Anomaly-Detection Report

Bianca-Nicola Dragomir[a]

[a] *Facultatea de Automatică și Calculatoare, UNSTPB, București.*

January, 2024

Abstract

Machine Learning is a powerful tool that is used in multiple domains, including medical diagnosis and disease detection. In this project, I performed anomaly detection on a medical dataset in order to predict the anomalies based on the given medical features. After conducting Feature Engineering on the provided data, I trained the Logistic Regression model on the improved data. I will discuss the taken approach, by explaining the methodology used as well as the obtained results.

1. Task Description

Data science has provided new opportunities for diagnosing diseases, with anomaly detection playing a pivotal role.

The motivation behind this study is to use advanced data science techniques to pinpoint possible medical anomalies.

During this challenge, I aim to develop a robust model that serves as a valuable diagnostic tool in clinical settings, by refining data pre-processing and model training methods.

2. Data

The dataset is designed for the challenging task of anomaly detection. There are 144 samples in the train data, and 192 in the test data. The columns are as follows:

- id: A unique identifier for each data point.
- features: Six different numerical features associated with each data point, indicating the incidence, tilt, angle, slope, radius, and grade of the taken measurements.
- *is_anomaly*: The target variable. A value of 0 indicates a normal data point, while a value of 1 indicates an anomaly.

3. Method

In this section, I'll discuss the steps I've taken, in the order employed by the Data Science Lifecycle, skipping the first two (Business Understanding and Data Mining) which were provided by the challenge itself.

3.1. Data Cleaning

I gathered the necessary data and looked for inconsistencies and missing values within the data. I decided not to eliminate the outliers, as the train data is already small enough, and there were no missing values.

3.2. Data Exploration

I visually analyzed the data to form hypotheses about the defined problem. Figures 1 and 2 show the distribution of both the normal and anomaly values, including the minimum, maximum, 25th percentile, median, and 75th percentile.

I noticed that, for most, there is considerable overlap between the two types of values, especially for 'feature_3'. The normal values have a wider range of values than the anomalies.

The anomalies seem to be in the lower values for all features besides 'feature_3'and 'feature_4'.

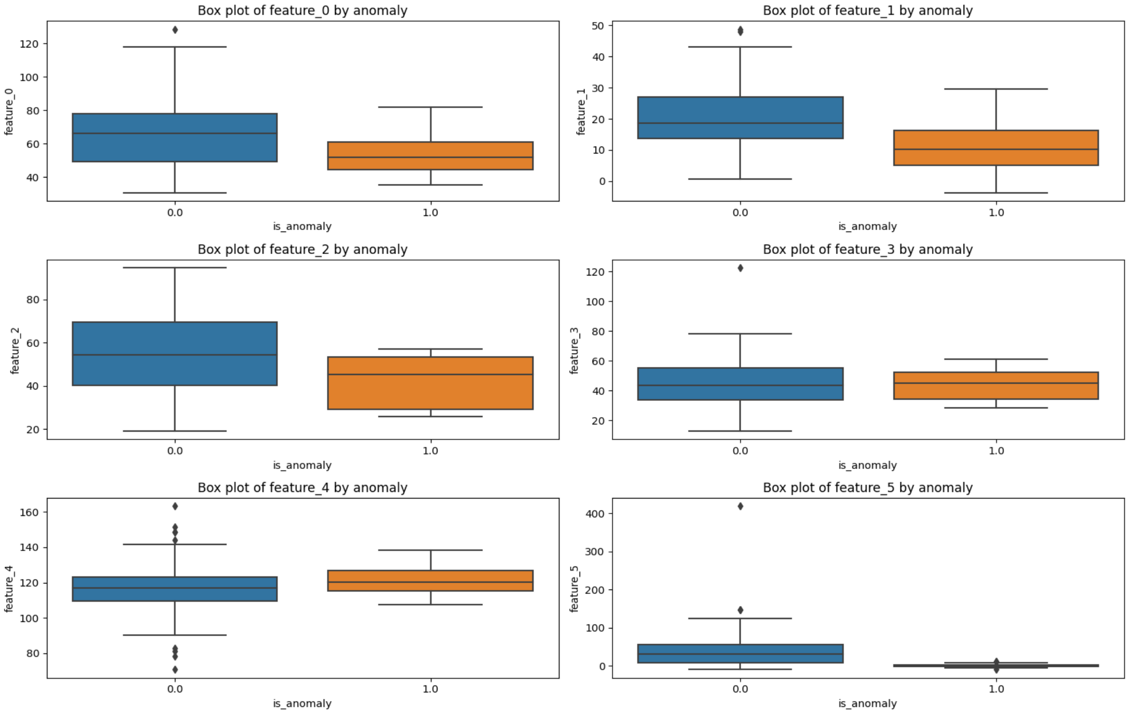

Figure 1: Box Plots of features (binary classification)

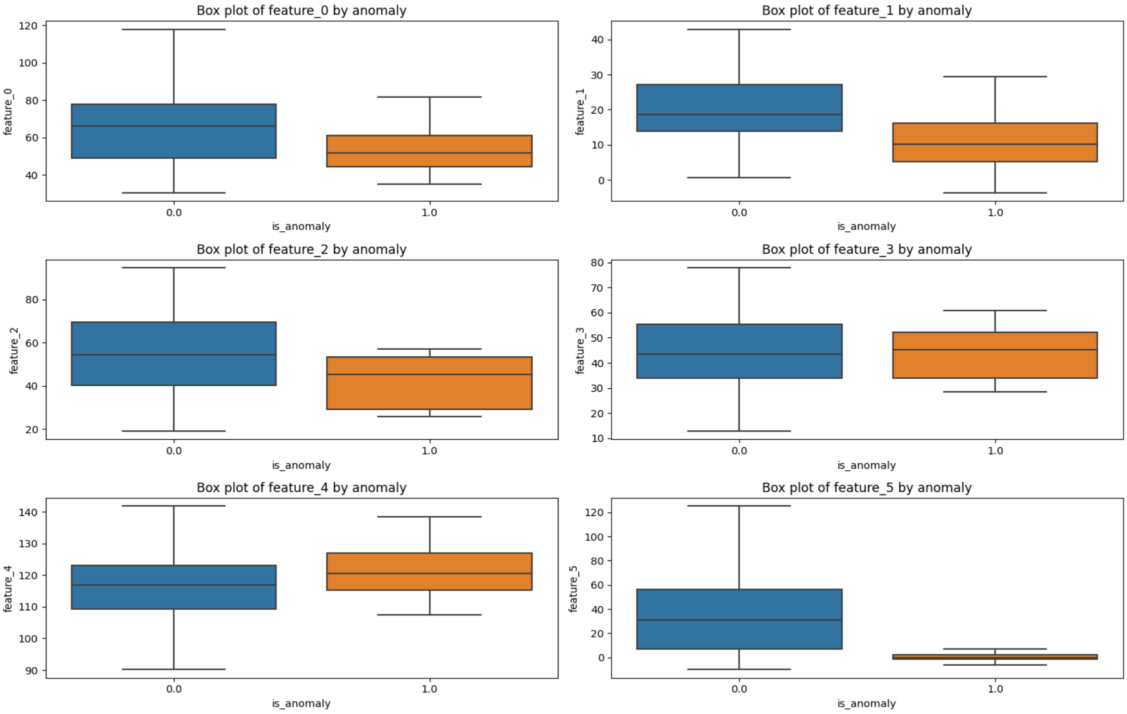

Figure 2: Box Plots of features with no outliers (binary classification)

The previous findings are confirmed by Figure 3, which shows a red coloring mostly in the lower part of the blue line.

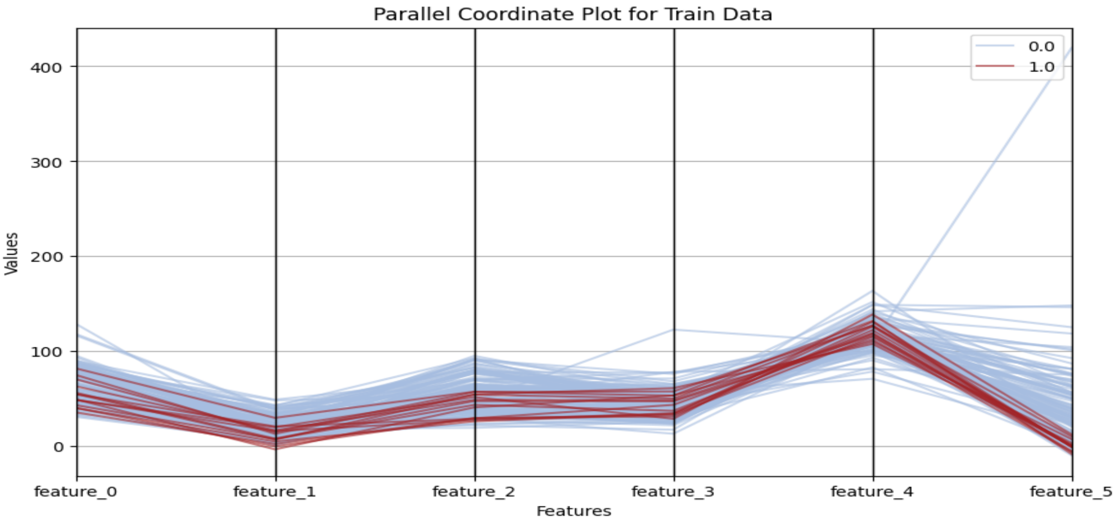

Figure 3: Parallel Coordinate Plot of features

Figure 4 is of a similar nature to Figure 2 and Figure 3, indicating the tendency for anomalies to appear in the lower range of the train data.

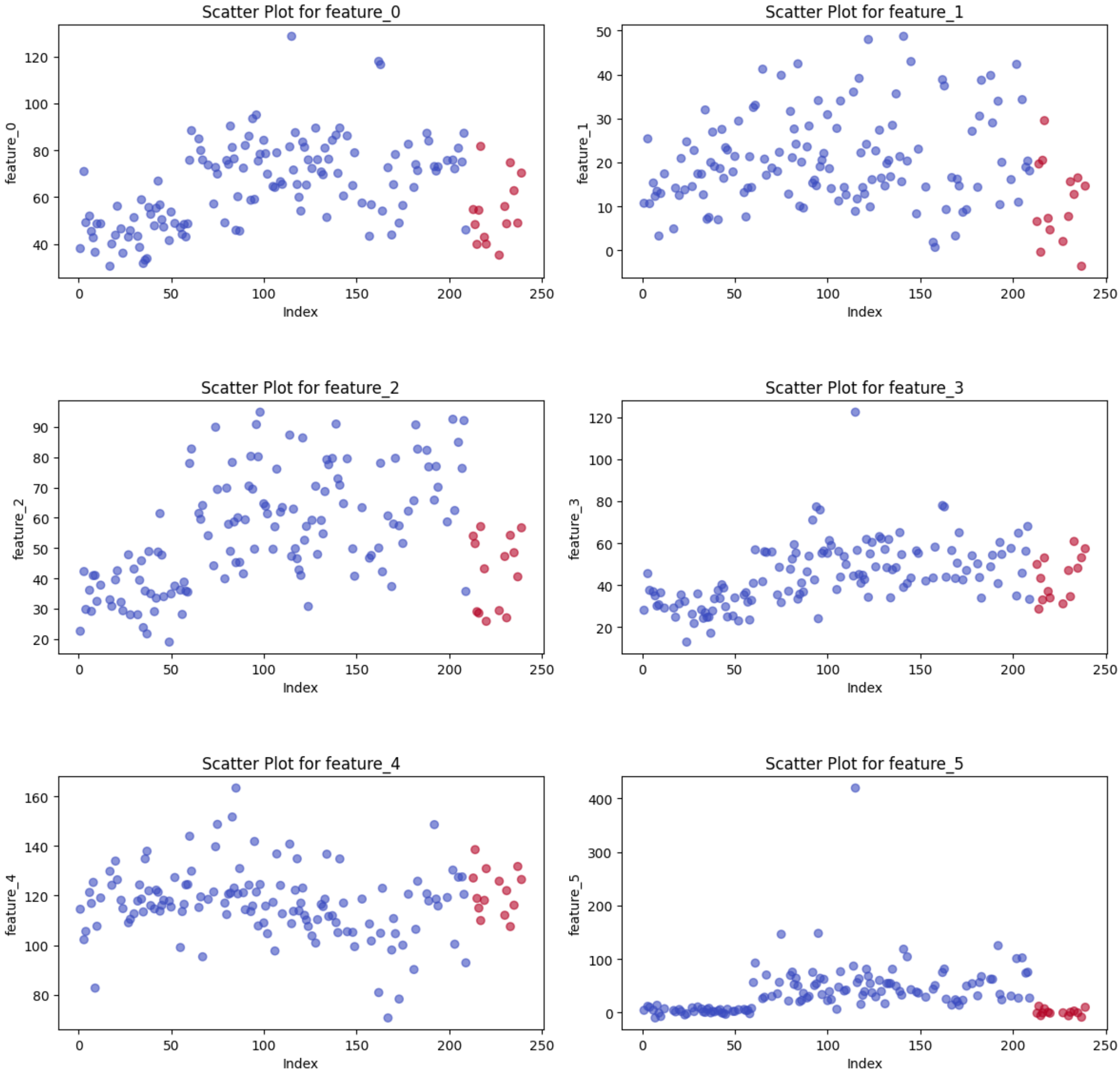

Figure 4: Scatter Plots of features

Next, Figure 5 plots the distribution of 1.0 and 0.0 values in the y_train data set, which appears to be quite imbalanced.

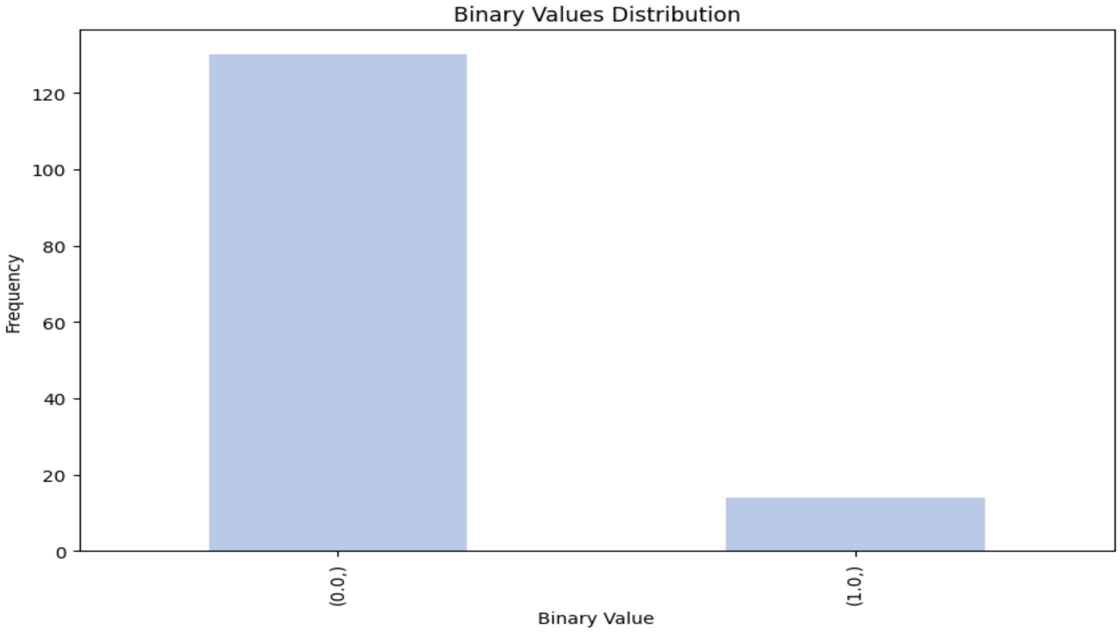

Figure 5: Bar Plot Distribution of binary values

3.3. Feature Engineering

I began considering the resulting plots to construct more meaningful features based on the given raw data. The Bar Plot of Figure 5 shows an imbalance concerning the number of binary values in y_train, such that there are 130 values of 0.0 and only 14 of 1.0. I applied the ADASYN algorithm to over-sample the data for a more balanced result. The resulting numbers are as follows: 130 values of 0.0 and 132 of 1.0, as can be seen in Figure 6. The data is now called x_train_balanced and y_train_balanced.

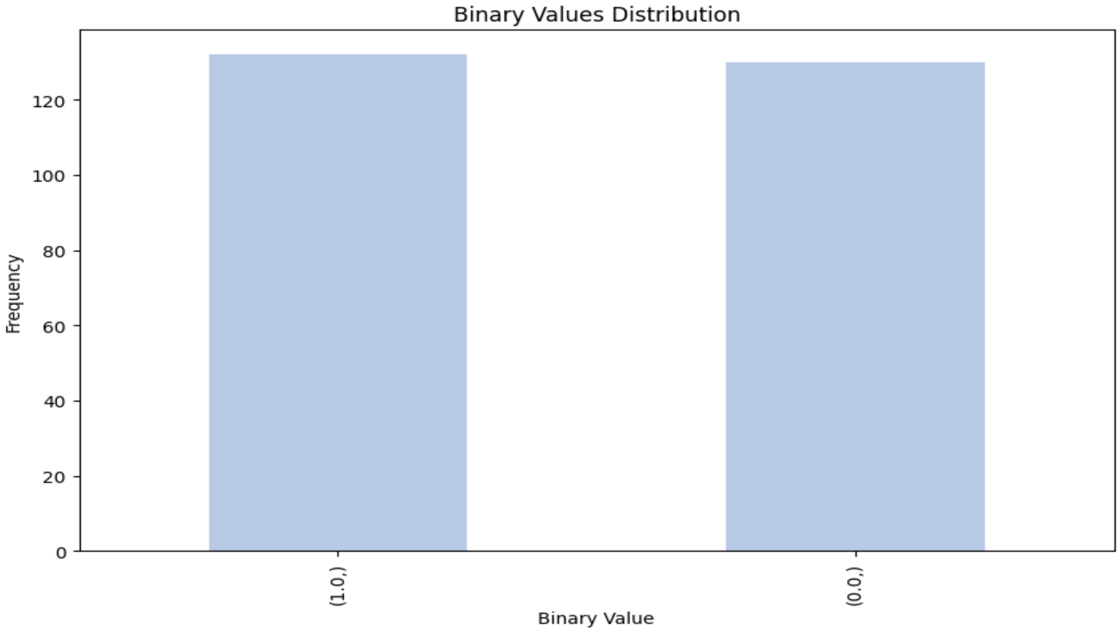

Figure 6: Bar Plot Distribution of balanced binary values

Moreover, observing the Box Plot of Figure 2, I have determined that 'feature_3' does not show a significant difference in the distribution between the normal and anomaly values. I have therefore dropped the column from both x_train_balanced, as well as x_test.

I followed the indications given in the laboratory, in the Pre-processing Data part of it. I scaled the data using both Robust Scaler and Standard Scaler, and then I applied the PCA algorithm for data dimension reduction.

3.4. Predictive Modeling

By usage of the Optuna optimization framework, I determined the best model to be used with the best set of hyper-parameters. I made multiple attempts and errors to make sure I reached the highest score I was capable of. The Optuna objective function takes a set of suggested classifiers and estimates the best hyper-parameters values based on its history record.

The tried classifiers are the following:

1. LogisticRegression
2. KNeighbors
3. SVC
4. DecisionTree
5. RandomForest
6. AdaBoost
7. GradientBoosting
8. GaussianNB

After 300 trials, Optuna returned the best parameters as: 'classifier': 'LogisticRegression', 'C': 0.1005670099499378, 'penalty': 'l2', 'solver': 'sag', with an accuracy of 92.70%.

4. Results

The analysis of the medical dataset using Logistic Regression revealed significant findings. The model achieved an accuracy of 92.7% according to its own score, and an accuracy of 81.25% on the kaggle website, indicating a strong performance in anomaly detection.

The balanced dataset, post feature engineering and application of the ADASYN algorithm, allowed for more effective model training and prediction.

In order to assess the final findings, I analyzed the resulting target data set by plotting the train data against the complete test data, as can be seen in Figure 7.

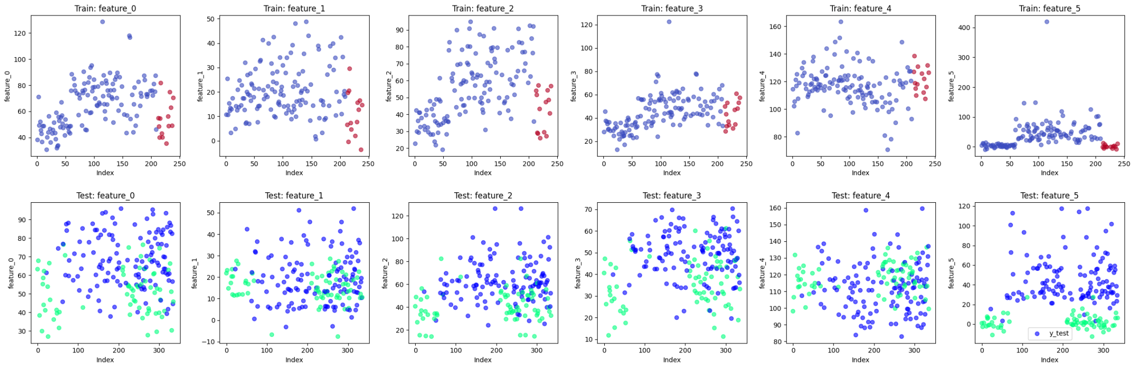

Figure 7: Scatter Plots of features

5. Conclusion

In conclusion, the best results came from a few key steps. First, we balanced the training data with the ADASYN algorithm. Next, I removed 'feature_3' because it didn't show important differences between normal and abnormal data in the box plots. I then scaled the data with Robust Scaler and Standard Scaler. I also used PCA to keep the data's most important information, preserving 95% of its original details. Lastly, I trained the model with Logistic Regression using the best settings I found with Optuna. This process gave us the highest score.