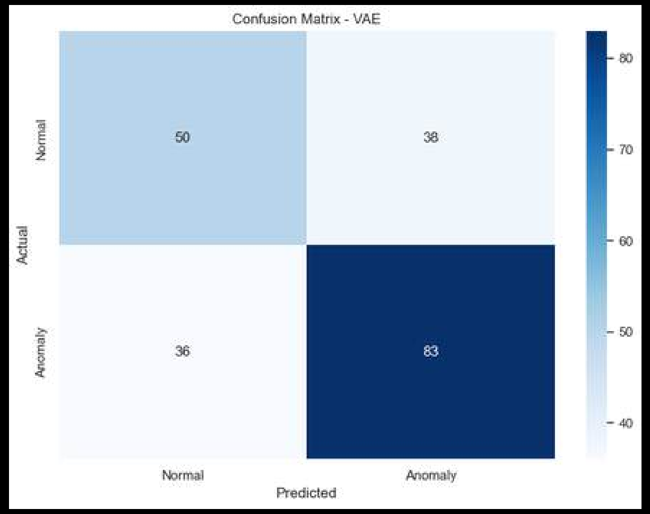
We implemented a Variational Autoencoder (VAE), an advanced generative model that extends traditional autoencoders by introducing probabilistic encoding into a structured latent space. The VAE encoder produces probability distributions characterized by mean and variance parameters, while the training process balances two objectives: **accurately reconstructing normal data** and **maintaining a regularized latent space through KL divergence**. The reparameterization trick enables backpropagation through stochastic sampling, creating more robust latent representations. For anomaly detection, we leverage **reconstruction error** as the detection criterion, where anomalies produce higher errors due to their deviation from the learned normal data distribution. The threshold is determined using the **cross-validation contamination rate**, and evaluation includes comprehensive visualizations of both input space and latent space projections.

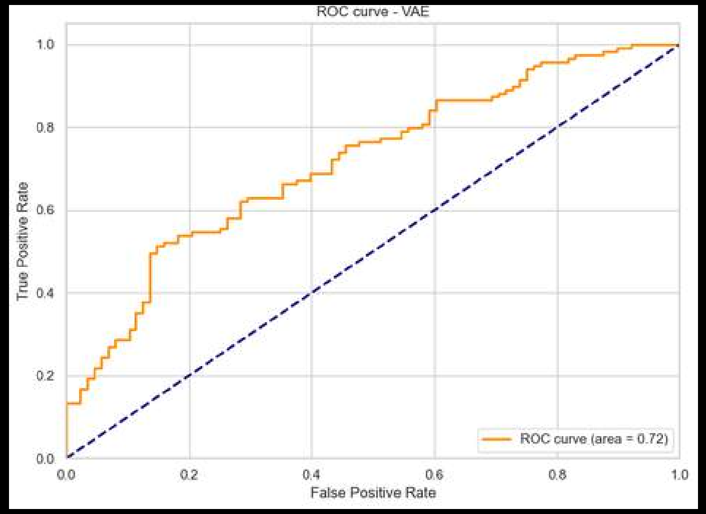


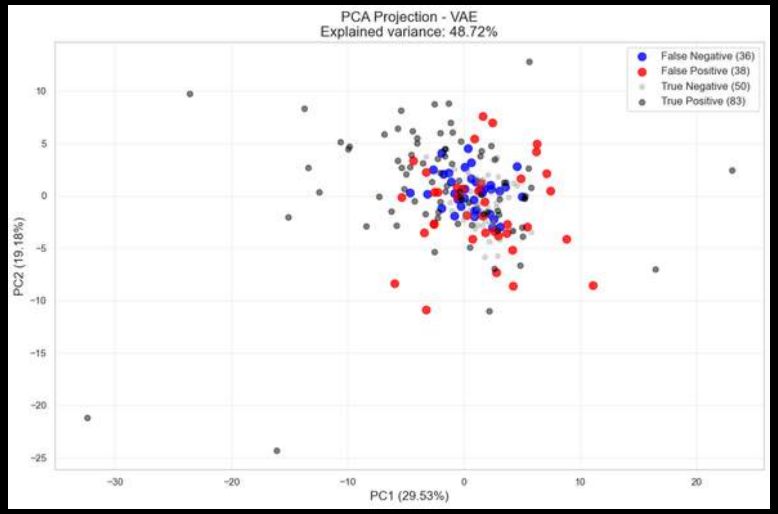
Precision: 0.6860

Recall: 0.6975

F1 Score: 0.6917







ENSEMBLE METHODS

- The ensemble methods implemented aims to improve anomaly detection performance by **combining predictions from multiple models** — spanning supervised, unsupervised, and deep learning approaches.

- The strategy involves evaluating all possible combinations of **2, 3, and 4** models from a predefined set (e.g., XGBoost, SVM, VAE, Isolation Forest, etc.). For each combination, a **logistic regression meta-model** is trained on the stacked predictions to generate final probabilities and classifications.

- Key performance metrics such as **Precision, Recall, F1-score, and ROC AUC** are computed for each ensemble. Additionally, the **average error correlation** among the selected models is calculated to promote diversity—favoring ensembles where individual models make different types of mistakes.

- The best-performing ensemble for each combination size (based on lowest error correlation) is then selected and visualized using a confusion matrix. This method **leverages the complementary strengths of heterogeneous models** to enhance robustness and predictive accuracy in detecting market anomalies.

Best combination of 2 methods :

GMM + XGBoost

- Precision: 0.7541
- Recall: 0.7731
- F1: 0.7635
- AUC: 0.7627
- Average Error Correlation: 0.2744

Best combination of 3 methods :

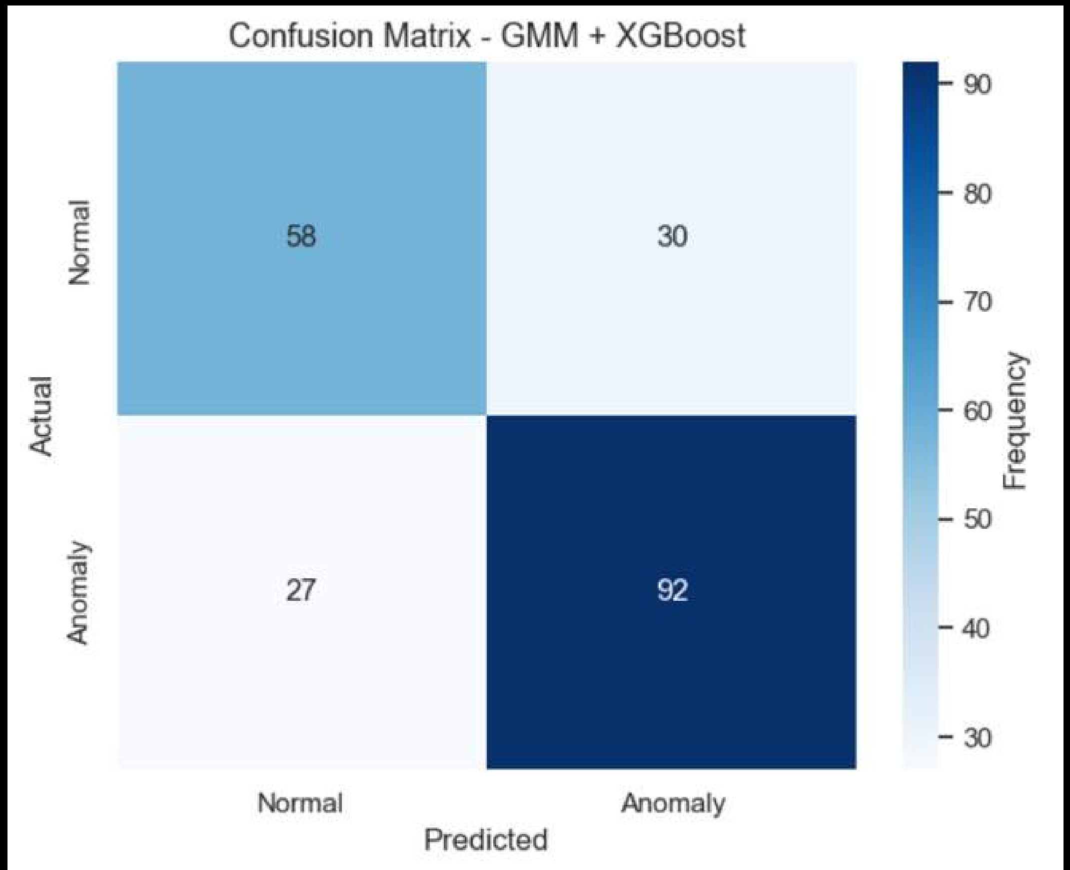
GMM + SVM + XGBoost

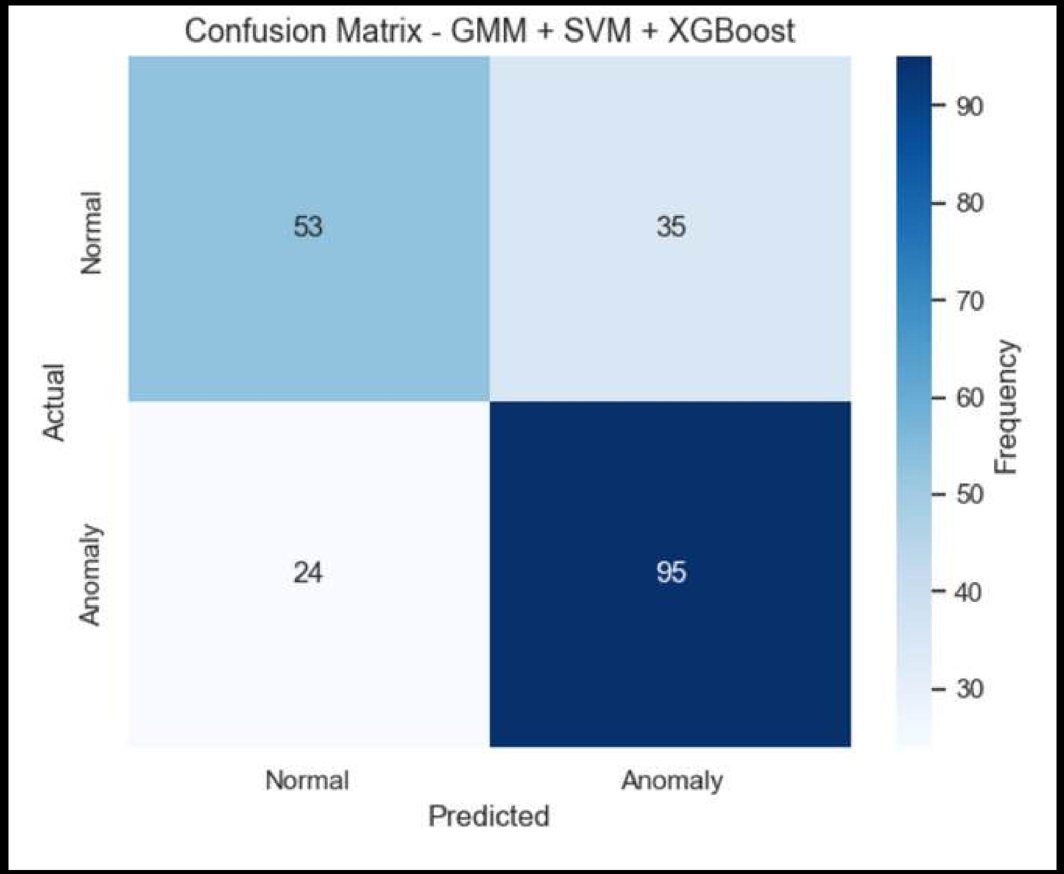
- Precision: 0.7308
- Recall: 0.7983
- F1: 0.7631
- AUC: 0.7826
- Average Error Correlation: 0.3674

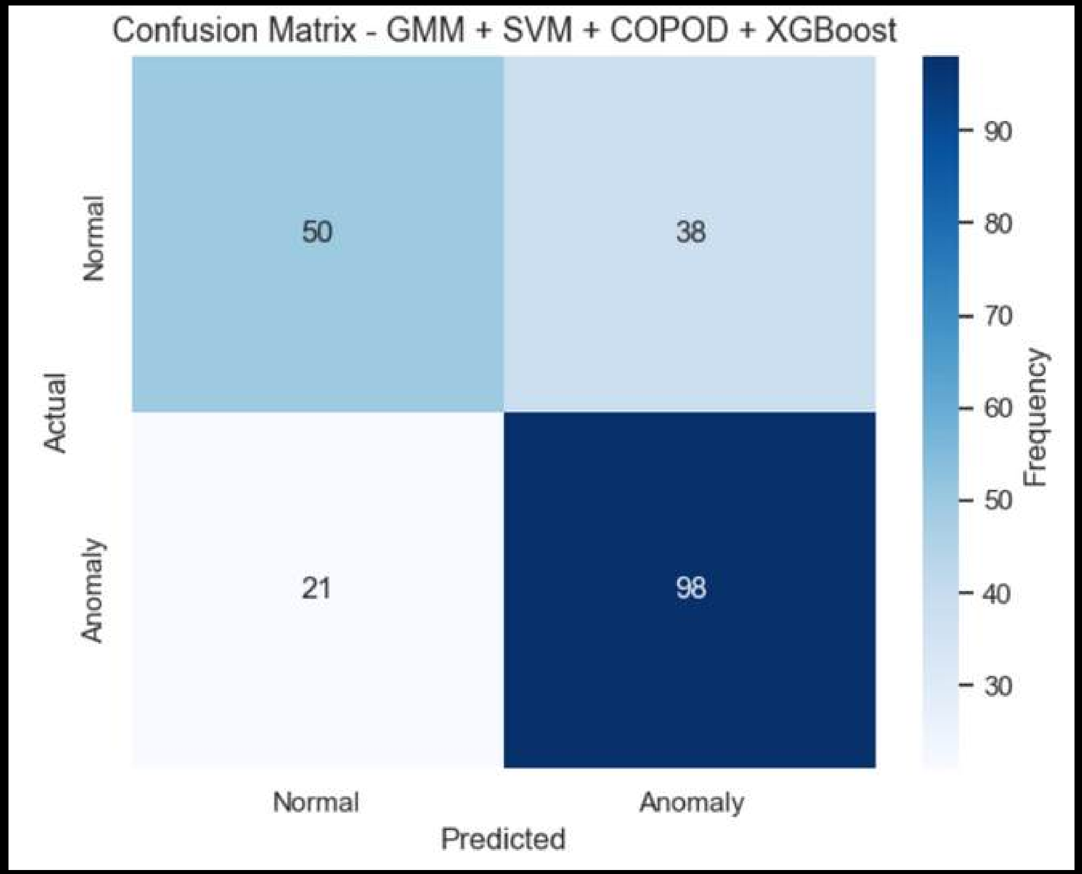
Best combination of 4 methods :

GMM + SVM + COPOD + XGBoost

- Precision: 0.7206
- Recall: 0.8235
- F1: 0.7686
- AUC: 0.7880
- Average Error Correlation: 0.4246







CONCLUSIONS

BEST MODEL	PRECISION	RECALL	F1 SCORE
GMM+SVM+COPOD+XGBoost	0.7206	0.8235	0.7686

In our analysis we focused on **minimizing** the average correlation between the **errors**, obtaining very robust methods; however this choice is not the unique possibility. For example a bank could choose to focus on the **Recall**, since a loan conceded to a very risky person is way more dangerous than the opposite case. But whatever the selected metric is the **ensemble methods** prove to be a very solid and reliable choice : they turn a group of imperfect models into a stronger overall predictor.

Group 10

THANK YOU!