TÉCNICO LISBOA

DS 2024

Data Science Project

Team nr: 4	**Student 1:** Vasco Roda Félix	**IST nr:** 99131
	Student 2: António Maria de Almeida Pinto Alves Jotta	**IST nr:** 99893
	Student 3: António Vasco Morais de Carvalho	**IST nr:** 102643
	Student 4: Tiago Lucas Borbinha	**IST nr:** 103382

CLASSIFICATION

1 DATA PROFILING

Data Dimensionality

Set 1 - Large dataset which will be useful for modeling later, reducing the risk of overfitting, and indicates the use of a sample. Won't suffer from the curse of dimensionality. Has 17 variables, only 4 being numeric as some of them are codes representing something. Has 3 variables with significant missing values which may mean they will be dropped; **Set 2** - many variables (87), suggesting the importance of feature engineering. Target variable is binary and the rest numeric. No missing values.

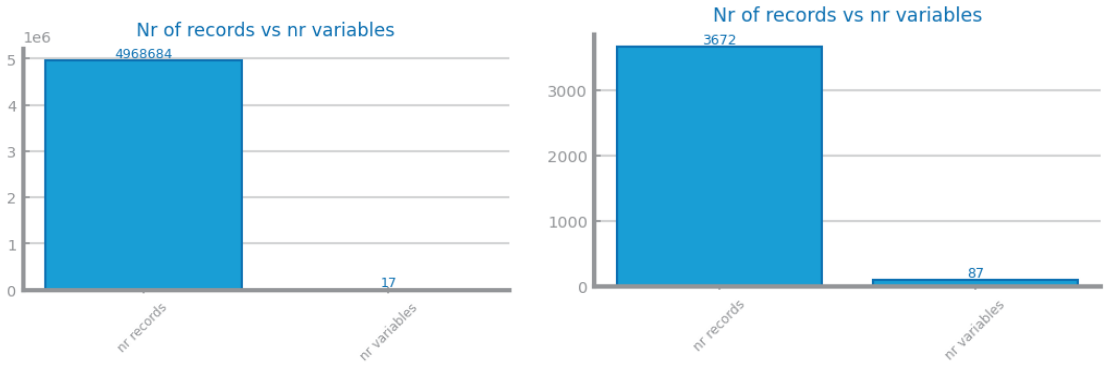

Figure 1: Nr Records x Nr variables for dataset 1 (left) and dataset 2 (right)

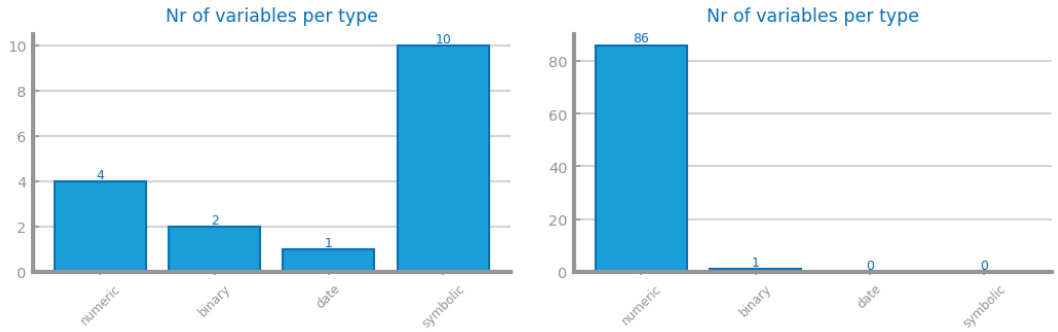

Figure 2: Nr variables per type for dataset 1 (left) and dataset 2 (right)

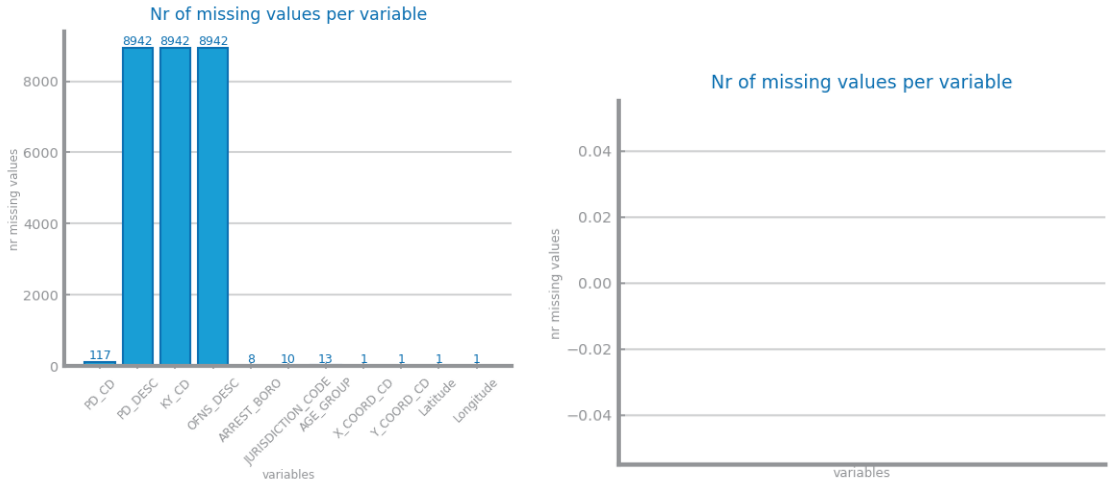

Figure 3: Nr missing values for dataset 1 (left) and dataset 2 (right)

Data Distribution

Set 1 - Numeric values are related to the location of crimes. So both horizontal and vertical related variables have a close resemblance to each other. X_COORD and Longitude closely match normal distributions. High number of outliers; **Set 2** - Boxplots show presence of outliers ($x_8, x_{14}, x_{15}, x_{31} \ldots$), typical of financial data. Some histograms resemble normal distributions; **Both** - Very unbalanced which suggests the use of balancing techniques in data preparation.

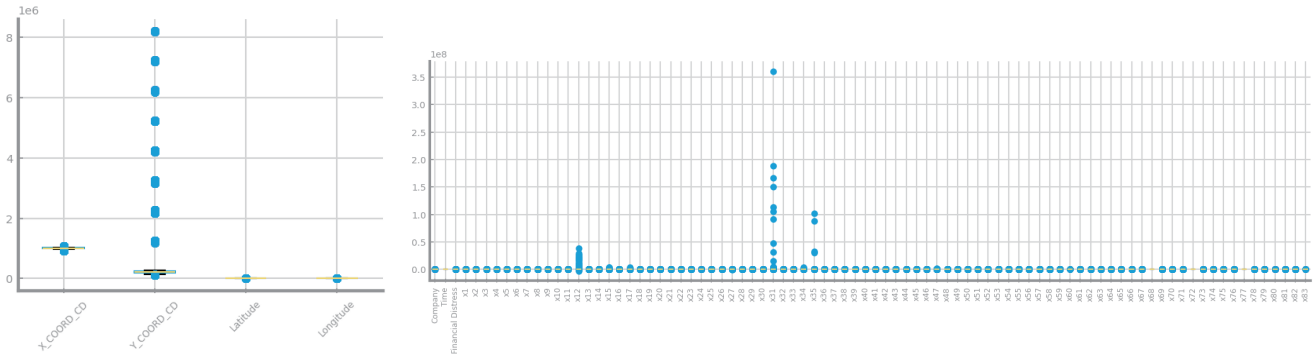

Figure 4: Global boxplots dataset 1 (left) and dataset 2 (right)

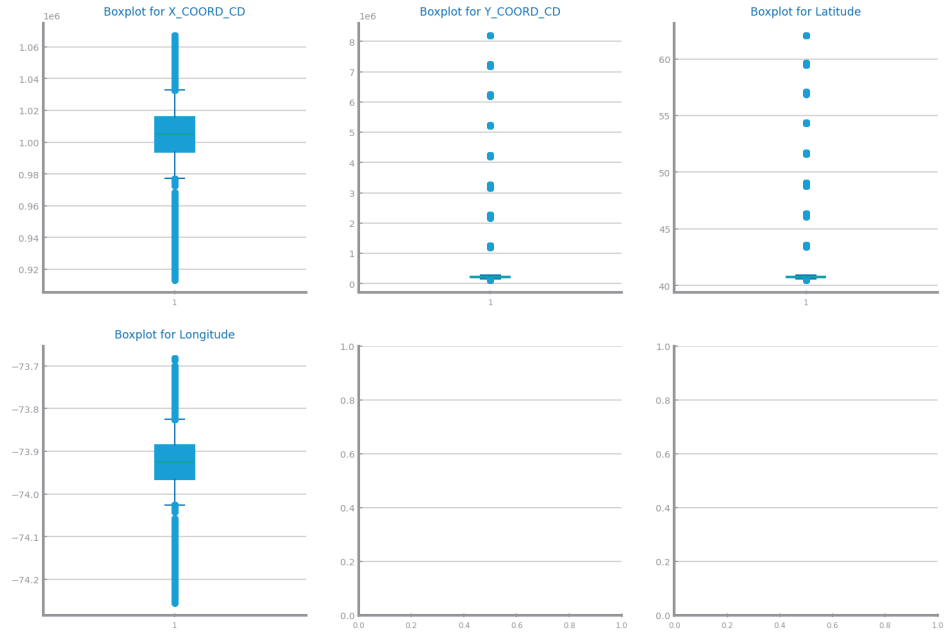

Figure 5: Single variables boxplots for dataset 1

This figure is too big to present in the report. See it here.

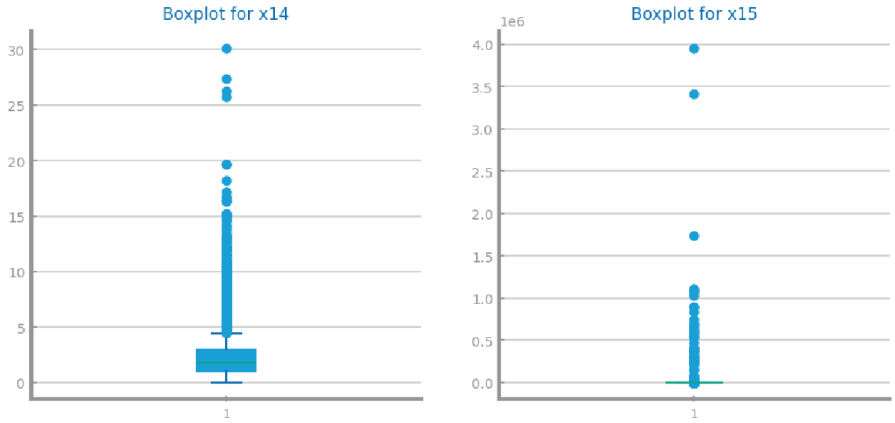

Figure 6: Single variables boxplots for dataset 2.

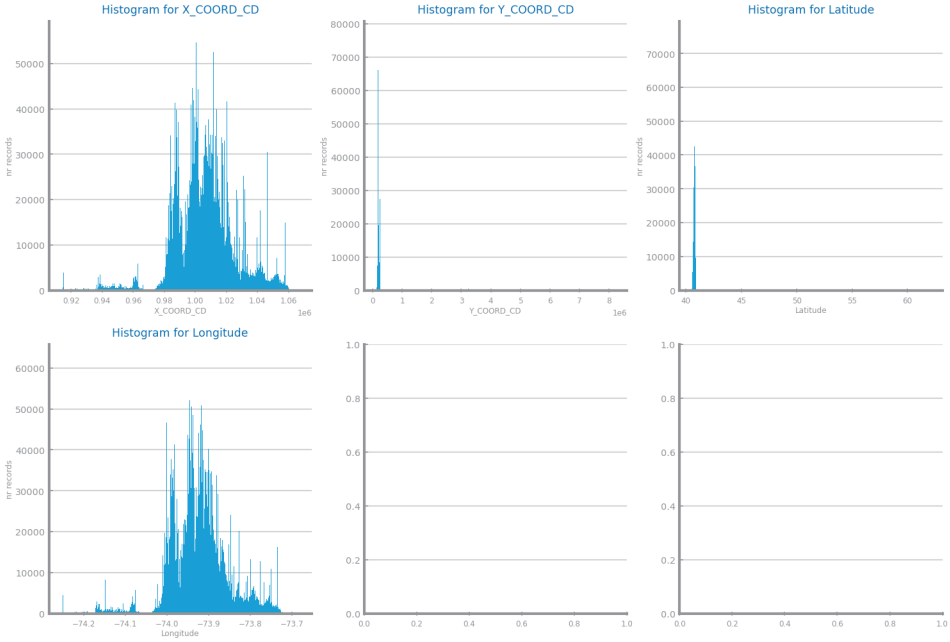

Figure 7: Histograms for dataset 1.

This figure is too big to present in the report. See it here.

Figure 8: Histograms for dataset 2

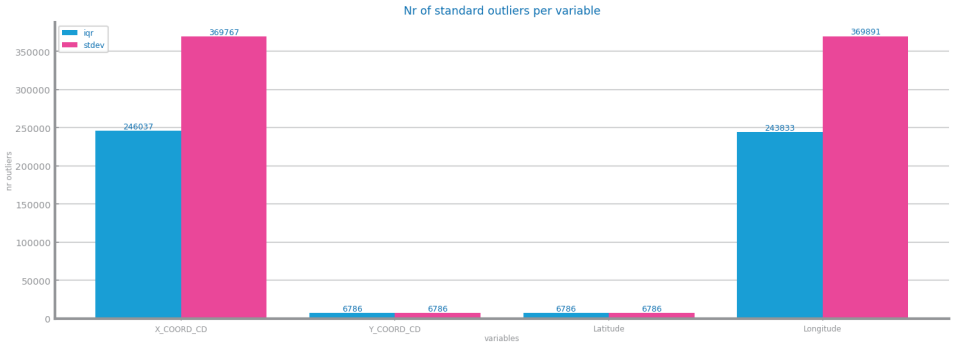

Figure 9: Outliers study dataset 1

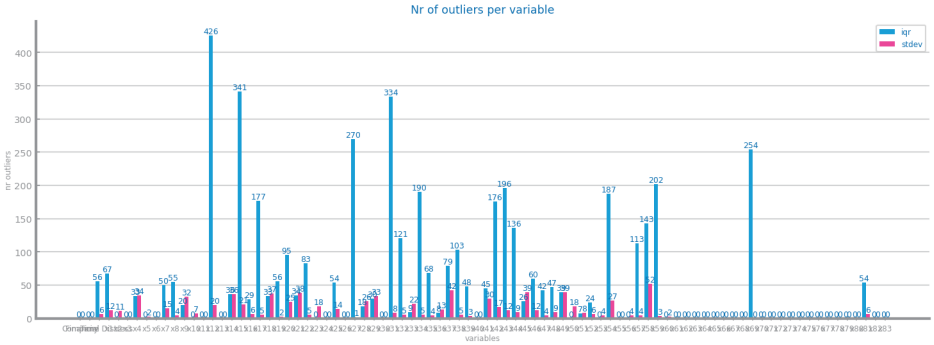

Figure 10: Outliers study dataset 2

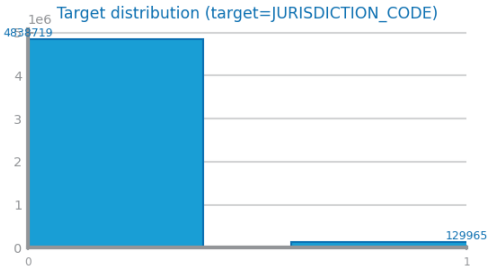

Figure 11: Class distribution for dataset 1

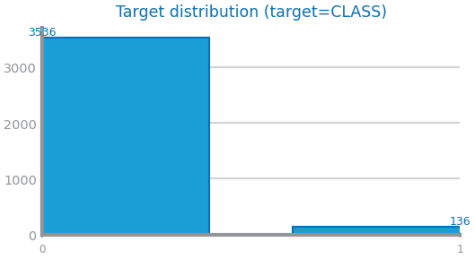

Figure 12: Class distribution for dataset 2

Data Granularity

Set 1 - Has a variable where granularity can be evaluated (ARREST_DATE). Temporarily created new numerical variables referring to the year, quarter, month and day. In Figure 13, the number of records has been decreasing over the years. The last quarter and months have the fewest records.The day with the least criminal cases is the 31^{st}, since only a few months have this number of days; **Set 2** - Contains only numeric/binary data - Figure 2. Therefore, no granularity analysis.

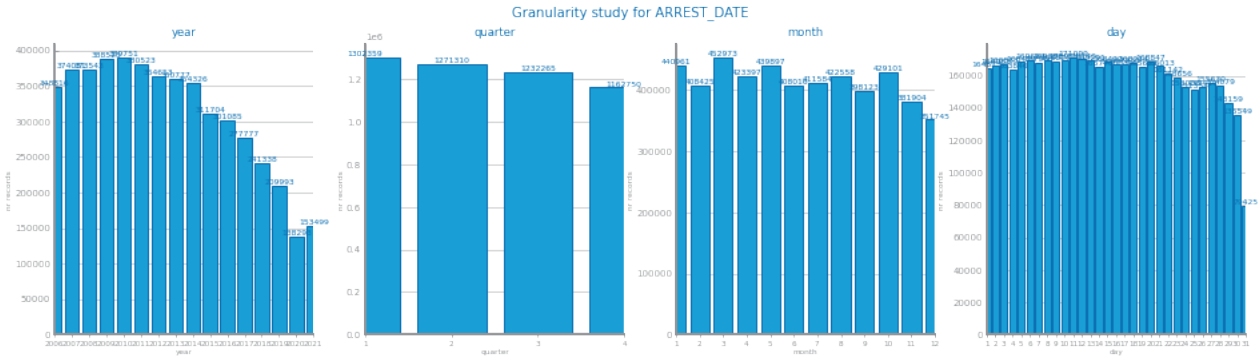

Figure 13: Granularity analysis for dataset 1

Set 2 contains only numeric/binary data - Figure 2.
Therefore, no granularity analysis.

Figure 14: Granularity analysis for dataset 2

Data Sparsity

Set 1 - Figure 17 only shows two pairs of variables with high levels of correlation, so it will be possible to remove one variable from each pair. None of the variables has a high binary correlation with the target; **Set 2** - Figure 18 shows two main clusters ($\{\text{Company}, \ldots, x_{14}\}$ and $\{x_{61}, \ldots, x_{79}\}$) indicating redundant variables that could be removed. Some variables exhibit low variance (x8 and x38 are shown), suggesting they may also be removed.

This figure is too big to present in the report. See it here.

Figure 15: Sparsity analysis for dataset 1

This figure is too big to present in the report. See it here.

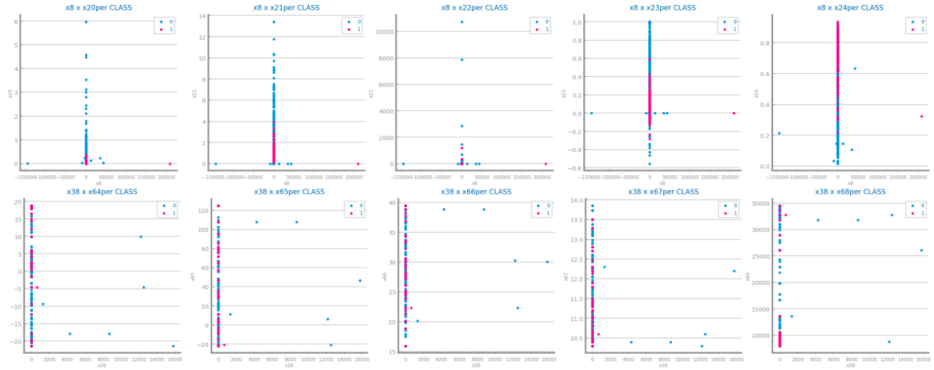

Figure 16: Sparsity analysis for dataset 2

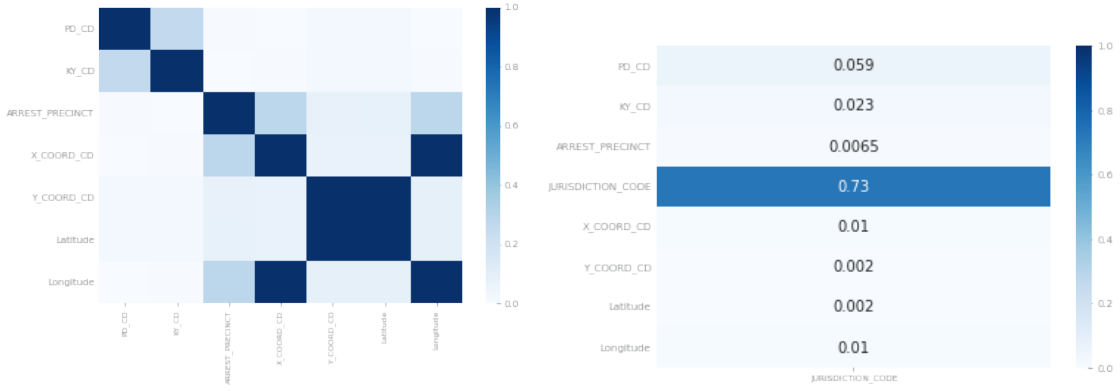

Figure 17: Correlation analysis for dataset 1 (right) and binary correlation for dataset 1 (left)

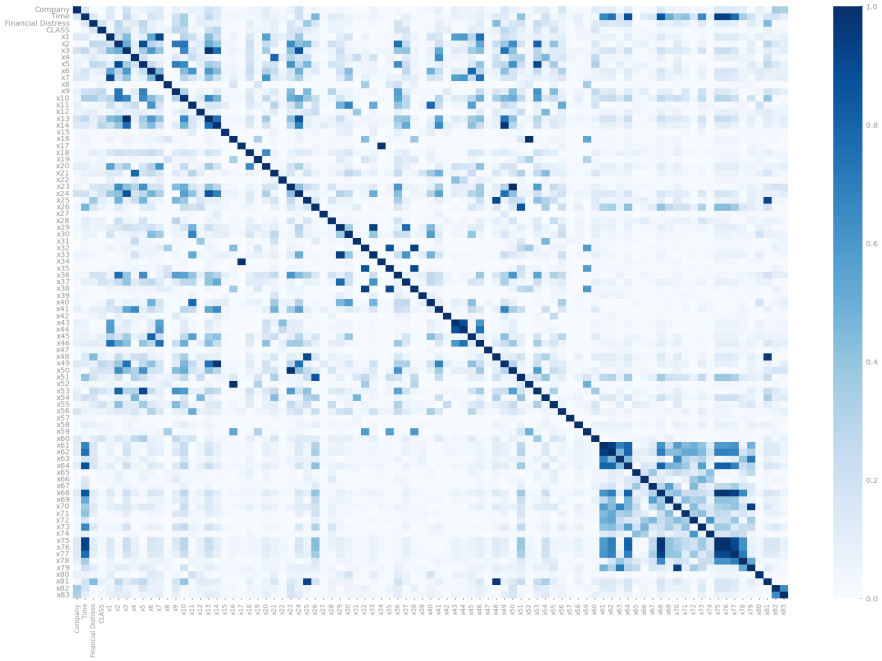

Figure 18: Correlation analysis for dataset 2

2 DATA PREPARATION

Variables Encoding

Set 1 - Used numeric encoding for the variables (JURISDICTION_CODE, LAW_CAT_CD, ARREST_BORO, PERPRACES, AGE_GROUP, OFNS_DESC, PD_CD, KY_CD, and PERP_SEX) assigning an integer to each different occurrence, cyclic encoding for (ARREST_DATE) in which a YEAR, DAY, MONTH_COS and MONTH_SIN variable were created, and dropped the variables (LAW_CODE and PD_DESC) as these are just more descriptive variables of others; **Set 2** - Contains only numeric/binary data - Figure 2. Therefore, no variable encoding.

Missing Value Imputation

Set 1 - When processing missing values, a slightly better average was obtained in the combined results of NB and KNN when the records that had these values were eliminated. And so the selected approach was "Drop missing values"; **Set 2** - Does not have any missing values - Figure 3. Therefore, no missing value imputation.

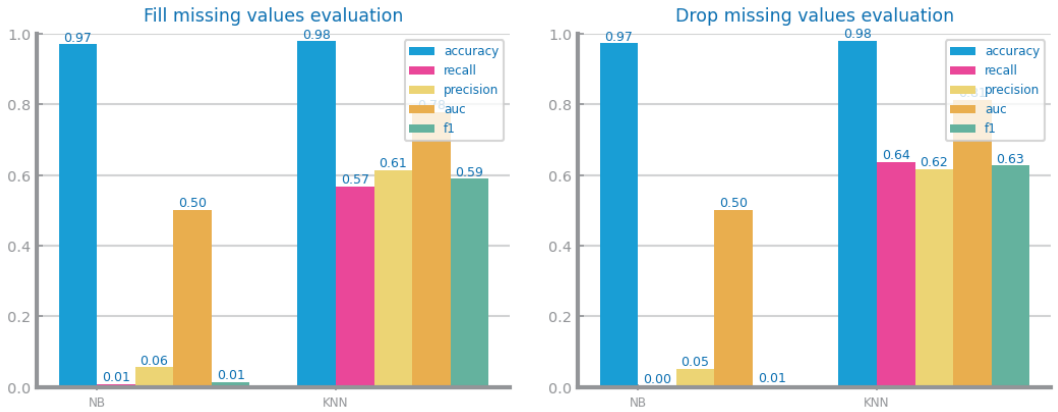

Figure 19: Missing values imputation results with different approaches for dataset 1

Set 2 does not have any missing values - Figure 3.
Therefore, no missing value imputation.

Figure 20: Missing values imputation results with different approaches for dataset 2

Outliers Treatment

Set 1 - Decided to keep the "Drop Missing Values", since none of the methods of dealing with outliers had a better average in the NB and KNN results. Most outliers in Figure 9 are not real outliers - Figure 5; **Set 2** - Used "Drop outliers", removing 95 records from the original dataset, with thresholds set for each variable. Yielded the best average combined results for NB and KNN. Recall does decrease notably for KNN which shows a problem with FN, vice-versa for NB.

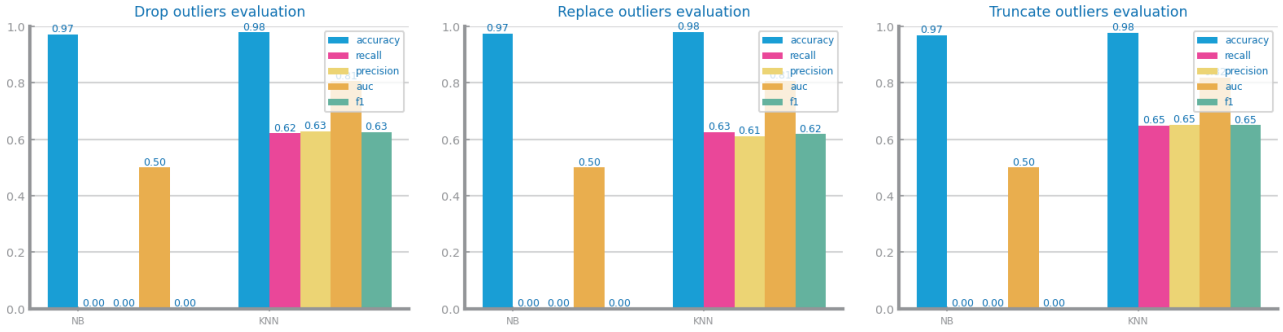

Figure 21: Outliers imputation results with different approaches for dataset 1

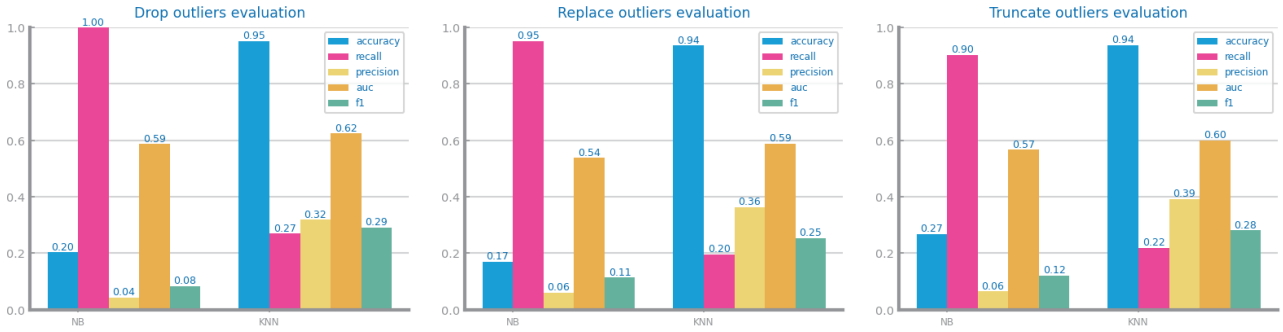

Figure 22: Outliers imputation results with different approaches for dataset 2

Scaling

Set 1 - Scaling was not applied since the previous step had better results; **Set 2** - Chose "Standard scaler" due to better results.

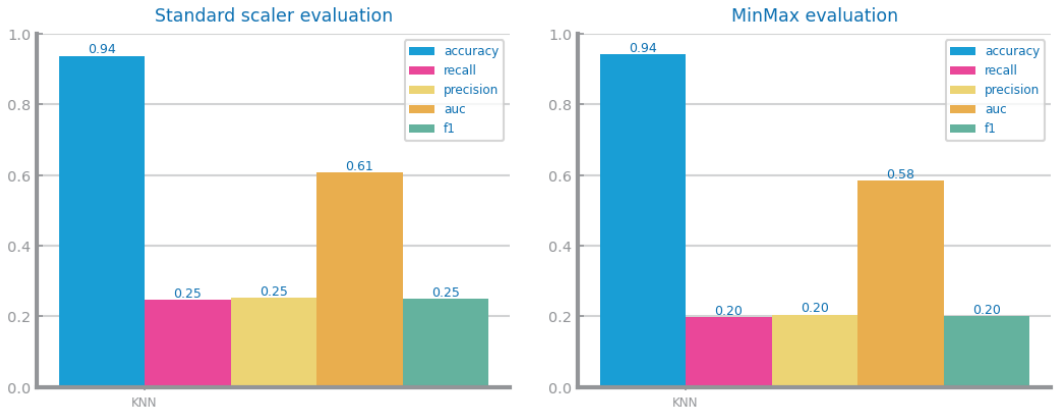

Figure 23: Scaling results with different approaches for dataset 1

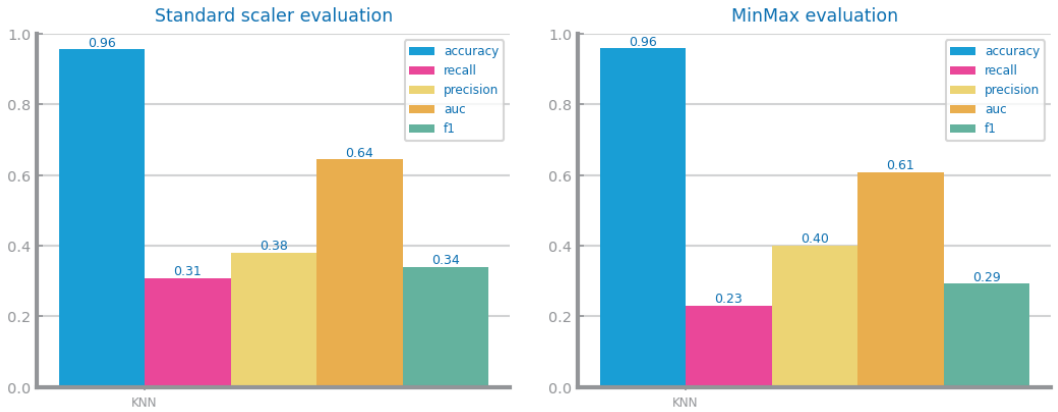

Figure 24: Scaling results with different approaches for dataset 2

Balancing

Set 1 - Selected "SMOTE" as the approach, since it had the best results. Undersampling is not ideal since a lot of valuable data would be lost; **Set 2** - Used "SMOTE", proceeding with 5516 records, more useful data for training. This method achieved the best average combined results for NB and KNN; **Both** - For KNN, Recall and Auc increase notably which means the model correctly identifies more TP and reduced FN, and is more able to distinguish between TP-TN and FP-FN, respectively.

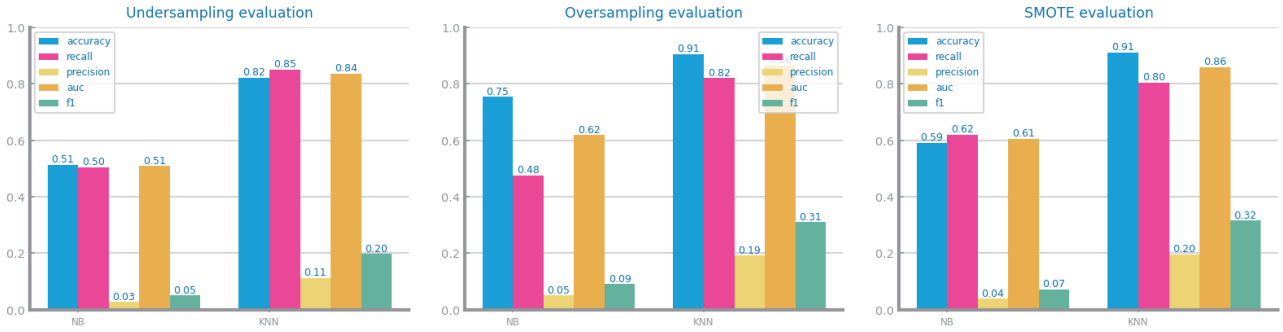

Figure 25: Balancing results with different approaches for dataset 1

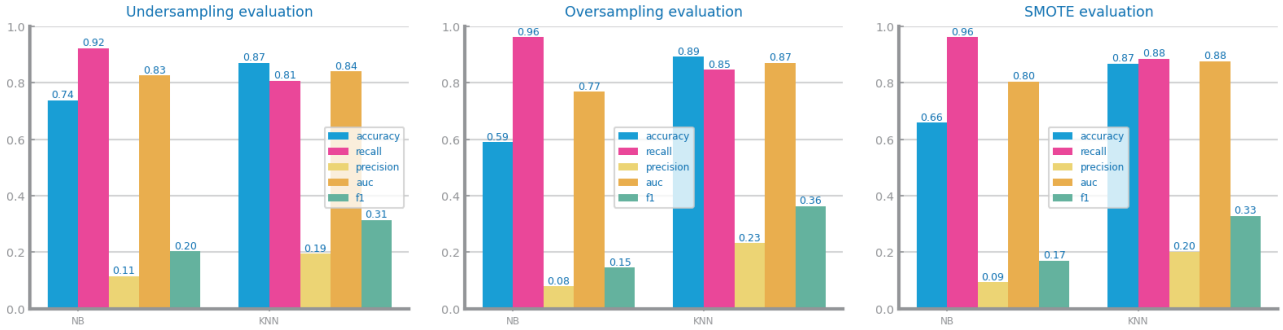

Figure 26: Balancing results with different approaches for dataset 2

Feature Selection

Set 1 - The average of NB and KNN scores improved slightly when removing variables with high correlation (3) and stayed practically the same when removing variables with a variance higher than 0.9 (9). Chose "Drop redundant variables" to continue; **Set 2** - KNN improved when removing variables with a correlation higher than 0.8 (27), NB decreased in performance. KNN performed worse when removing variables with variance higher than 0.9 (9). "Removing Low Variance variables" was the chosen approach.

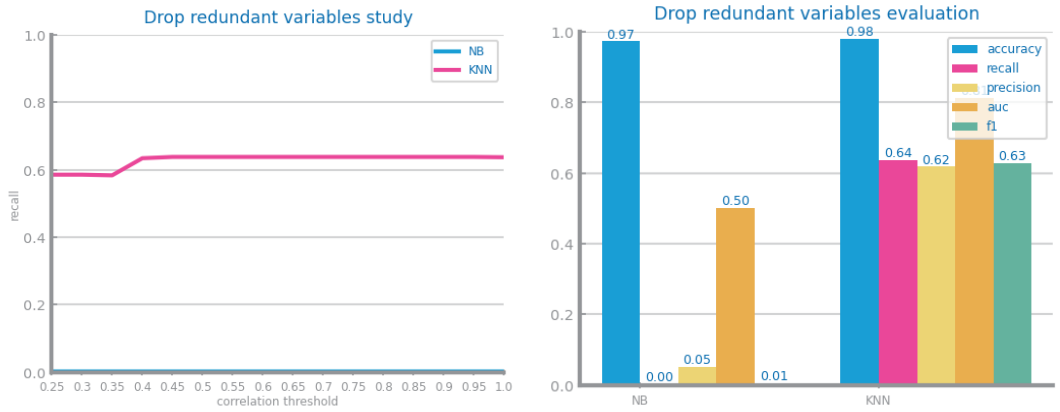

Figure 27: Feature selection of redundant variables results with different parameters for dataset 1

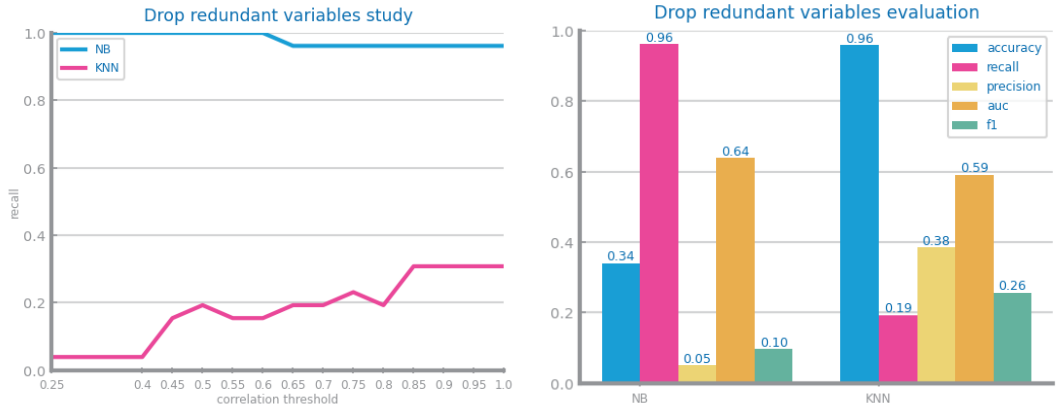

Figure 28: Feature selection of redundant variables results with different parameters for dataset 2

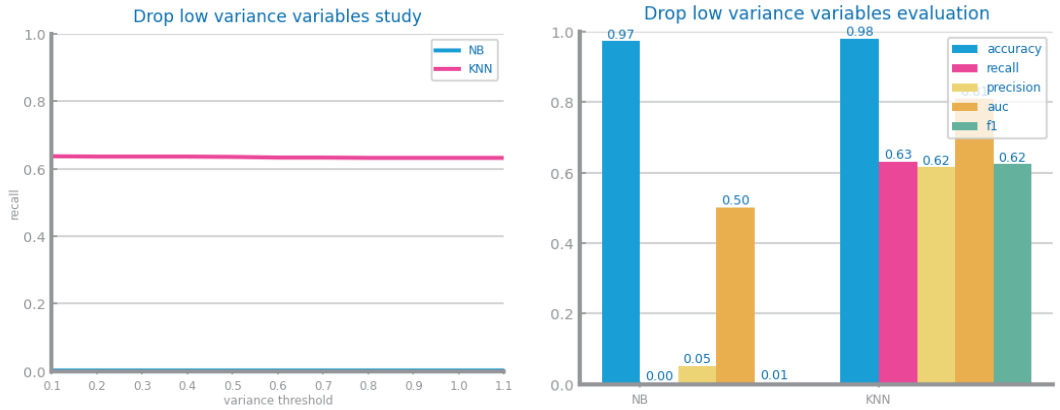

Figure 29: Feature selection of relevant variables results with different parameters for dataset 1 (variance study)

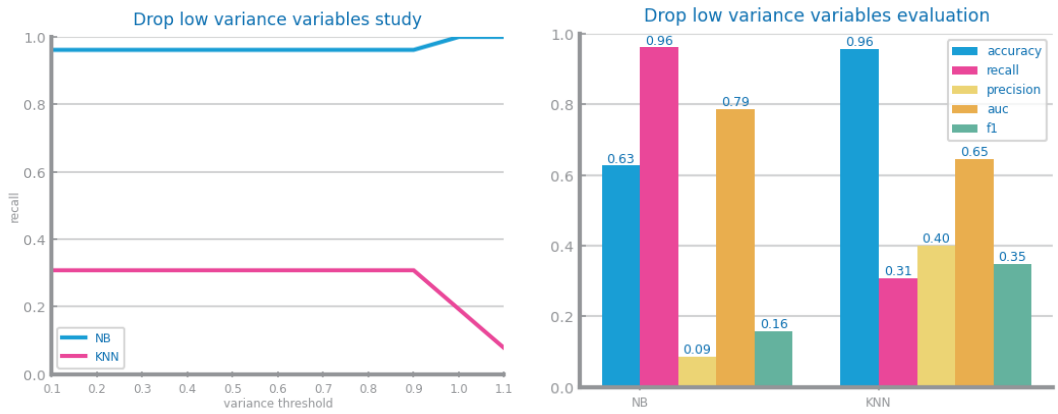

Figure 30: Feature selection of relevant variables results with different parameters for dataset 2 (variance study)

Feature Extraction (optional)

Not done.

Not done.

Figure 31: Principal components analysis and feature extraction results for dataset 1

Not done.

Figure 32: Principal components analysis and feature extraction results for dataset 2

Additional Feature Generation (if done)

Not done.

Not done.

Figure 33: Feature generation results for dataset 1

Not done.

Figure 34: Feature generation results for dataset 2

3 MODELS' EVALUATION

Set 1 - A sample of 1% (50k of ≈5M) of the dataset was used. Sample hold-out was applied; **Set 2** - Hold-out was used; **Both** - Training was performed with an 80%-20% split for training and testing, respectively. Evaluation measures include Accuracy (Acc), Recall, Precision, Auc, and F1-Score. Accuracy was prioritized since sets are now balanced.

Naïve Bayes

Both - Multinomial NB doesn't work over variables with negative values, wasn't used. GaussianNB outperformed BernoulliNB, assuming a normal distribution for features, aligning with the dataset's numeric variables. BernoulliNB struggled due to its binary assumption.

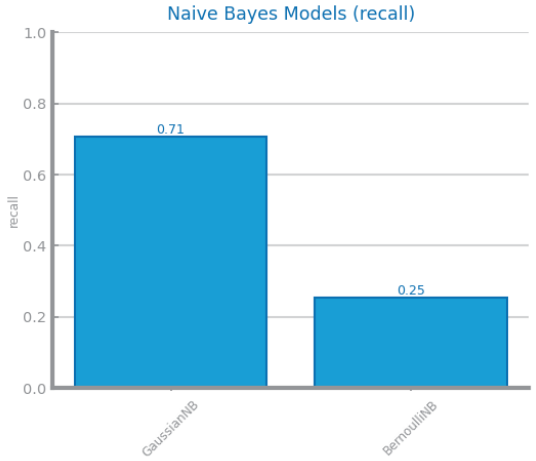

Figure 35: Naïve Bayes alternative comparison for dataset 1

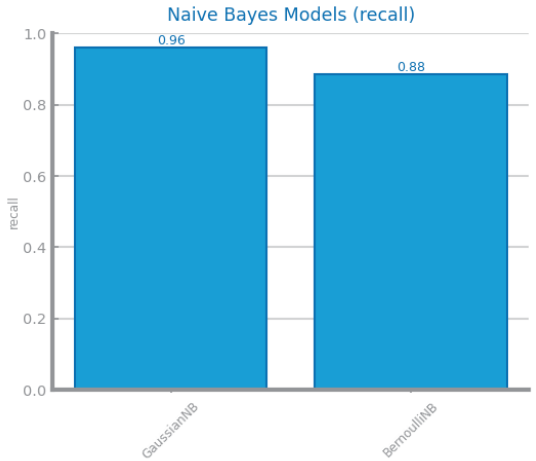

Figure 36: Naïve Bayes alternative comparison for dataset 2

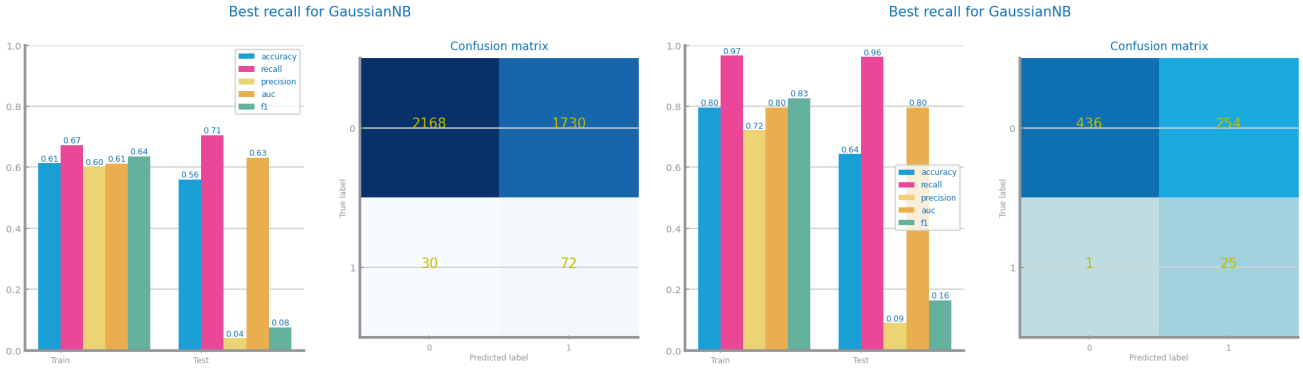

Figure 37: Naïve Bayes best model results for dataset 1 (left) and dataset 2 (right)

KNN

Set 1 - All distances have similar high accuracy. The best model (k=1, Manhattan) got high test ACC and Recall, but got a low precision, meaning that it has a problem with FP; **Set 2** - Manhattan and euclidean distance show similar robust Acc across k, outperforming Chebyshev, which struggles with feature sensitivity. The best model (k=1, Manhattan) achieves high test Acc but suffers from low recall and precision (problem with FP). No overfitting.

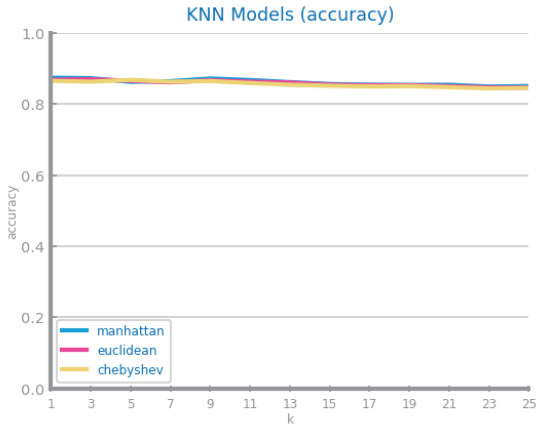

Figure 38: KNN different parameterisations comparison for dataset 1

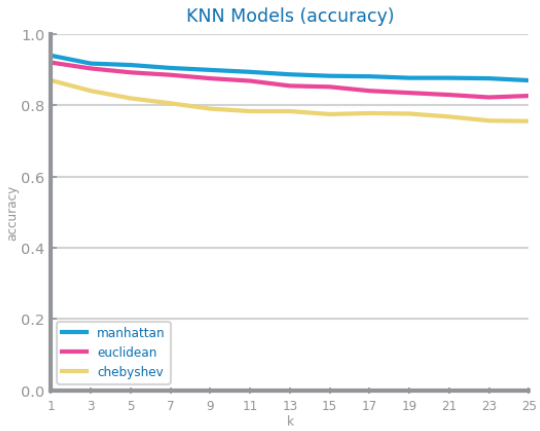

Figure 39: KNN different parameterisations comparison for dataset 2

Figure 40: KNN overfitting analysis for dataset 1 (left) and dataset 2 (right)

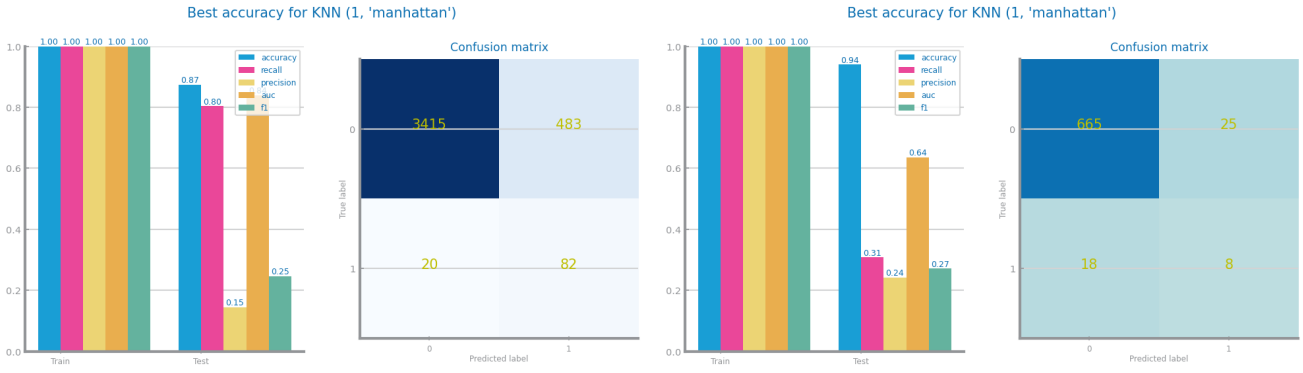

Figure 41: KNN best model results for dataset 1 (left) and dataset 2 (right)

Decision Trees

Set 1 - The best model is (depth 10, 'gini'). The most important feature is AGE_GROUP, at the top of the tree, and then PD_CD, PERP_RACE and X_COORD_CD have some importance as well; **Set 2** - The best model is (depth

18, 'entropy'). Key feature is x_{36} and then branches to two important predictors, x_{10}, x_{14}; **Both** - Acc levels of both methods are high and similar, but overfitting is present. Some more important features are below less ones as those might still create the best split at a parent node.

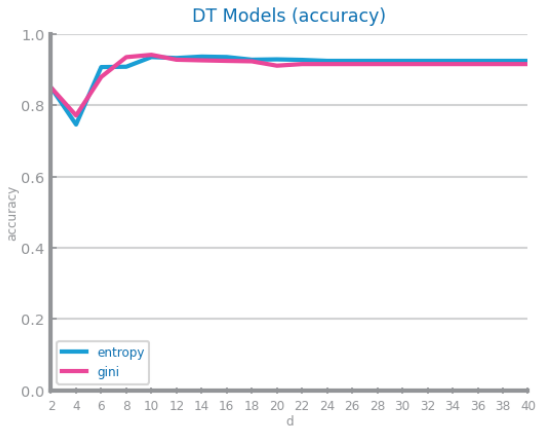

Figure 42: Decision Trees different parameterisations comparison for dataset 1

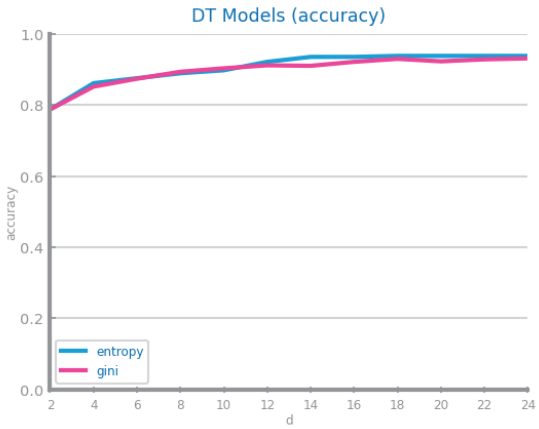

Figure 43: Decision Trees different parameterisations comparison for dataset 2

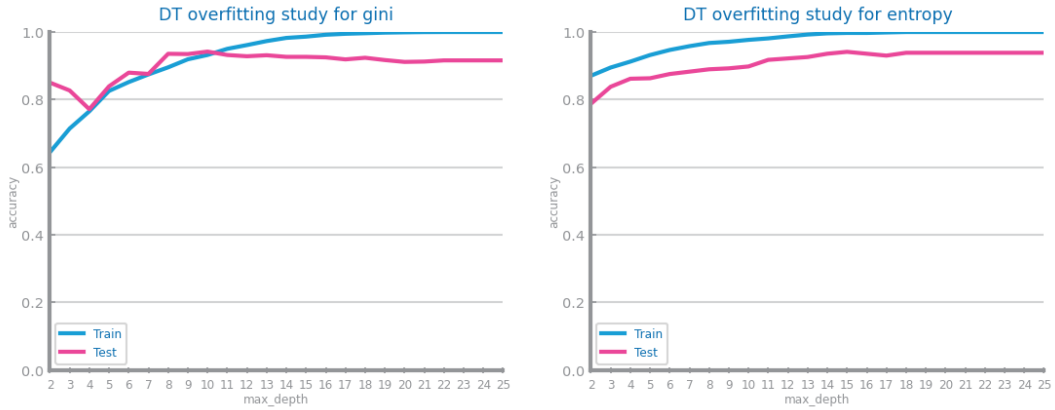

Figure 44: Decision Trees overfitting analysis for dataset 1 (left) and dataset 2 (right)

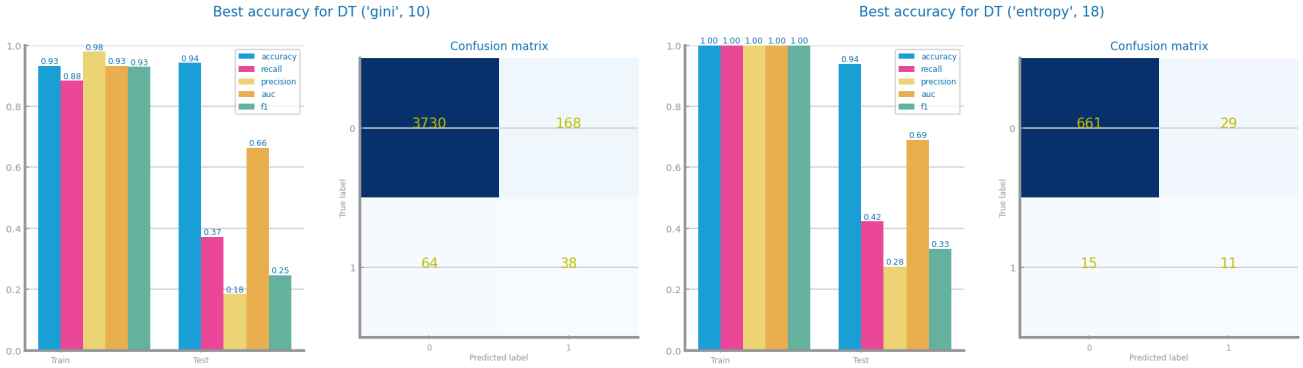

Figure 45: Decision trees best model results for dataset 1 (left) and dataset 2 (right)

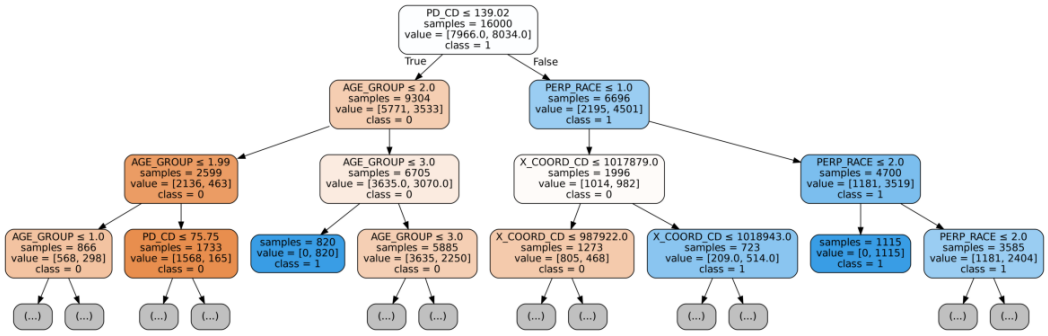

Figure 46: Best tree for dataset 1

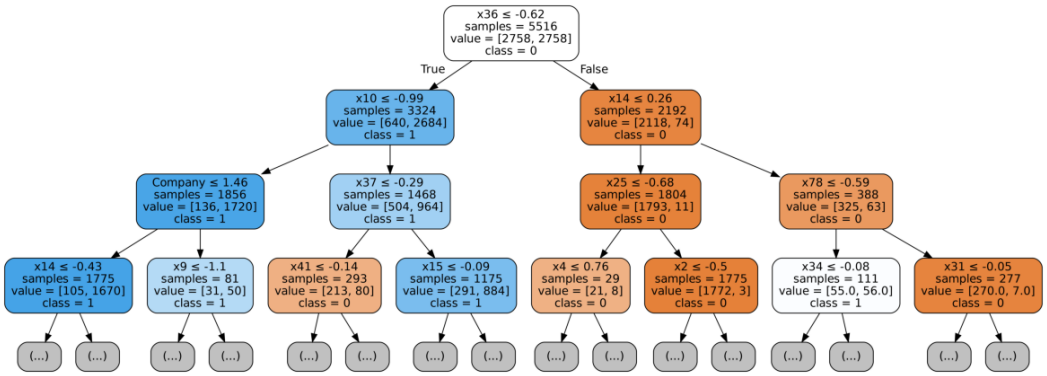

Figure 47: Best tree for dataset 2

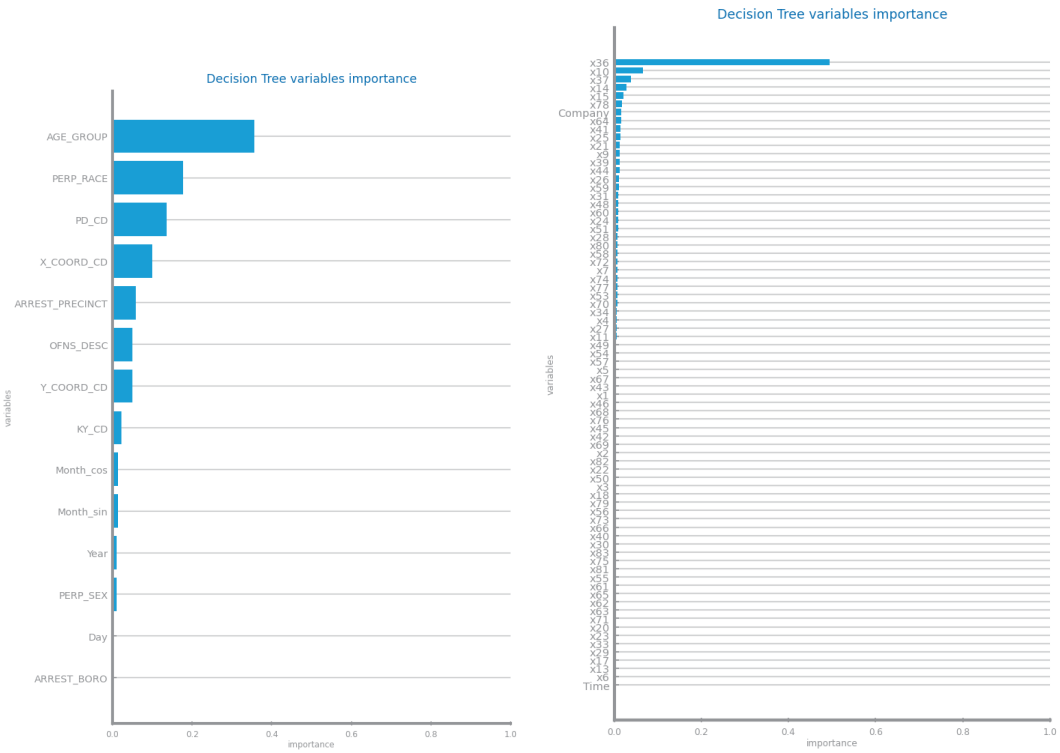

Figure 48: Decision tree variables importance for dataset 1 (left) and dataset 2 (right)

Random Forests

Set 1 - Just like for DT, in RF the most important variable is AGE_GROUP. Struggles with FN visible with recall; **Set 2** - Similar important variables with DT: x_{36}, x_{10}, x_{25}; **Both** - No overfitting, since the Acc increases with nr of estimators until it plateaus, making it clear that the models generalize the data well. Order of variable importance changes from DT to RF as in RF, in each split, a random subset of features is considered rather than all features (line in DT).

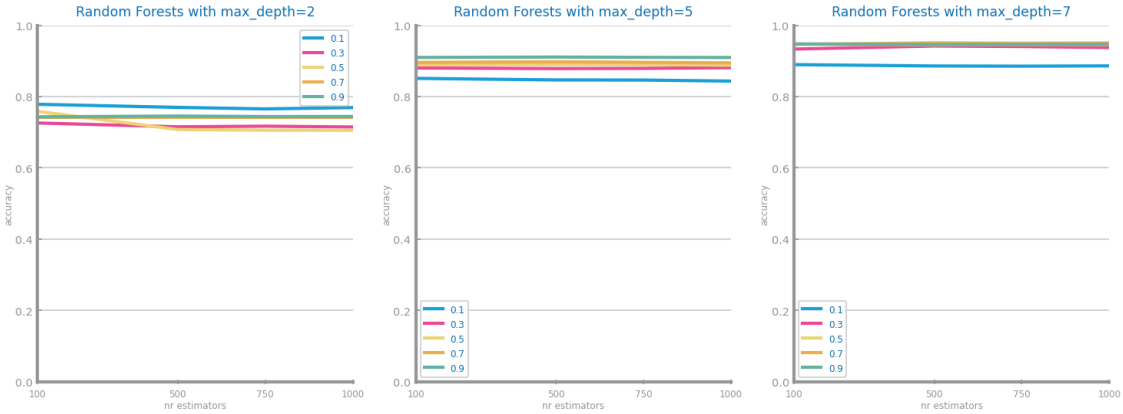

Figure 49: Random Forests different parameterisations comparison for dataset 1

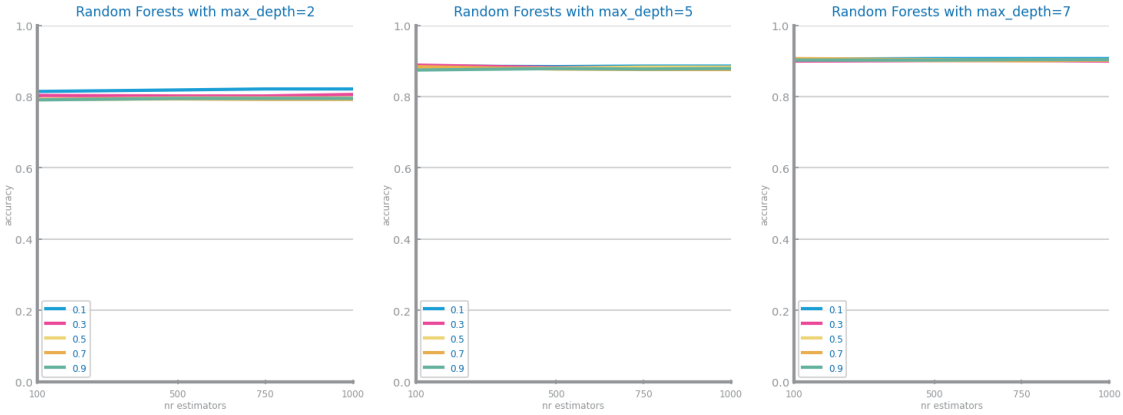

Figure 50: Random Forests different parameterisations comparison for dataset 2

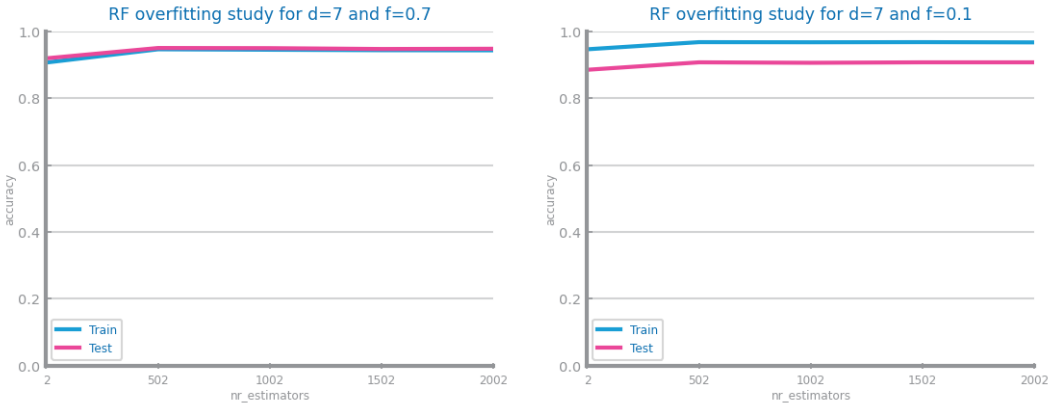

Figure 51: Random Forests overfitting analysis for dataset 1 (left) and dataset 2 (right)

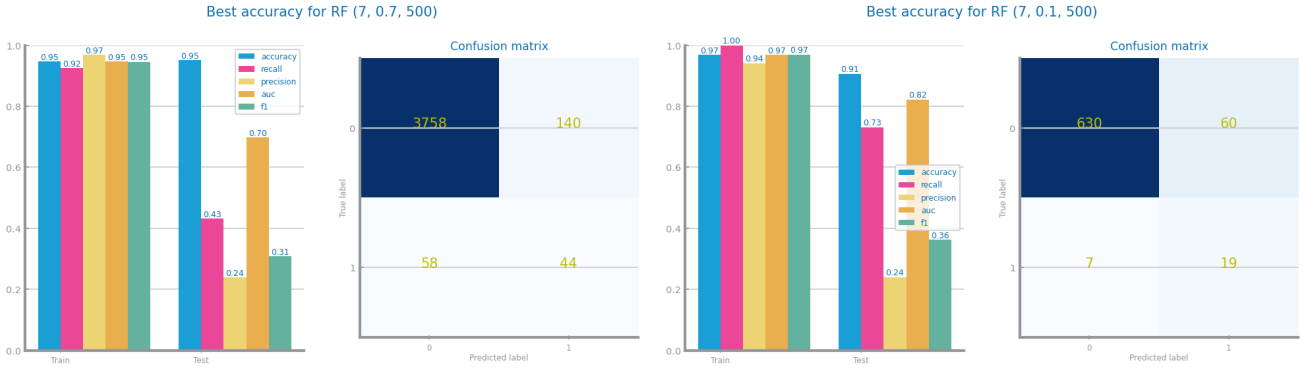

Figure 52: Random Forests best model results for dataset 1 (left) and dataset 2 (right)

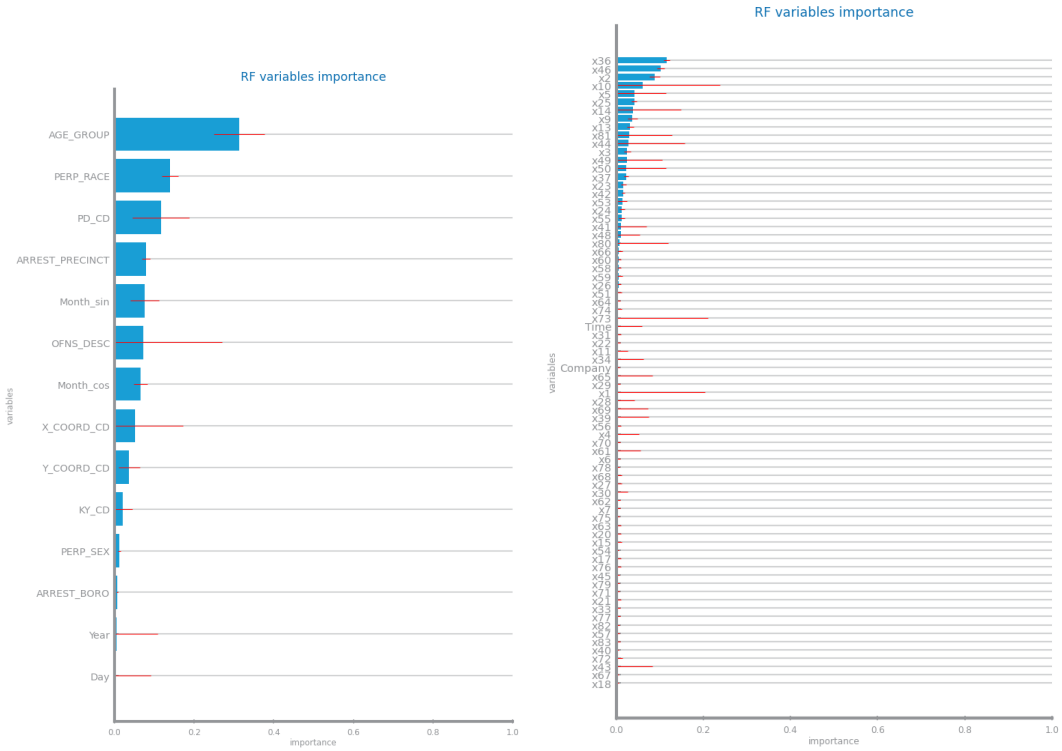

Figure 53: Random Forests variables importance for dataset 1 (left) and dataset 2 (right)

Gradient Boosting

Set 1 - Overfit is present as test Acc starts higher than train Acc. MONTH_SIN and MONTH_COS gained more relevance than previously, AGE_GROUP kept being the most important; **Set 2** - No overfit, as the Acc's have the same trend. x_{36} still is the most important variable along with x_{10} and x_{14}; **Both** - Acc is high and depth doesn't have a significant impact since the Acc values are maintained. The best depths suggest meaningful feature interactions. 500 trees suggest high error correction.

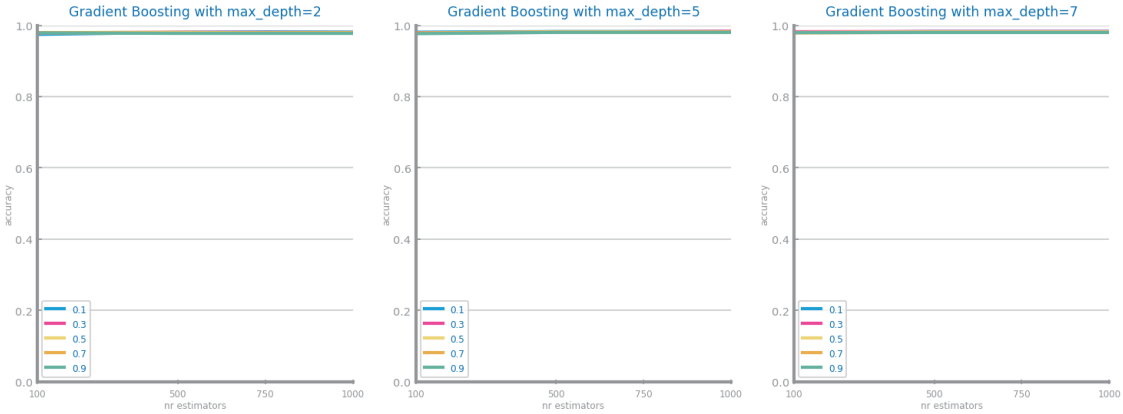

Figure 54: Gradient boosting different parameterisations comparison for dataset 1

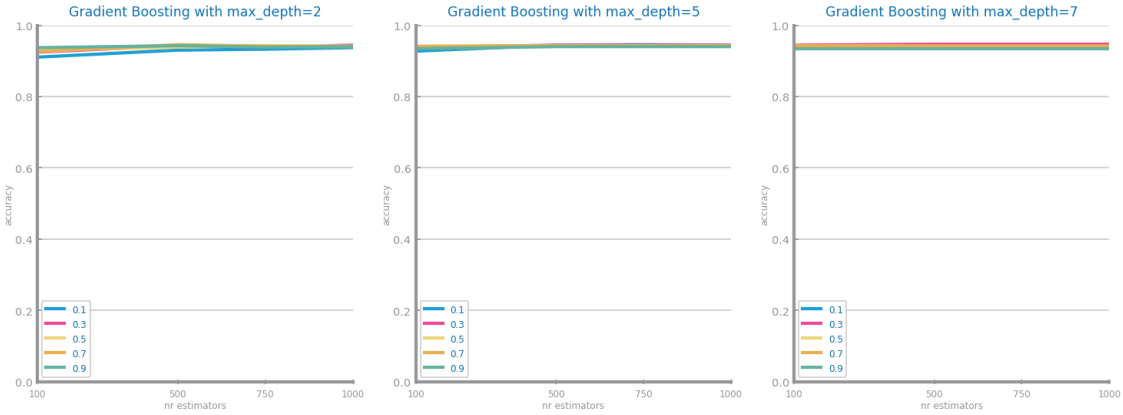

Figure 55: Gradient boosting different parameterisations comparison for dataset 2

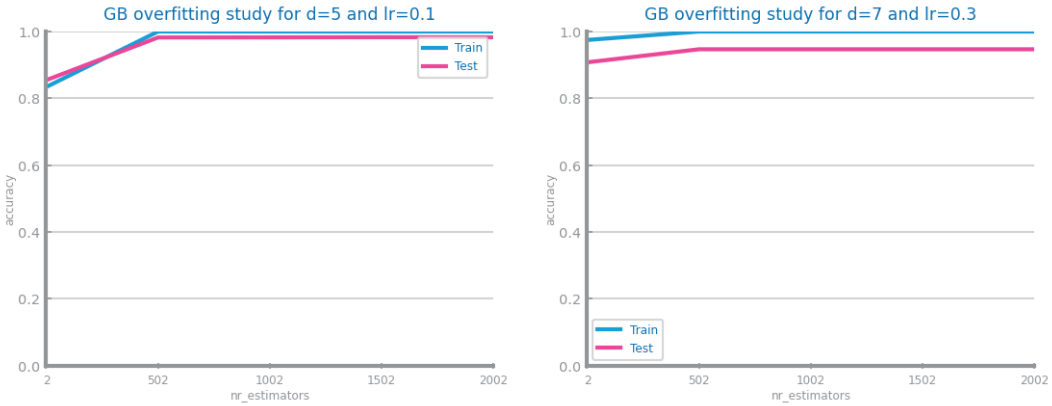

Figure 56: Gradient boosting overfitting analysis for dataset 1 (left) and dataset 2 (right)

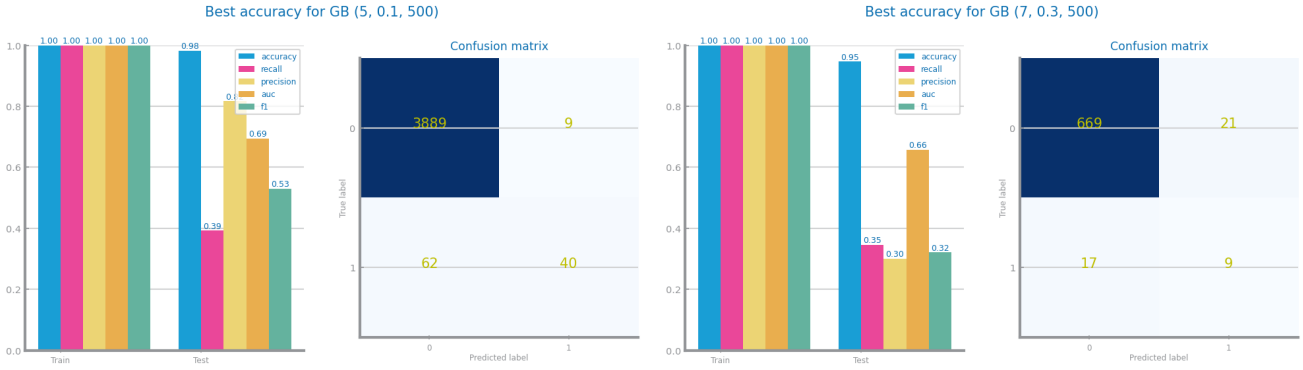

Figure 57: Gradient boosting best model results for dataset 1 (left) and dataset 2 (right)

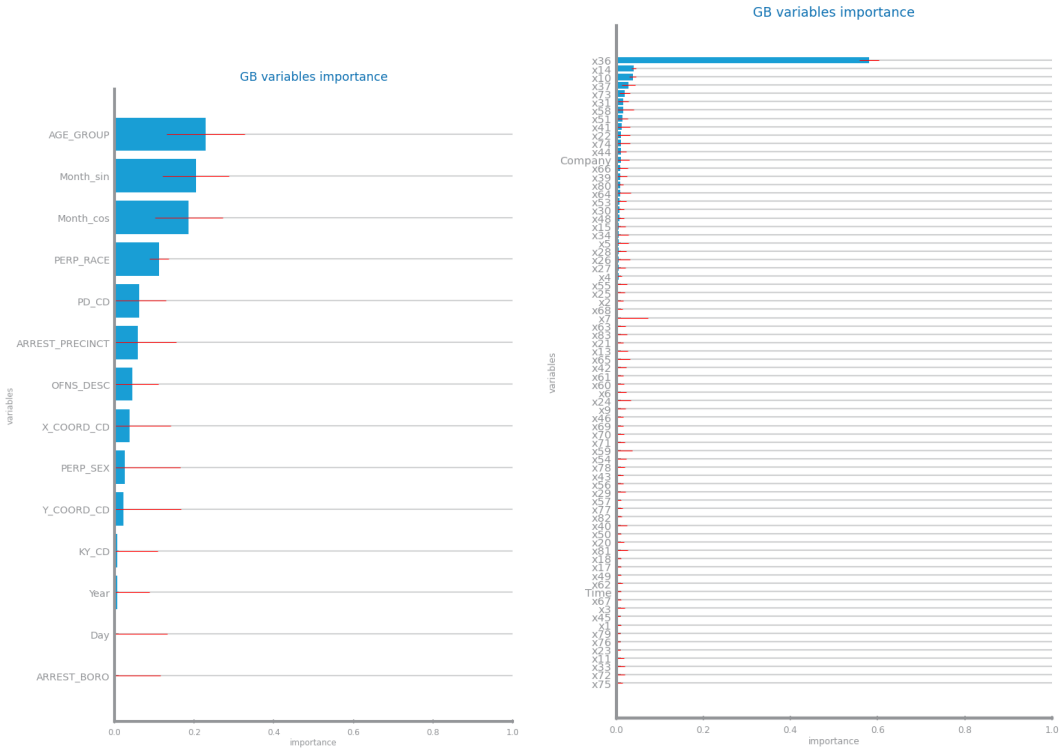

Figure 58: Gradient boosting variables importance for dataset 1 (left) and dataset 2 (right)

Multi-Layer Perceptrons

Set 1 - Fluctuating Acc values depending on the number of iterations, constantly going from 0 to 100 and back to 0. Overfitting is occurring. This might be due to not applying scaling, since the MLP is sensitive to non-normalized data. Hard time when predicting label 1; **Set 2** - Constant and adaptive have similar and superior performance compared to invscalling. No overfitting is seen. Consistency in metrics showing good Auc, being able to distinguish both classes well.

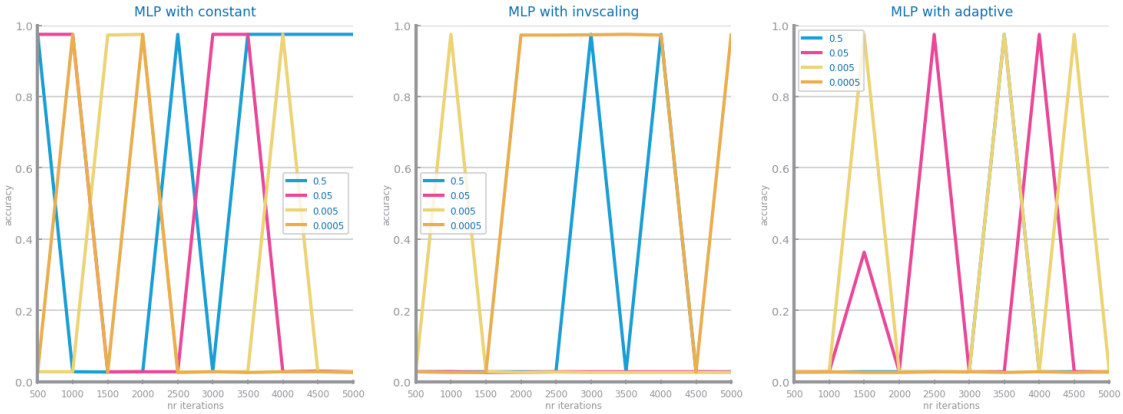

Figure 59: MLP different parameterisations comparison for dataset 1

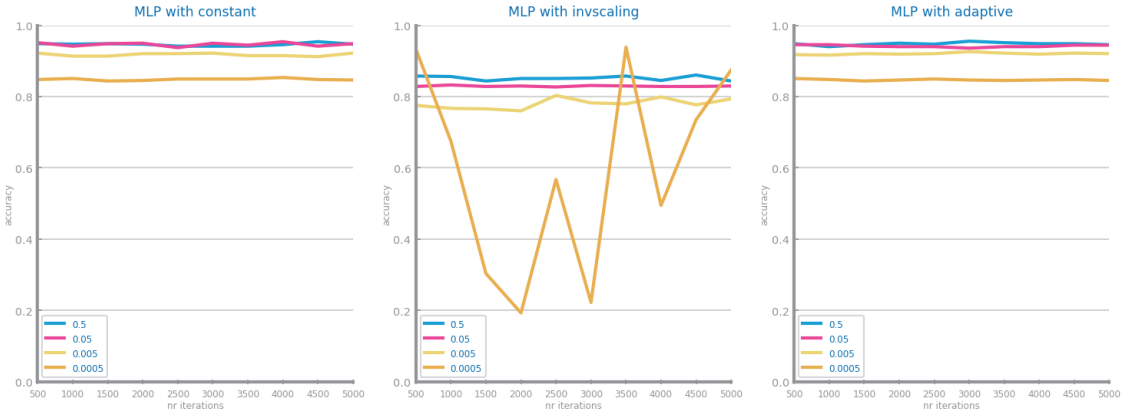

Figure 60: MLP different parameterisations comparison for dataset 2

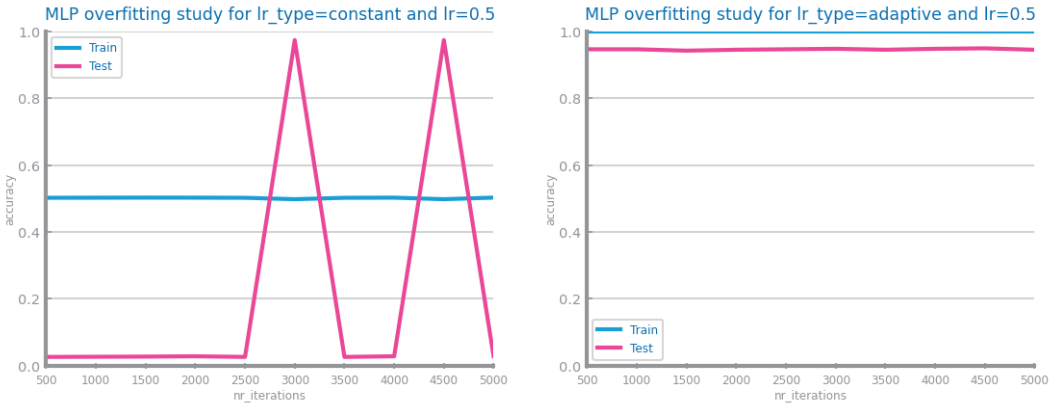

Figure 61: MLP overfitting analysis for dataset 1 (left) and dataset 2 (right)

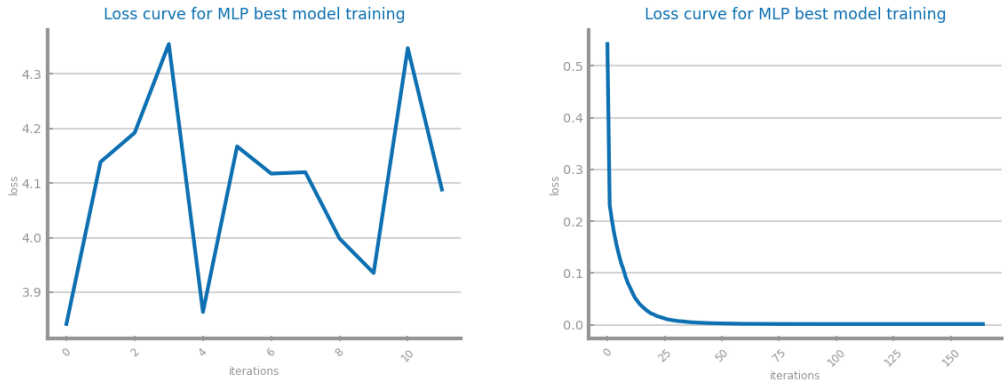

Figure 62: Loss curve analysis for dataset 1 (left) and dataset 2 (right)

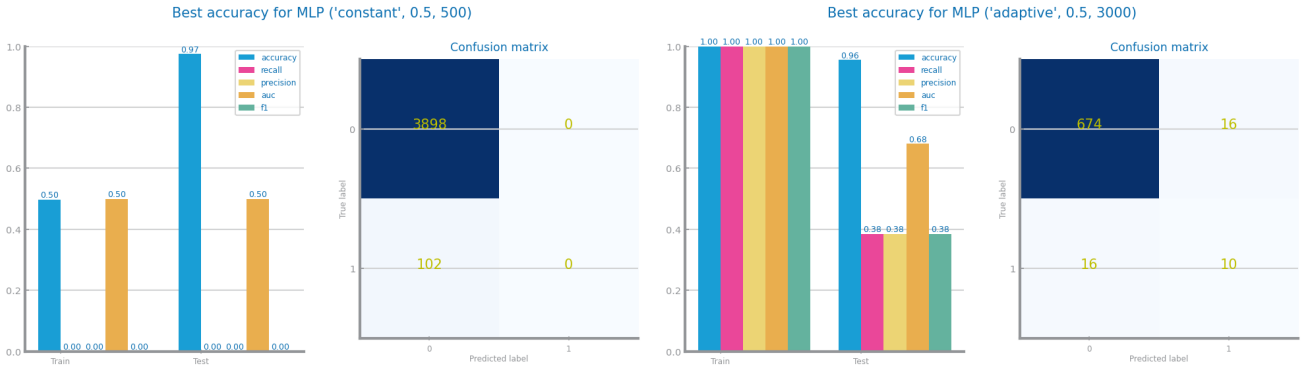

Figure 63: MLP best model results for dataset 1 (left) and dataset 2 (right)

4 CRITICAL ANALYSIS

Set 1 - Dropping missing values and applying SMOTE improved minority-class detection. The sample of 1% reduced overfitting but still left the MLP vulnerable without scaling. GaussianNB leveraged normal features well but sometimes over-predicted positives. KNN performed strongly overall yet struggled with low precision for minority classes when k=1. Decision Trees, although quick to train, showed clear overfitting at greater depths - still, AGE_GROUP, PD_CD, and PERP_RACE emerged as frequent, high-impact features (as other models). Random Forest generally outperformed other models, with bootstrapped trees mitigating noise while capturing important splits. Gradient Boosting achieved high accuracy but revealed mild overfitting when hyperparameters were not carefully tuned; **Set 2** - The larger number of numeric features demanded thorough feature selection and balancing. Dropping outliers aided NB and KNN, though it hurt recall, reflecting potential losses in valuable data points. Standard scaling was crucial for MLP performance, eliminating the instability seen with unscaled inputs. Here too, Random Forest excelled, showcasing robust train-test accuracy across depths and features, often prioritizing x_{36}, x_{10}, x_{14} (as other models). Gradient Boosting, when deeper trees were employed, approached Random Forest's accuracy without the overfitting seen at shallow depths. Overall, strong correlation and numeric distributions favored tree-based ensembles; **Both** - Crucially, Random Forest stands out in both sets for delivering consistent, high accuracy, balanced variance, and straightforward feature importance interpretation. These models are sufficiently strong for the problem at hand, though future work could refine hyperparameters or employ more sophisticated ensemble methods if domain constraints require even higher minority-class recall or precision.

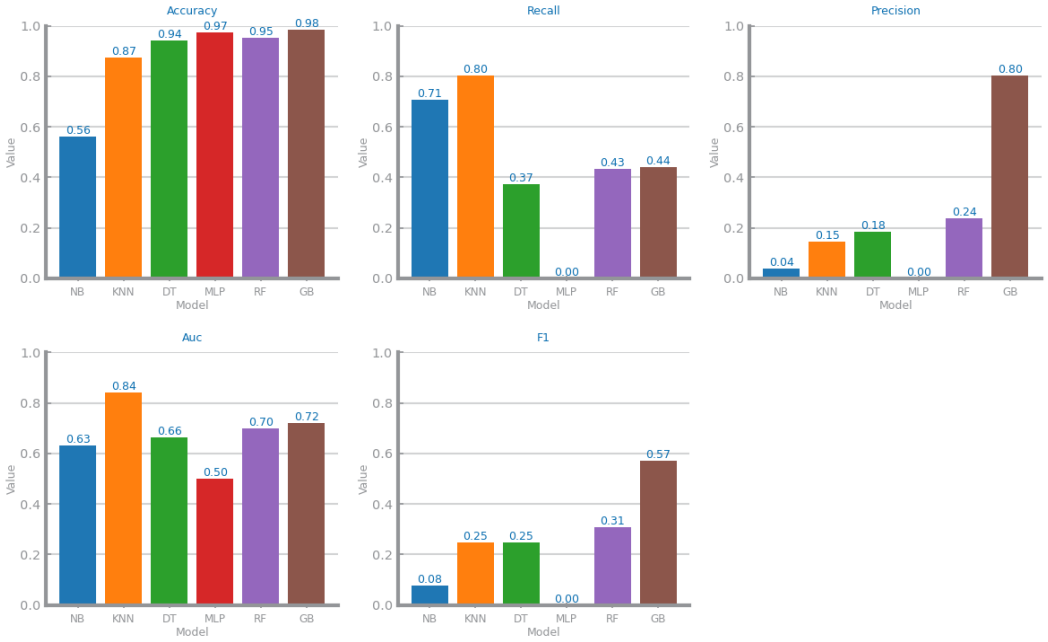

Figure 64: Models' evaluation per metric for dataset 1

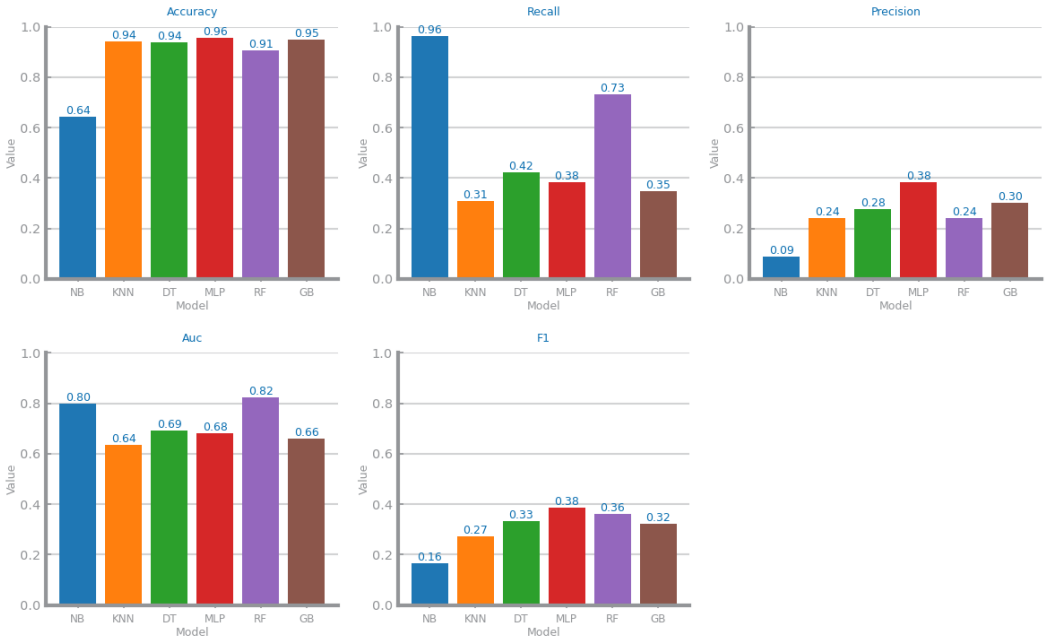

Figure 65: Models' evaluation per metric for dataset 2

TIME SERIES ANALYSIS

5 DATA PROFILING

Data Dimensionality and Granularity

Set 1 - Finest granularity is daily. Chose weekly and monthly for the second and third granularity, respectively. Daily granularity is not very readable but weekly and monthly might lose some important details; **Set 2** - Yearly is the finest granularity. 5 and 10-year aggregation (agg) were also chosen. Although 10-year intervals lose details, 5-year intervals capture them well without sacrificing much information.

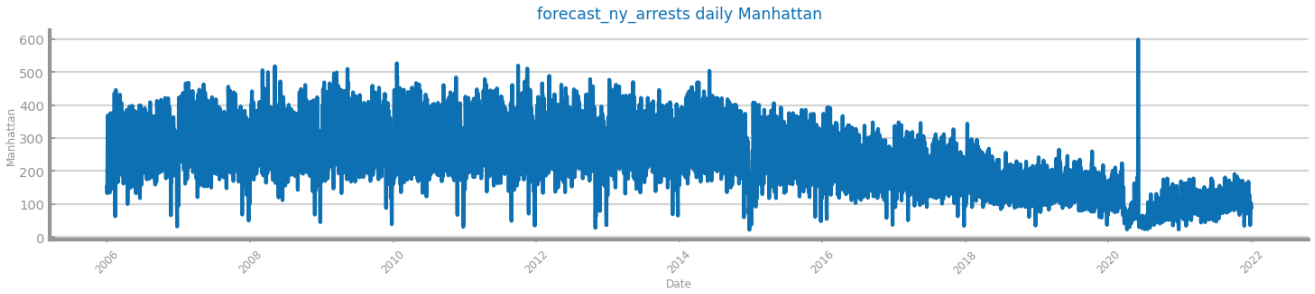

Figure 66: Time series 1 at the most granular detail

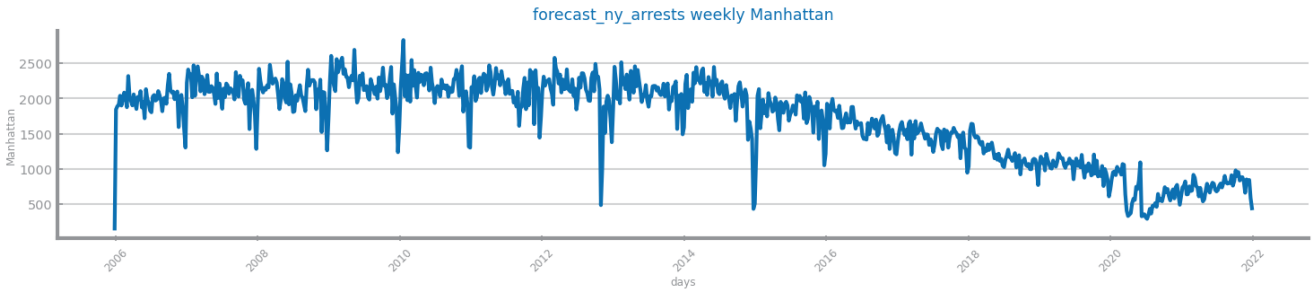

Figure 67: Time series 1 at the second chosen granularity

Figure 68: Time series 1 at the third chosen granularity

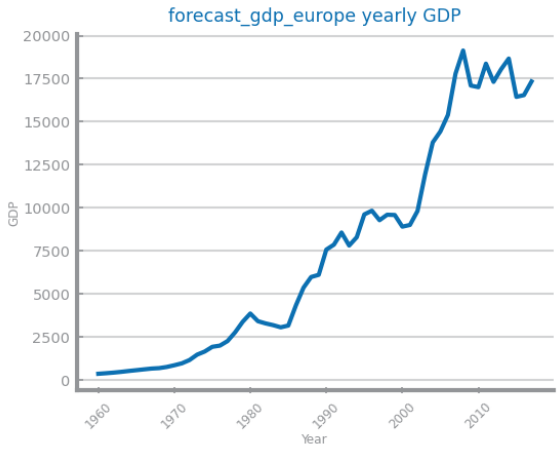

Figure 69: Time series 2 at the most granular detail

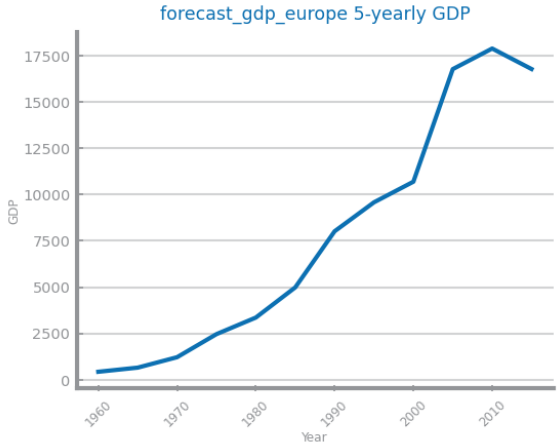

Figure 70: Time series 2 at the second chosen granularity

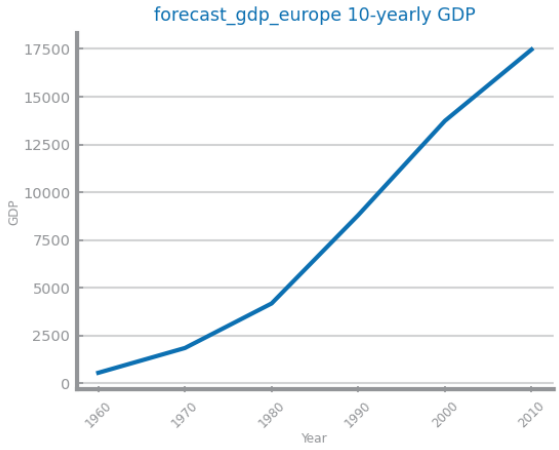

Figure 71: Time series 2 at the third chosen granularity

Data Distribution

Set 1 - Has some outliers. Weekly and monthly distributions show a negative skew while the daily chart shows a small positive skew. Lag plots and autocorrelation plots suggest both a trending and seasonal series; **Set 2** - Daily histogram resembles a normal distribution while other agg show skewness. Lag plots show high autocorrelation which might indicate ARIMA will perform well. The slow decay in the correlogram indicates non-stationarity, due to the TS trend.

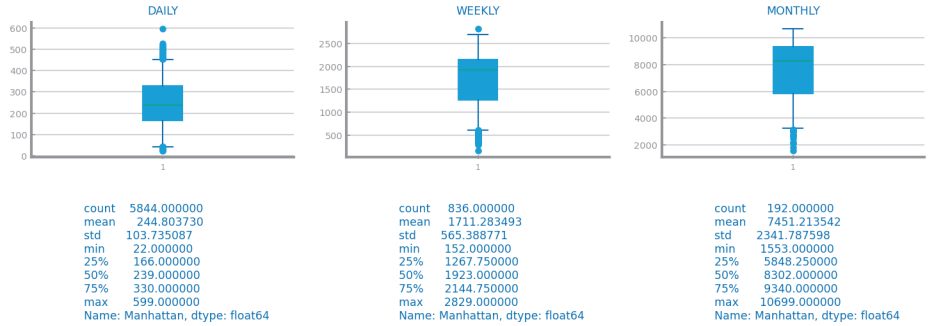

Figure 72: Boxplot(s) for time series 1

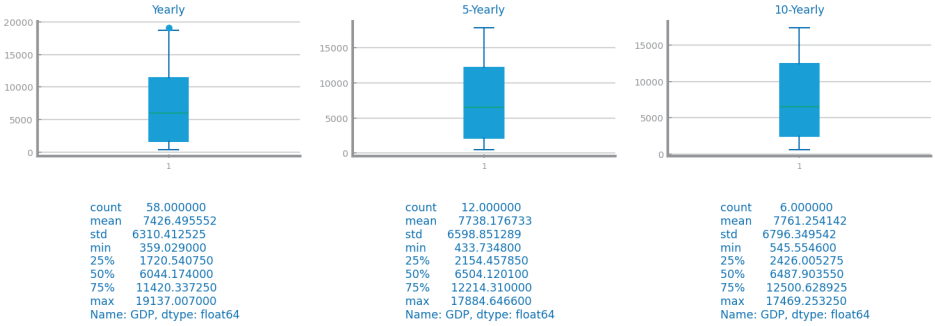

Figure 73: Boxplot(s) for time series 2

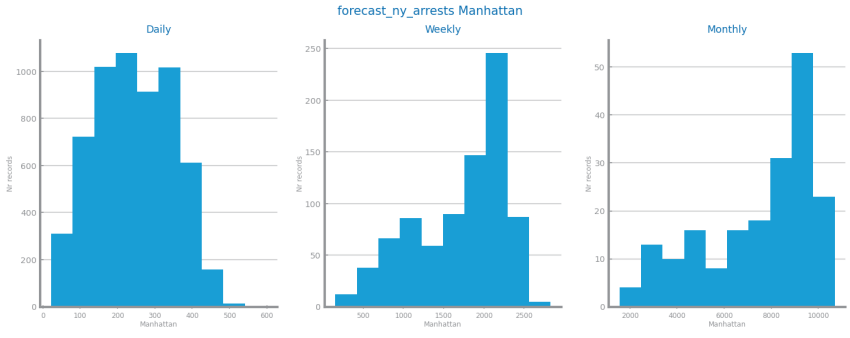

Figure 74: Histogram(s) for time series 1

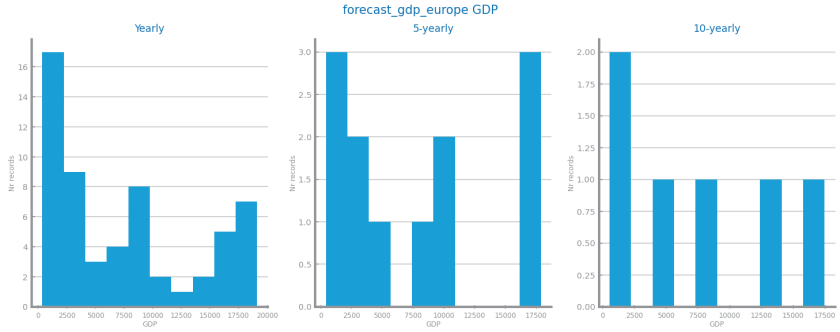

Figure 75: Histogram(s) for time series 2

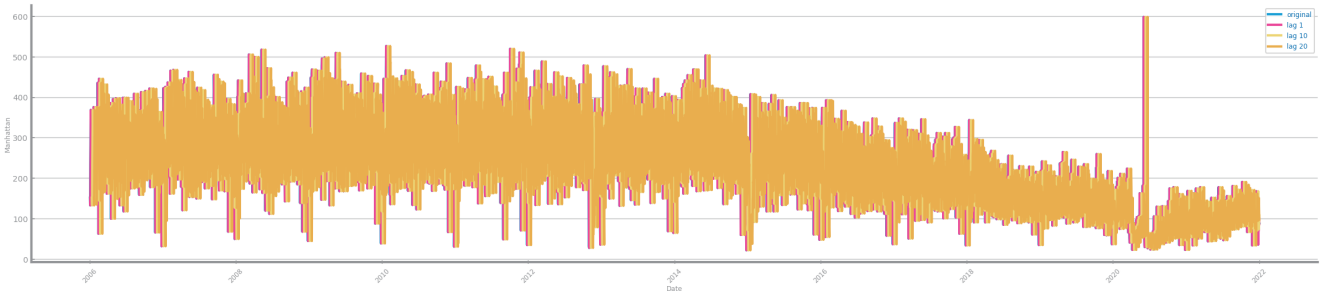

Figure 76: Autocorrelation lag-plots for original time series 1

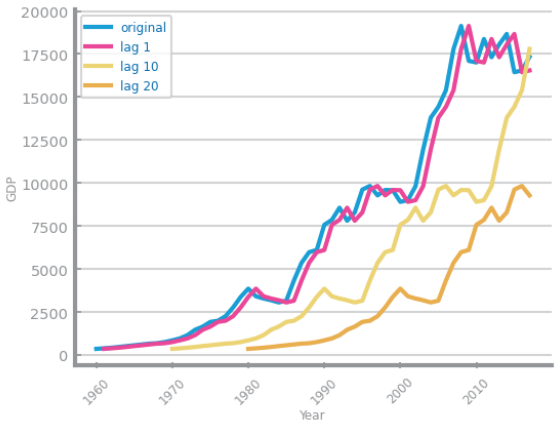

Figure 77: Autocorrelation lag-plots for original time series 2

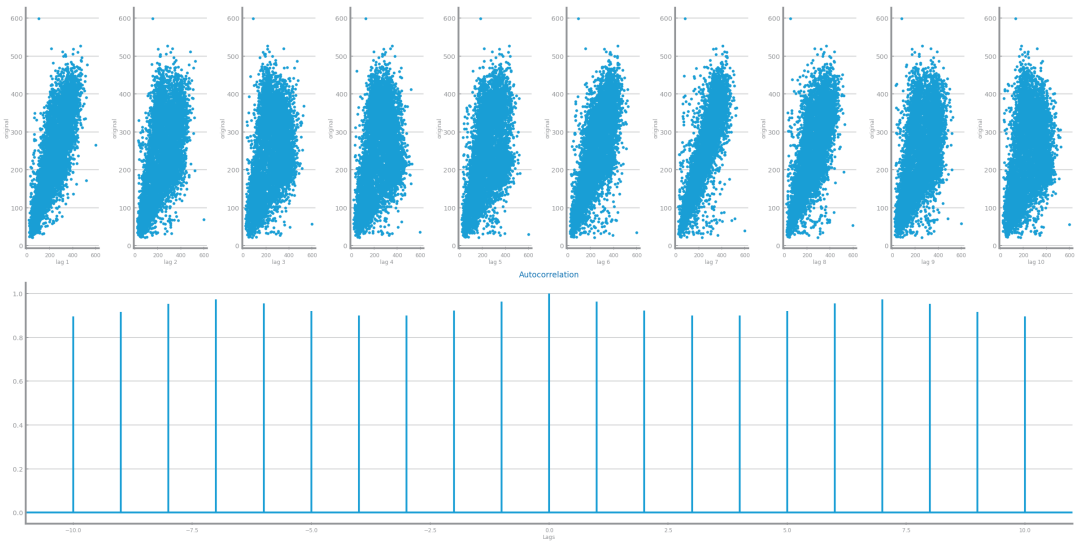

Figure 78: Autocorrelation correlogram for original time series 1

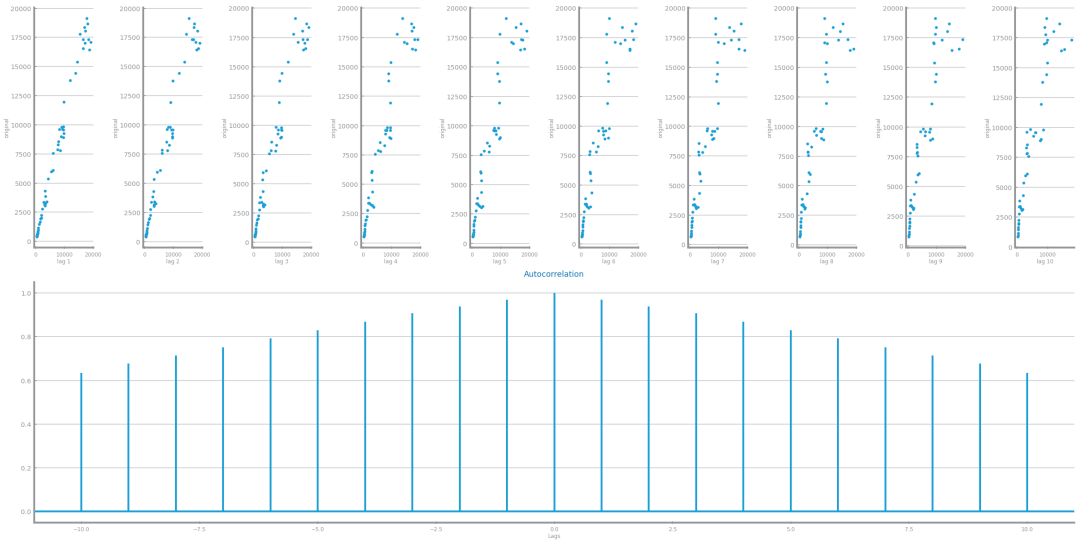

Figure 79: Autocorrelation correlogram for original time series 2

Data Stationarity

Set 1 - Manhattan arrests are non-stationary (p=0.338>0.05) with a slight downward trend. It also shows a seasonal effect; **Set 2** - Annual GDP is non-stationary (p=0.869>0.05). Strong upward trend, no seasonal effect, and inconsistent mean/variance over time.

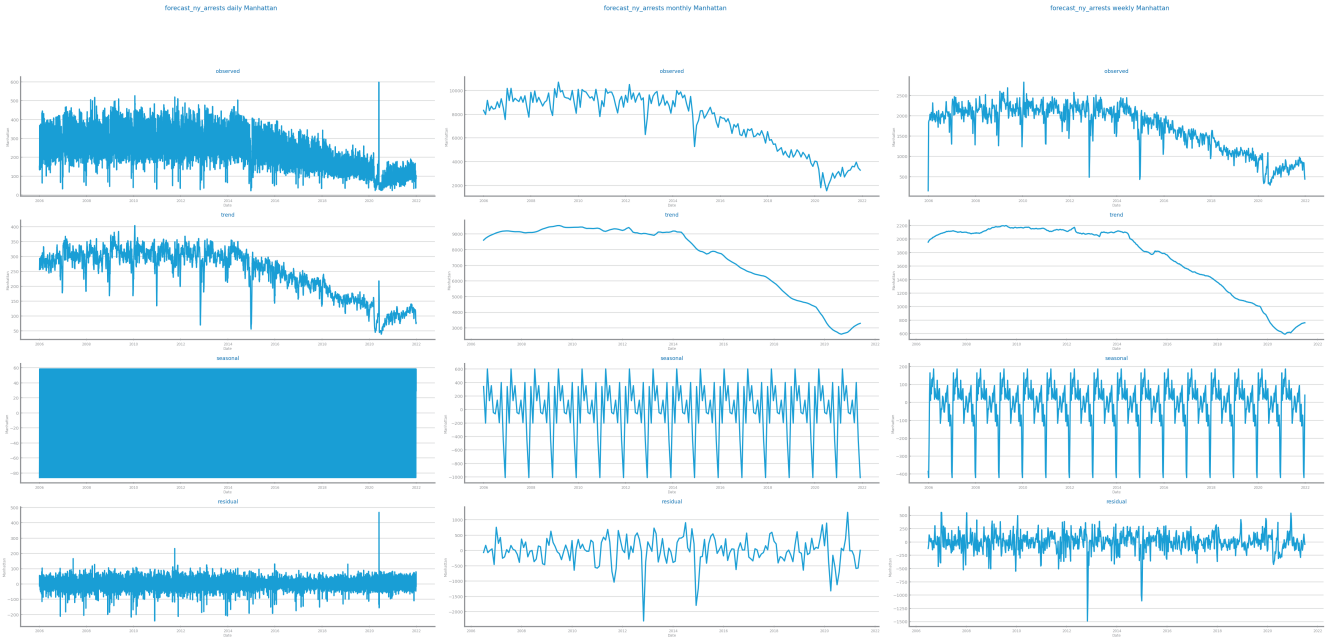

Figure 80: Components study for time series 1

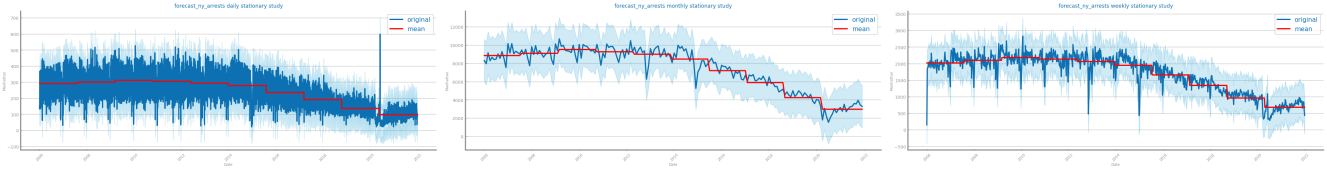

Figure 81: Stationarity study for time series 1

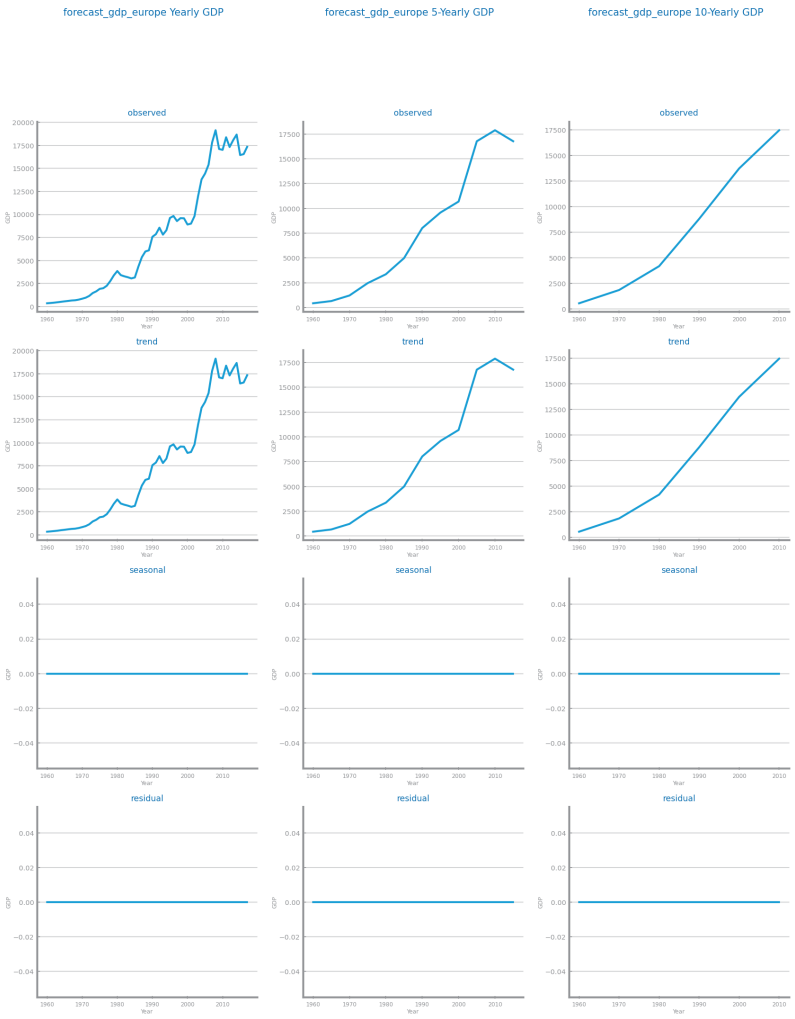

Figure 82: Components study for time series 2

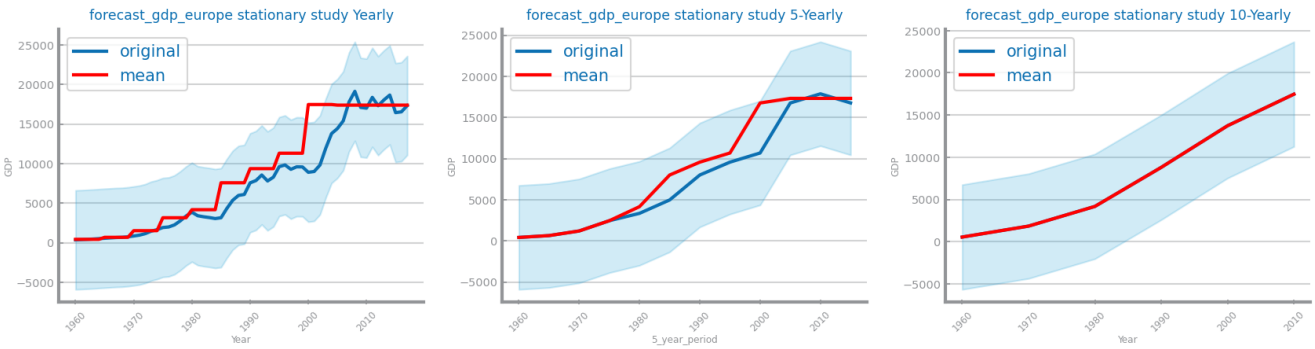

Figure 83: Stationarity study for time series 2

6 DATA TRANSFORMATION

Aggregation

Set 1 - Daily agg was chosen as it has lower R^2. Sum agg function was used; **Set 2** - 5-Year agg was chosen for higher R^2 (after doing MVI). Mean agg function was used, as the average economic performance is interesting to study.

Figure 84: Forecasting plots after different aggregations on time series 1

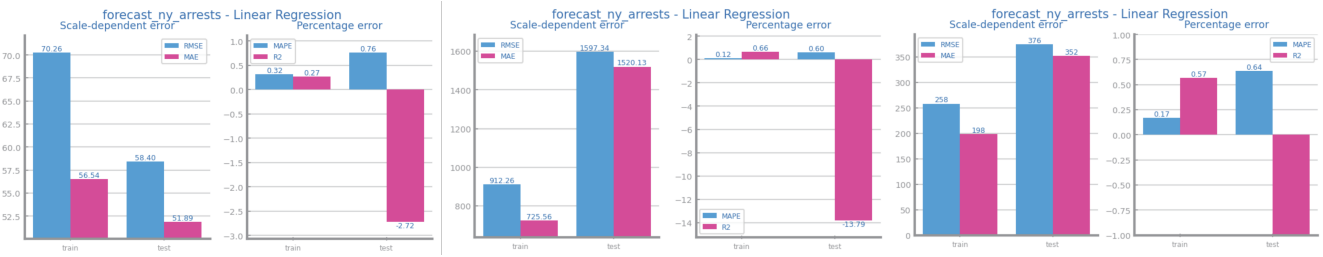

Figure 85: Forecasting results after different aggregations on time series 1

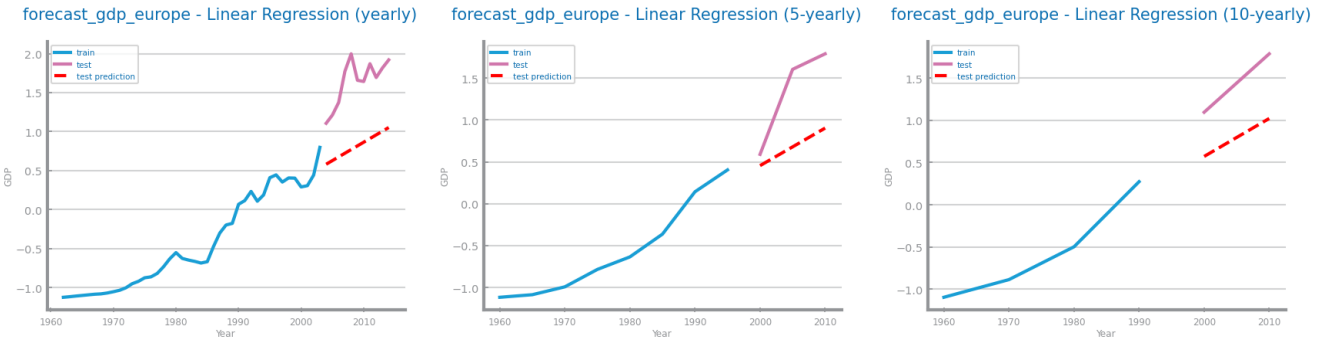

Figure 86: Forecasting plots after different aggregations on time series 2

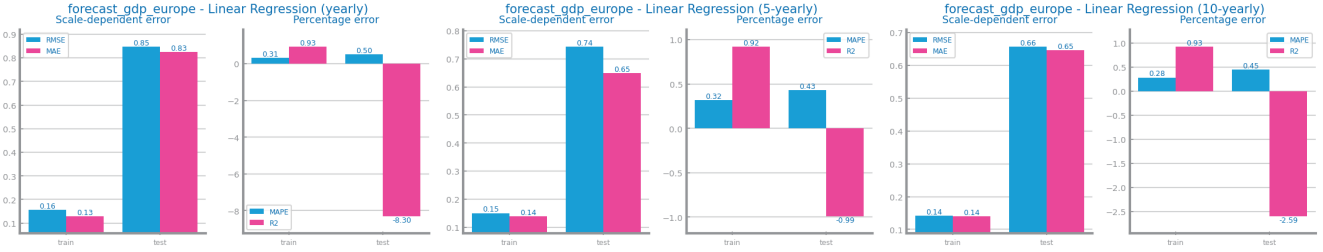

Figure 87: Forecasting results after different aggregations on time series 2

Smoothing

Set 1 - Due to visible noisy data, a smoothing size window of 4 was applied (best R^2); **Set 2** - Smoothing was studied yet not applied since 5-year agg does not show noise, therefore yielded better R^2 score.

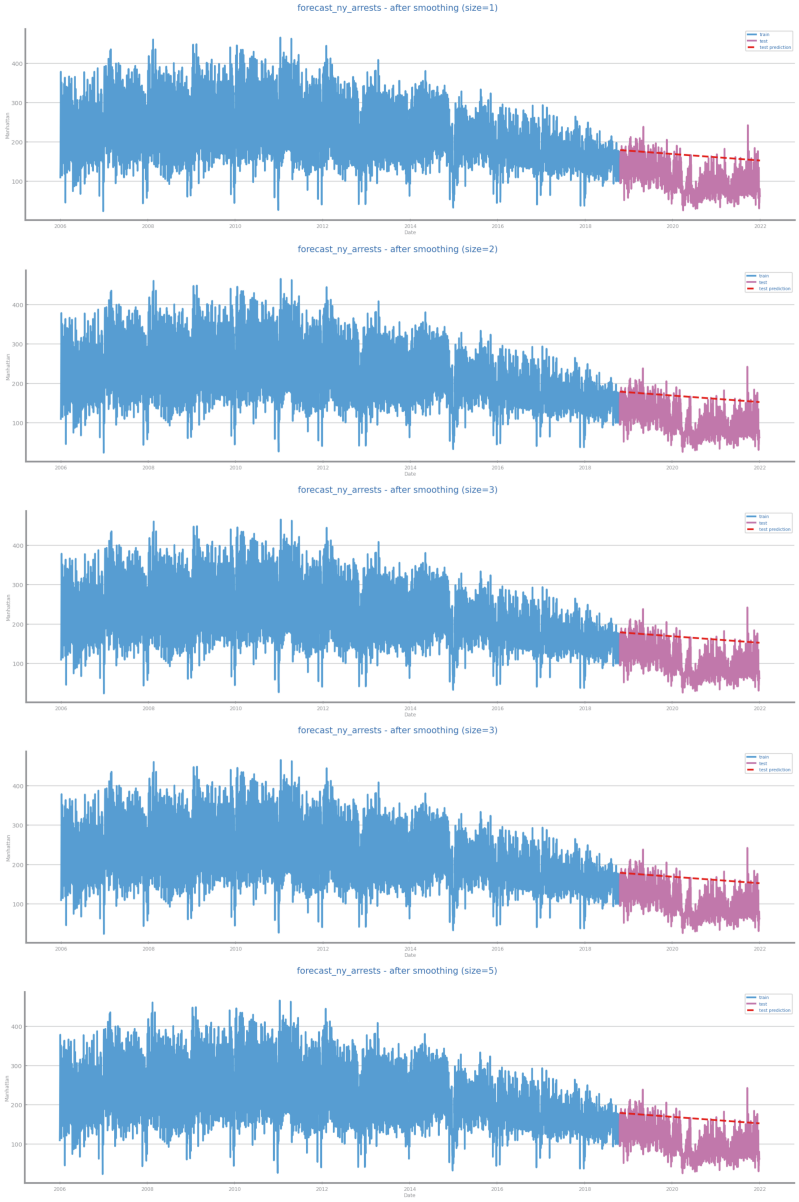

Figure 88: Forecasting plots after different smoothing on time series 1

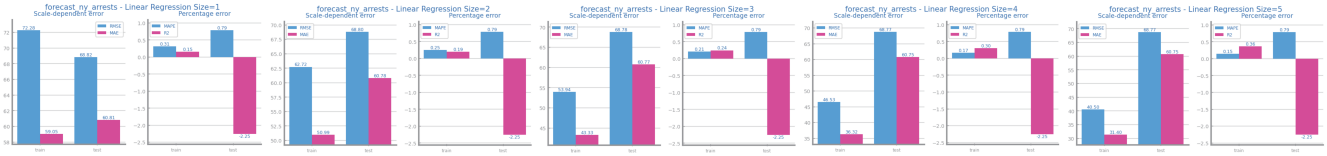

Figure 89: Forecasting results after different smoothing on time series 1

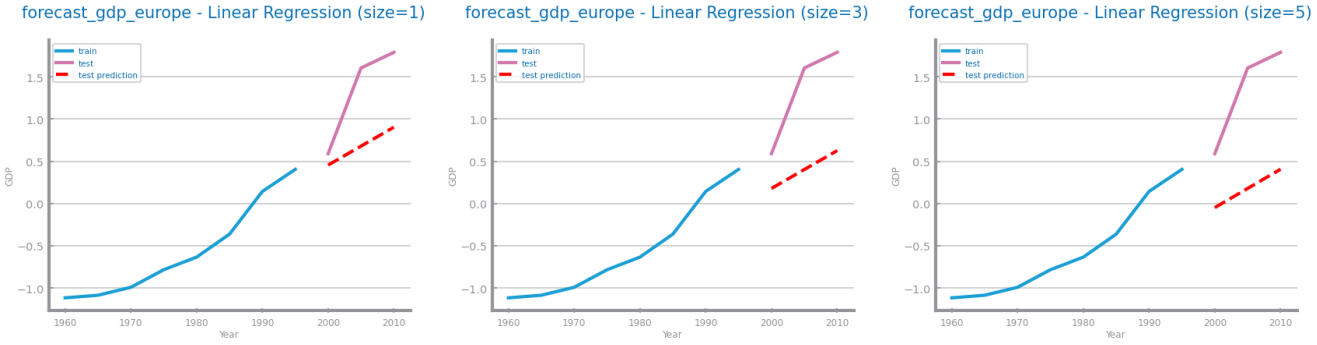

Figure 90: Forecasting plots after different smoothing on time series 2

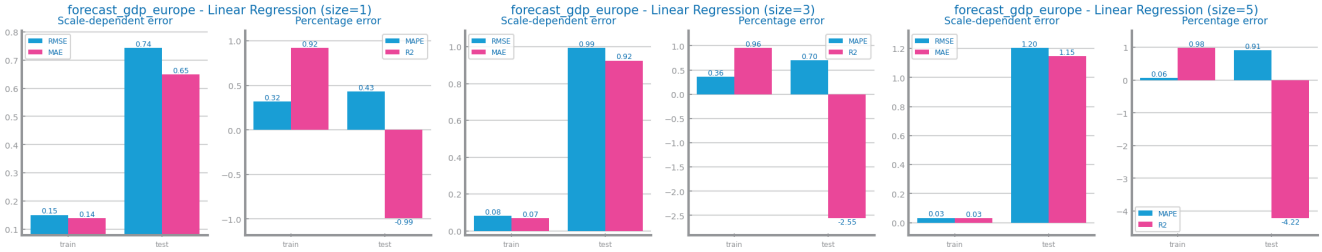

Figure 91: Forecasting results after different smoothing on time series 2

Differentiation

Both - By applying the first differentiation a better R^2 was achieved but a huge increase in MAPE is seen so it was decided not to apply first differentiation and consequently the second.

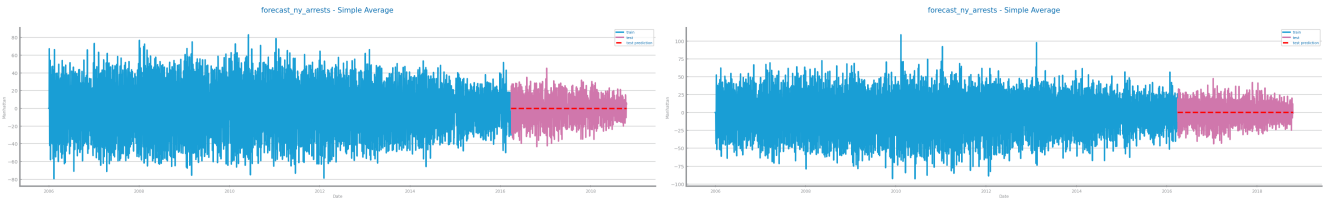

Figure 92: Forecasting plots after first and second differentiation of time series 1

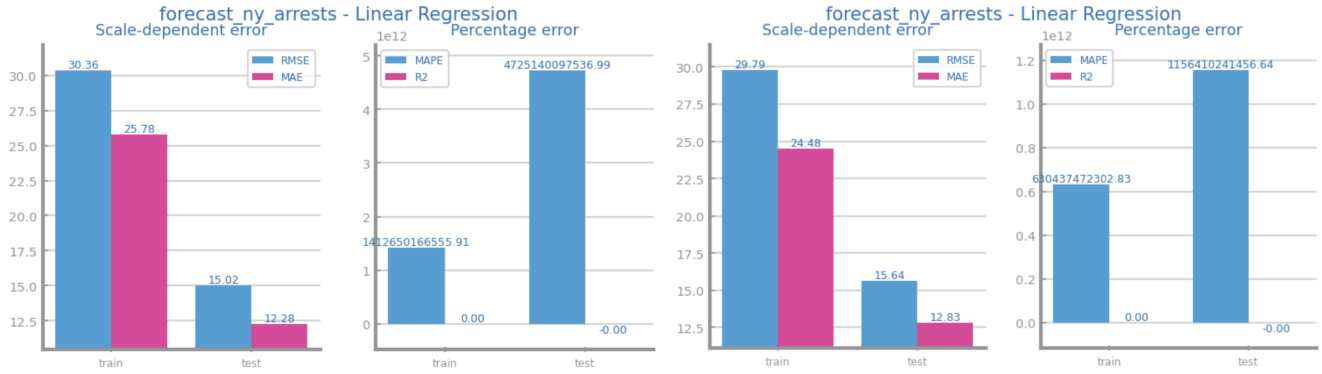

Figure 93: Forecasting results after first and second differentiation of time series 1

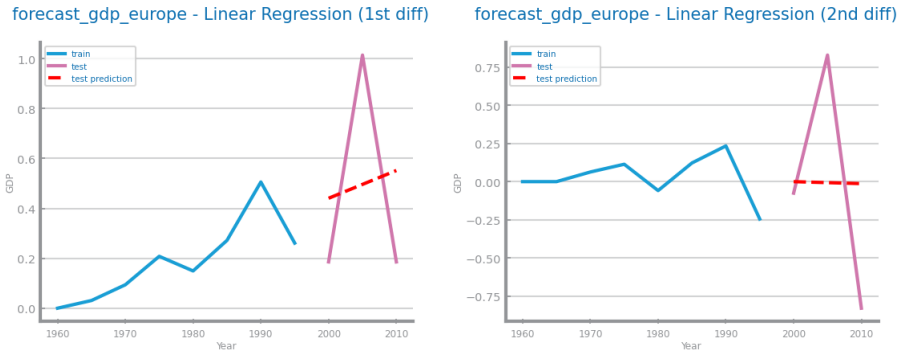

Figure 94: Forecasting plots after first and second differentiation of time series 2

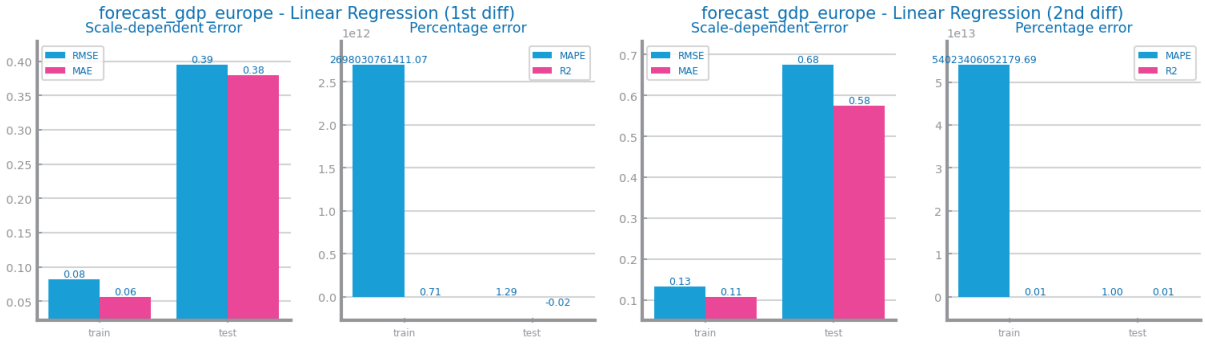

Figure 95: Forecasting results after first and second differentiation of time series 2

Other transformations (optional)

Set 2 - Shows several missing values, therefore the need of MVI. Tested drop and filling MV. The best results were with drop (5 records dropped). Scaling achieved slightly better results (after MVI) so it was applied, confining GDP to a smaller range of values which is usually useful for ML models. After these two transformations, the pipeline was followed.

Not done.

Figure 96: Forecasting plots after applying other transformations over time series 1

Not done.

Figure 97: Forecasting results after applying other transformations over time series 1

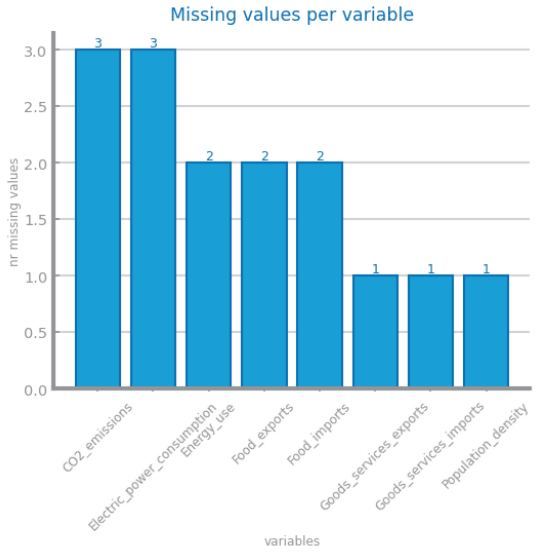

Figure 98: Nr missing values for time series 2

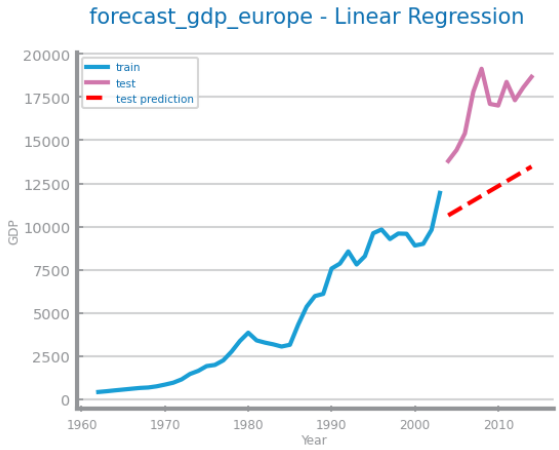

Figure 99: Forecasting plots after applying MVI (drop) over time series 2

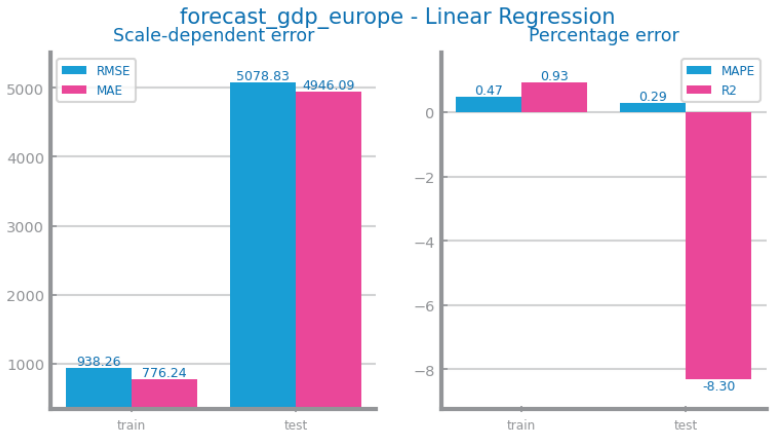

Figure 100: Forecasting results after applying MVI (drop) over time series 2

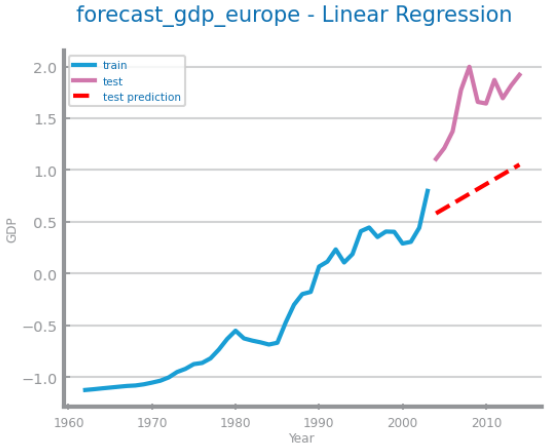

Figure 101: Forecasting plots after applying scaling over time series 2

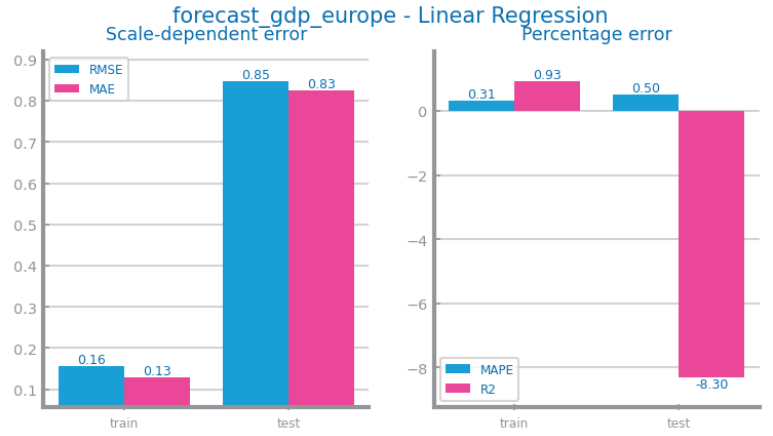

Figure 102: Forecasting results after applying scaling over time series 2

7 MODELS' EVALUATION

Set 1 - Modeling was performed over data with just smoothing transformations and daily aggregation and so there probably will be lower values of RMSE/MAE in the results; **Set 2** - Decisions made previously were reverted since problems arose in ARIMA and LSTM due to the immense reduction of the TS (due to agg and differentiation). Modeling was performed over data with transformations: MVI and Scaling; **Both** - The maximized measure will be R^2.

Simple Average Model

Both - R^2 very low as expected (compared to further used models) which means the model predictions are far worse than the mean as a predictor.

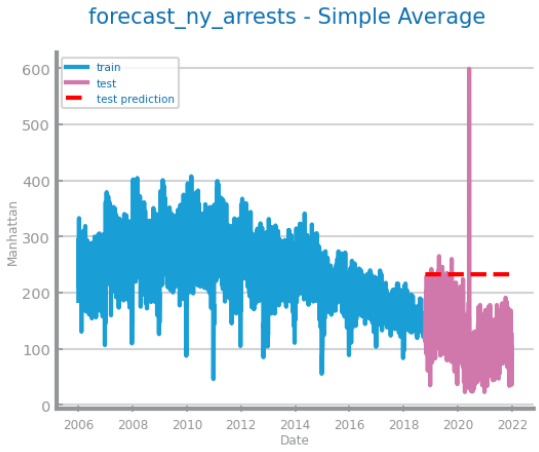

Figure 103: Forecasting plots obtained with Simple Average model over time series 1

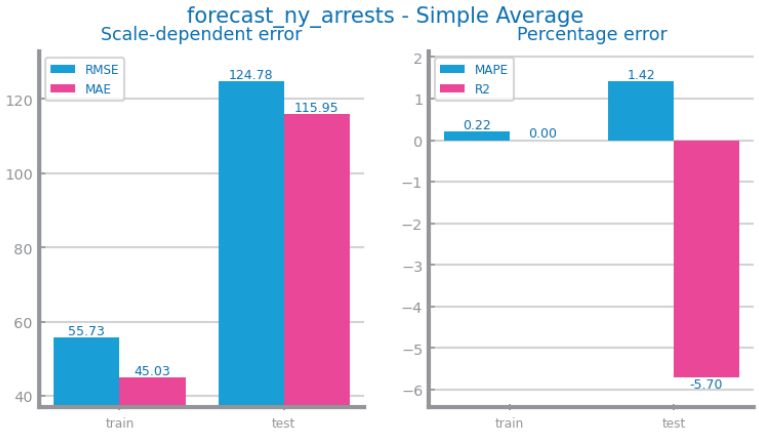

Figure 104: Forecasting results obtained with Simple Average model over time series 1

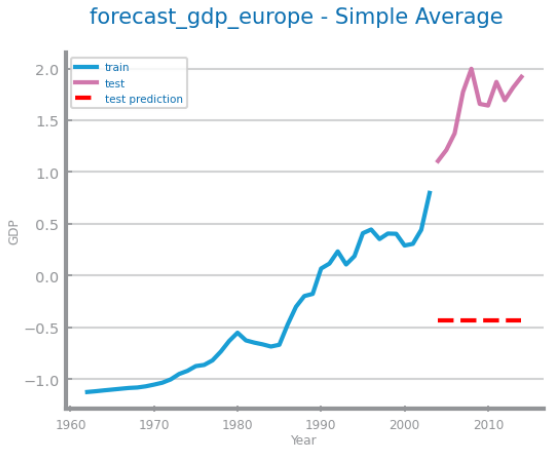

Figure 105: Forecasting plots obtained with Simple Average model over time series 2

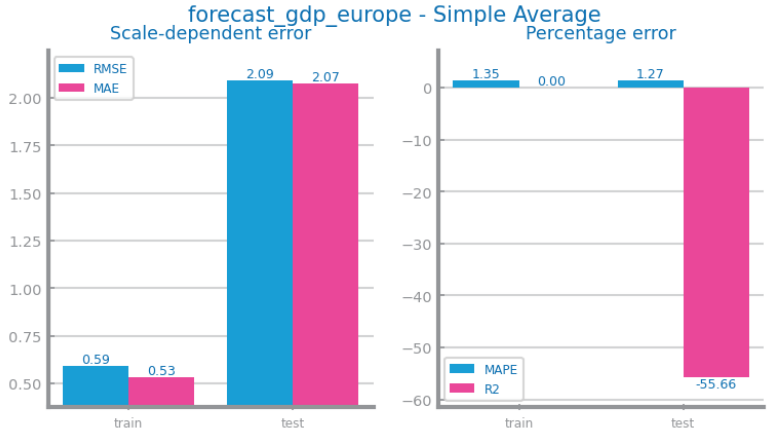

Figure 106: Forecasting results obtained with Simple Average model over time series 2

Persistence Model

Both - Although shows better performance, the optimistic is not comparable with other models used in this project. For that the realistic one is used which looks ahead in the future, just using the training dataset and the model's predictions for the rest of the time. The long term generalizes better to test data, seeing a tight fit to test data and one-set behind performs worse than the mean as a predictor.

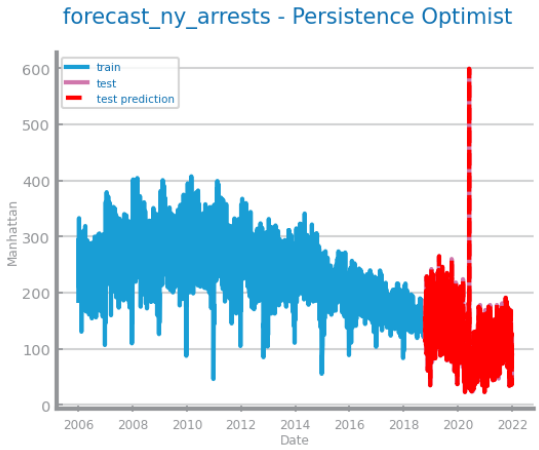

Figure 107: Forecasting plots obtained with Persistence model (long term) over time series 1

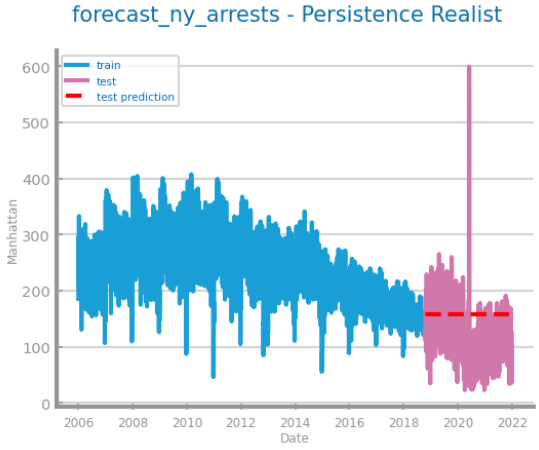

Figure 108: Forecasting plots obtained with Persistence model (one-set-behind) over time series 1

Figure 109: Forecasting results obtained with Persistence model in both situations over time series 1

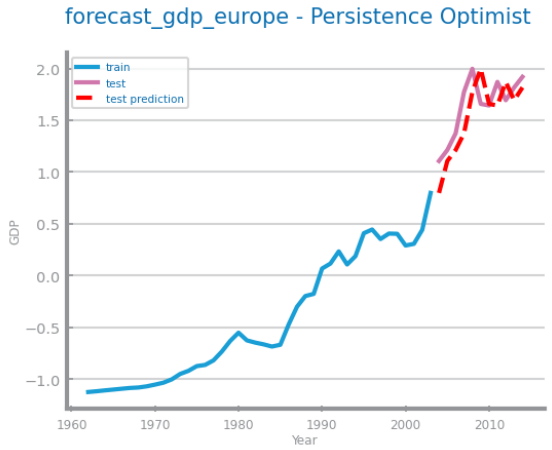

Figure 110: Forecasting plots obtained with Persistence model (long term) over time series 2

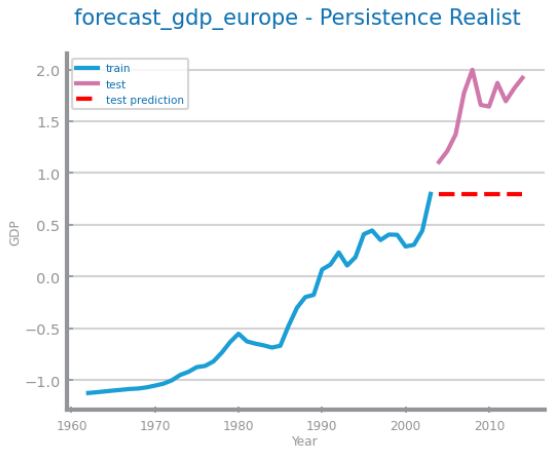

Figure 111: Forecasting plots obtained with Persistence model (one-set-behind) over time series 2

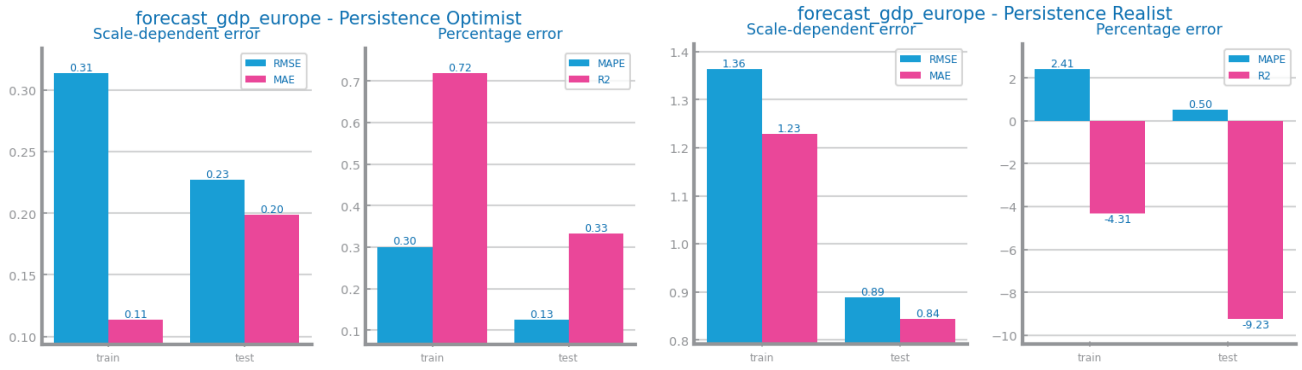

Figure 112: Forecasting results obtained with Persistence model in both situations over time series 2

Rolling Mean Model

Set 1 - Best window size is 40. It achieves lower values of RMSE/MAE and higher R^2 than simple averages, but still fails to generalize; **Set 2** - Best window size found is 1 (no smoothing as mentioned in profiling). Fails to generalize on unseen data and by smoothing out fluctuations, it ignores important dynamics, leading to poor forecast.

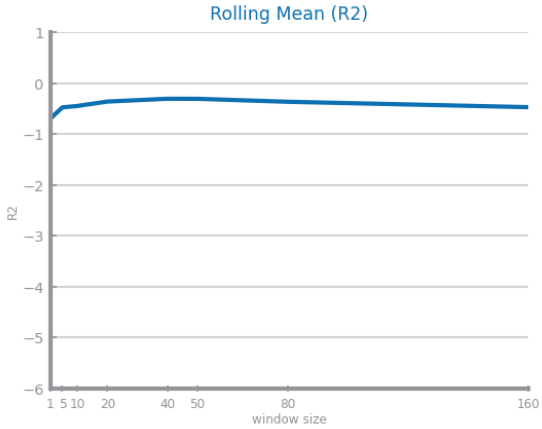

Figure 113: Forecasting study over different parameterisations of the Rolling Mean algorithm over time series 1

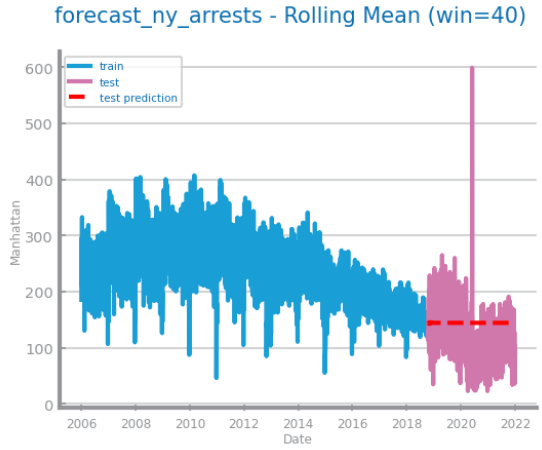

Figure 114: Forecasting plots obtained with the best parameterisation of Rolling Mean algorithm, over time series 1

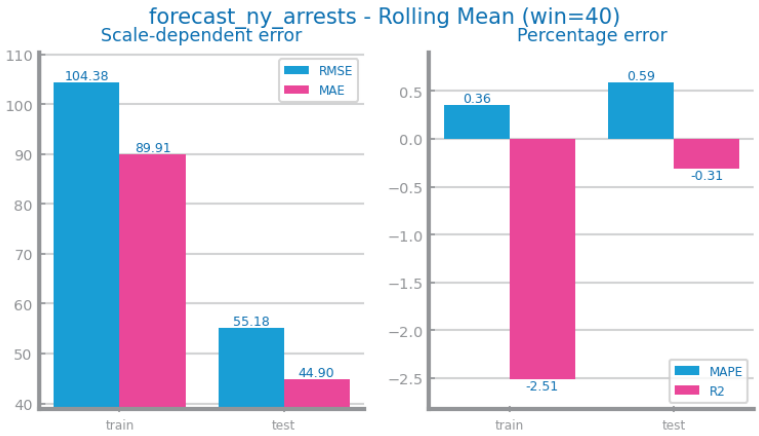

Figure 115: Forecasting results obtained with the best parameterisation of Rolling Mean algorithm, over time series 1

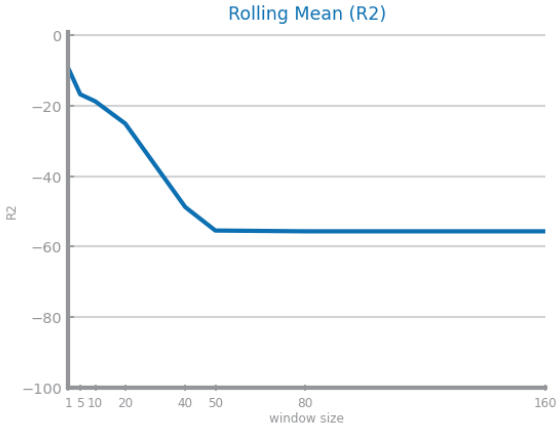

Figure 116: Forecasting study over different parameterisations of the Rolling Mean algorithm over time series 2

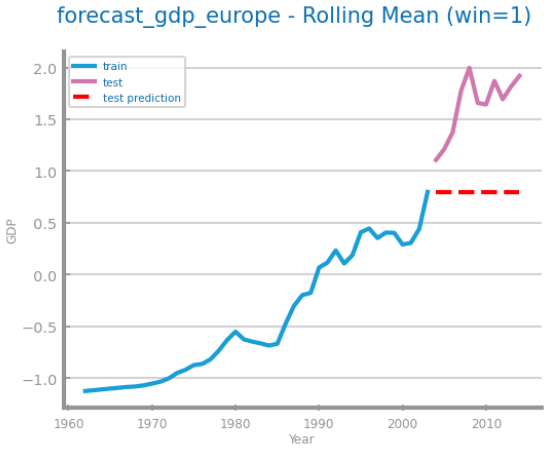

Figure 117: Forecasting plots obtained with the best parameterisation of Rolling Mean algorithm, over time series 2

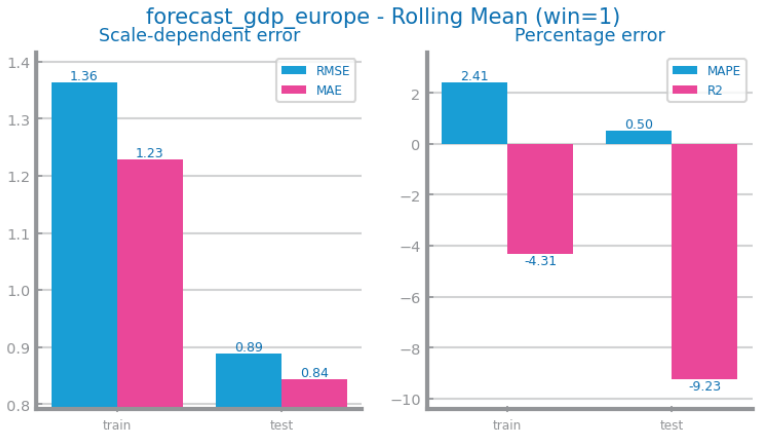

Figure 118: Forecasting results obtained with the best parameterisation of Rolling Mean algorithm, over time series 2

Exponential Smoothing Model

Set 1 - The best alpha was 0.1 giving more importance to smoothing. Has very similar error levels and R^2 values to the Rolling mean model; **Set 2** - The best alpha found was near 1, which gives less weight to smoothing (confirming analysis in transformation) and more importance to recent observations. Performs better the greater alpha is, when responding to recent variations.

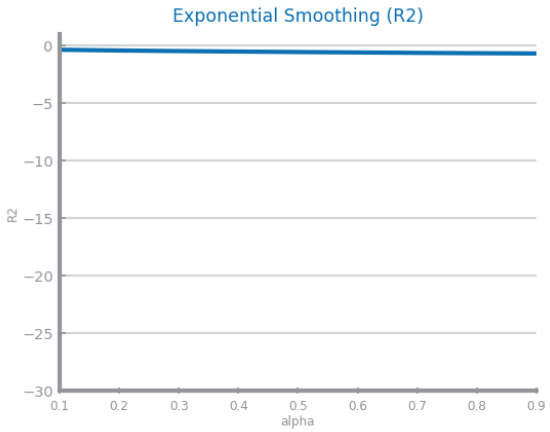

Figure 119: Forecasting study over different parameterisations of the Exponential Smoothing algorithm over time series 1

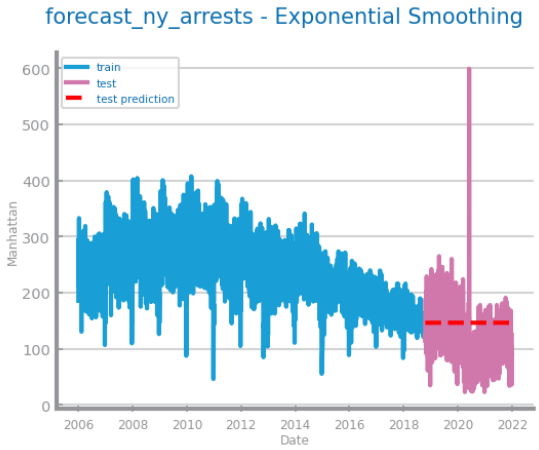

Figure 120: Forecasting plots obtained with the best parameterisation of Exponential Smoothing algorithm, over time series 1

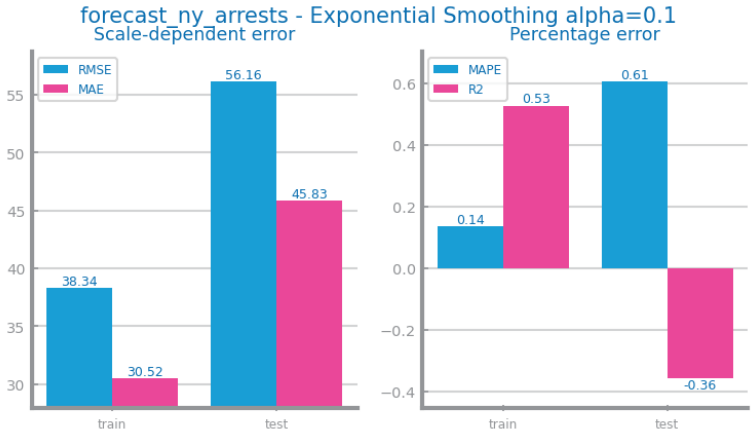

Figure 121: Forecasting results obtained with the best parameterisation of Exponential Smoothing algorithm, over time series 1

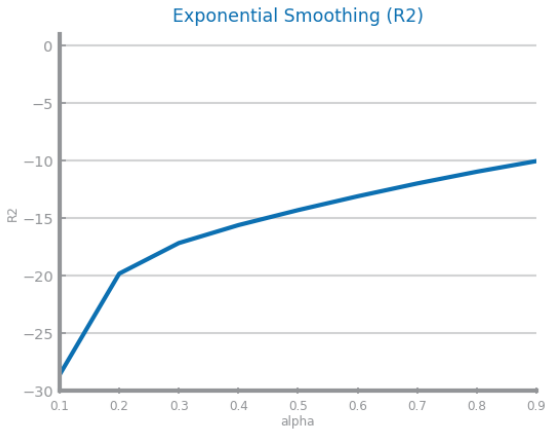

Figure 122: Forecasting study over different parameterisations of the Exponential Smoothing algorithm over time series 2

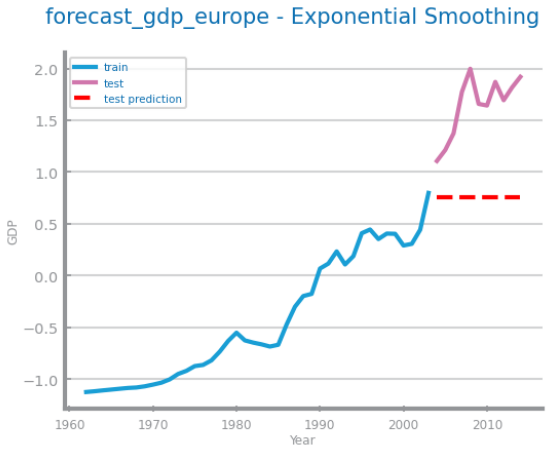

Figure 123: Forecasting plots obtained with the best parameterisation of Exponential Smoothing algorithm, over time series 2

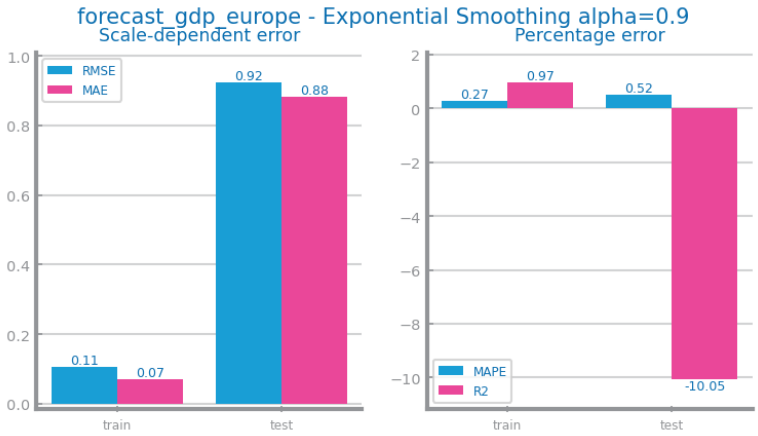

Figure 124: Forecasting results obtained with the best parameterisation of Exponential Smoothing algorithm, over time series 2

Linear Regression Model

Set 1 - Although it wasn't by a lot, it had slightly worse results than the previous two models; **Set 2** - Catches the upward trend of the series. Its R^2 performance is similar to previous models.

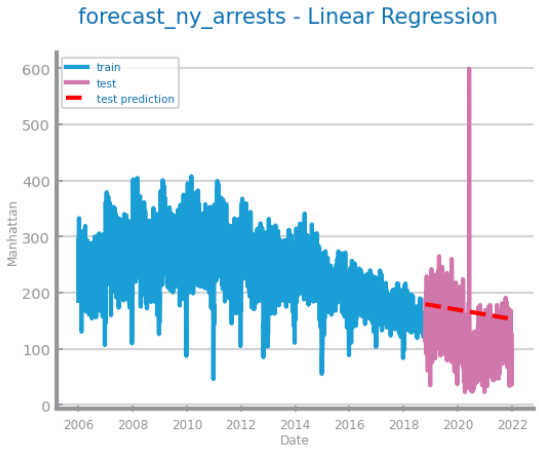

Figure 125: Forecasting plots obtained with Linear Regression model over time series 1

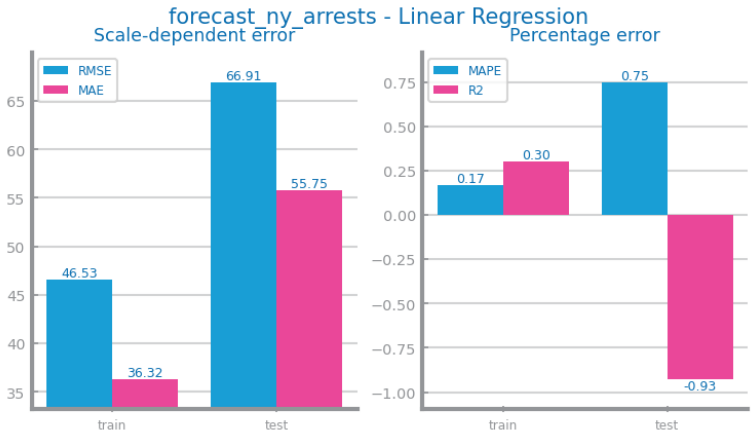

Figure 126: Forecasting results obtained with Linear Regression model over time series 1

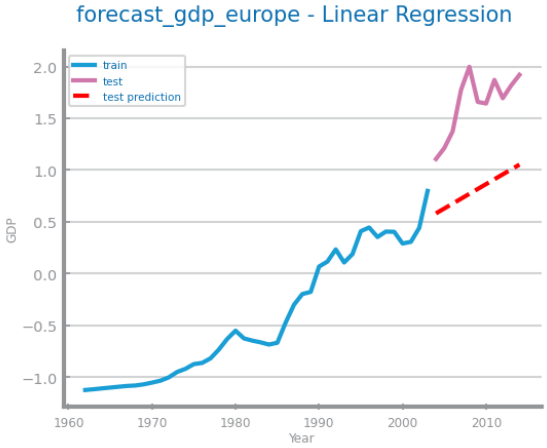

Figure 127: Forecasting plots obtained with Linear Regression model over time series 2

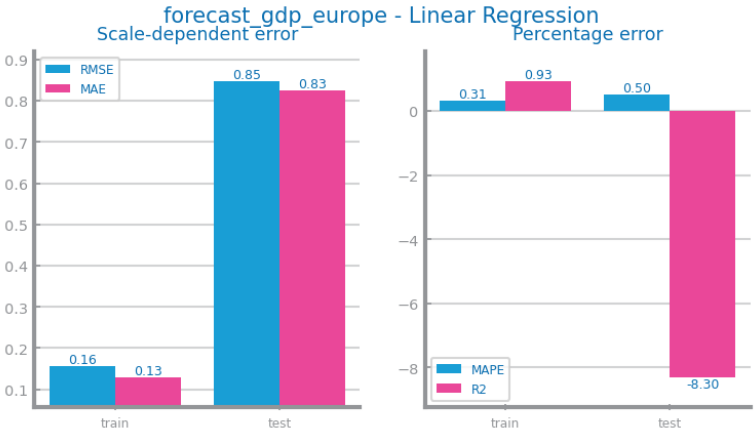

Figure 128: Forecasting results obtained with Linear Regression model over time series 2

ARIMA Model

Both - ARIMA with multiple variables outperforms its univariate form by leveraging extra features. Its superior R^2 suggests it best captures the strong autocorrelation visible in lag plots. Data size issues forced reverting prior transformations and returning to the original aggregation. Highest R^2 score achieved (besides Persistence Optimistic which is not comparable), also reducing the RMSE/MAE values.

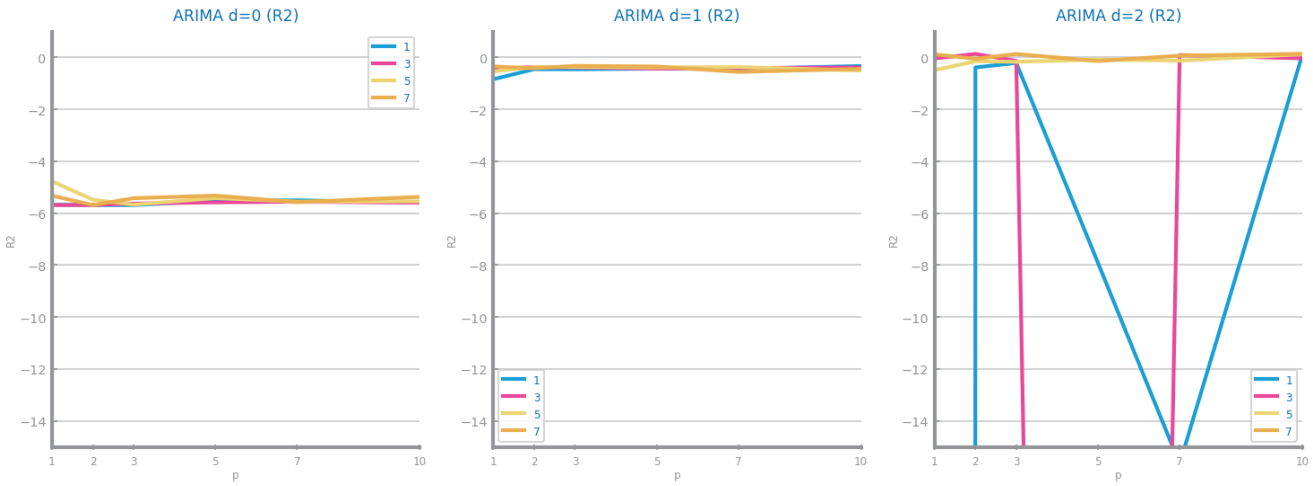

Figure 129: Forecasting study over different parameterisations of the ARIMA algorithm over time series 1, only with the target variable

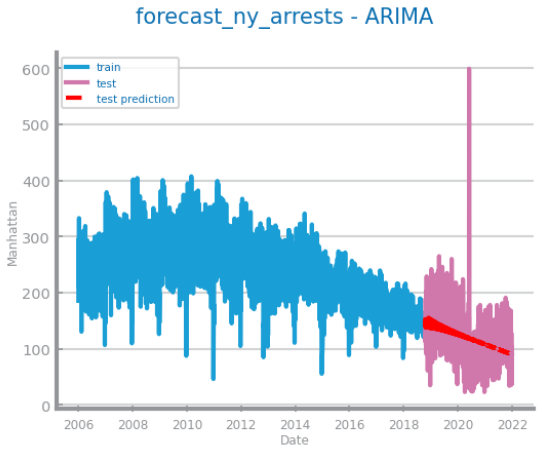

Figure 130: Forecasting plots obtained with the best parameterisation of ARIMA algorithm, over time series 1, only with the target variable

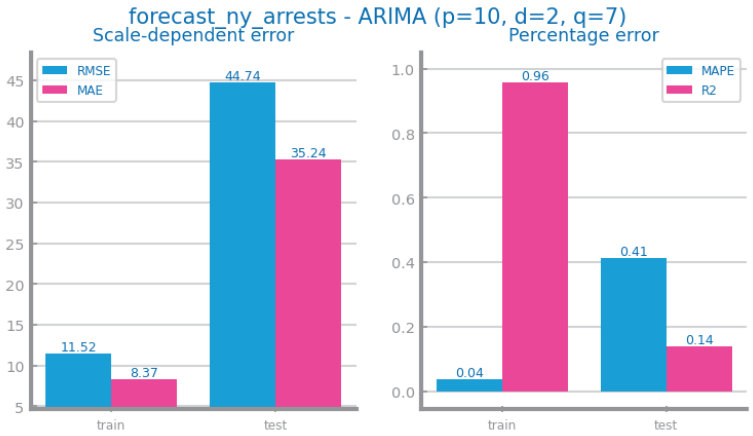

Figure 131: Forecasting results obtained with the best parameterisation of ARIMA algorithm, over time series 1, only with the target variable

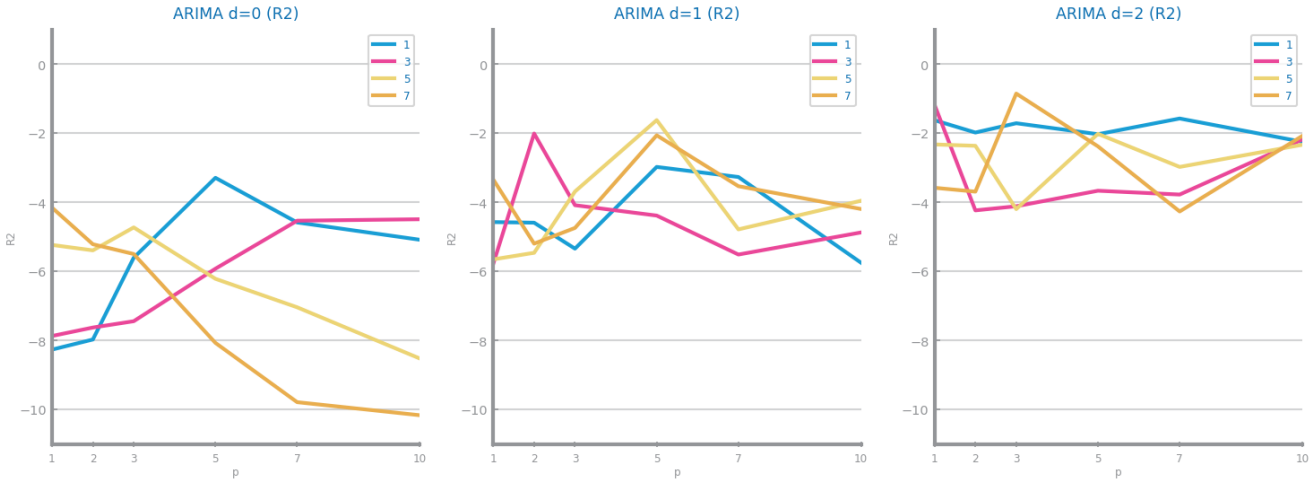

Figure 132: Forecasting study over different parameterisations of the ARIMA algorithm over time series 2, only with the target variable

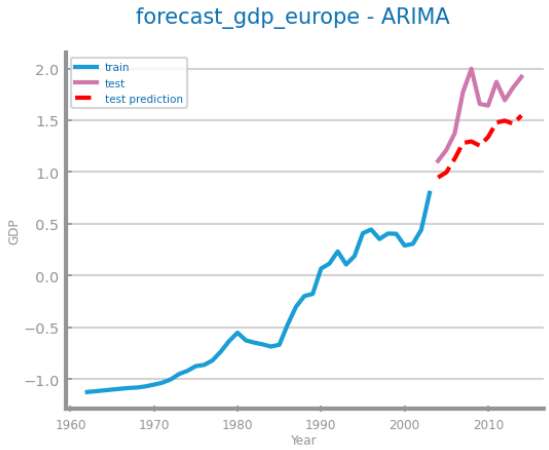

Figure 133: Forecasting plots obtained with the best parameterisation of ARIMA algorithm, over time series 2, only with the target variable

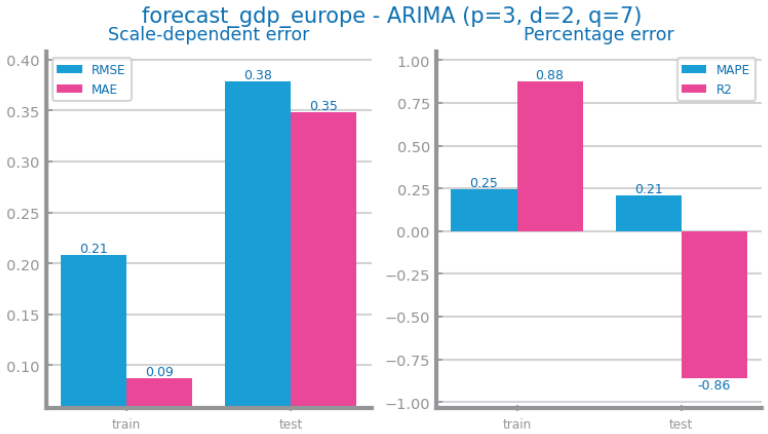

Figure 134: Forecasting results obtained with the best parameterisation of ARIMA algorithm, over time series 2, only with the target variable

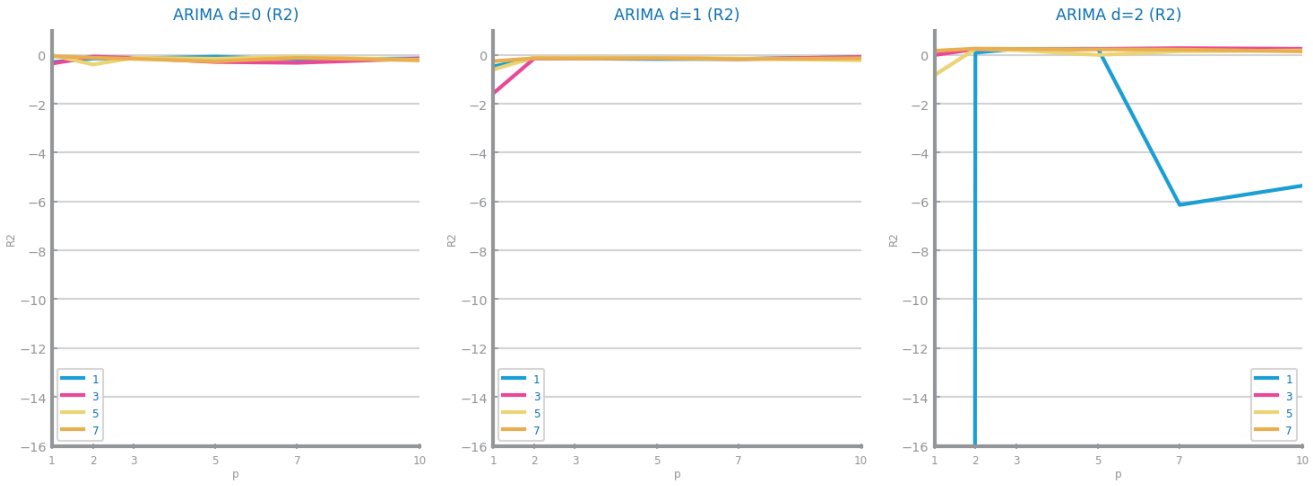

Figure 135: Forecasting study over different parameterisations of the ARIMA algorithm with multiple variables over time series 1

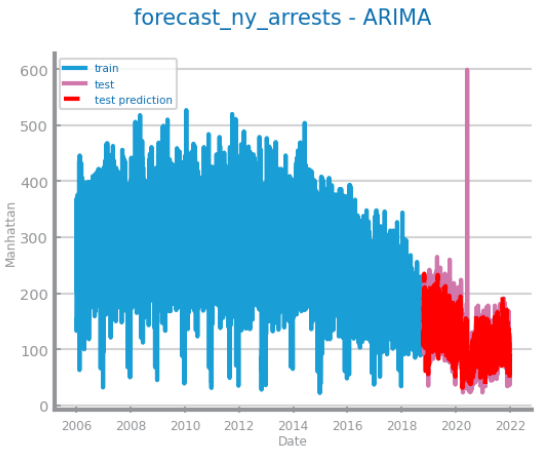

Figure 136: Forecasting plots obtained with the best parameterisation of ARIMA algorithm with multiple variables over time series 1

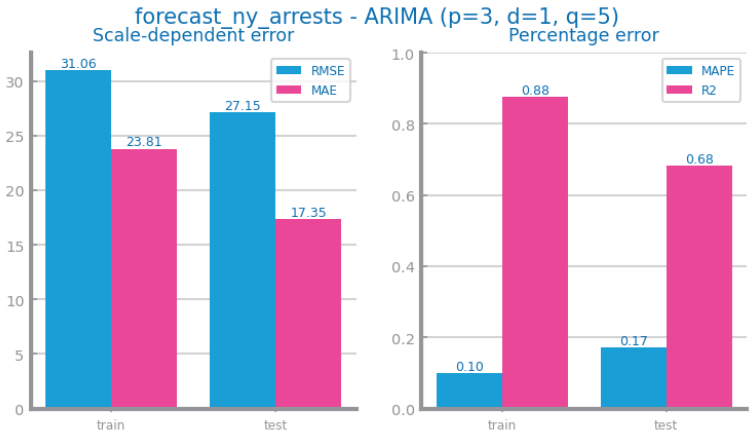

Figure 137: Forecasting results obtained with the best parameterisation of ARIMA algorithm with multiple variables over time series 1

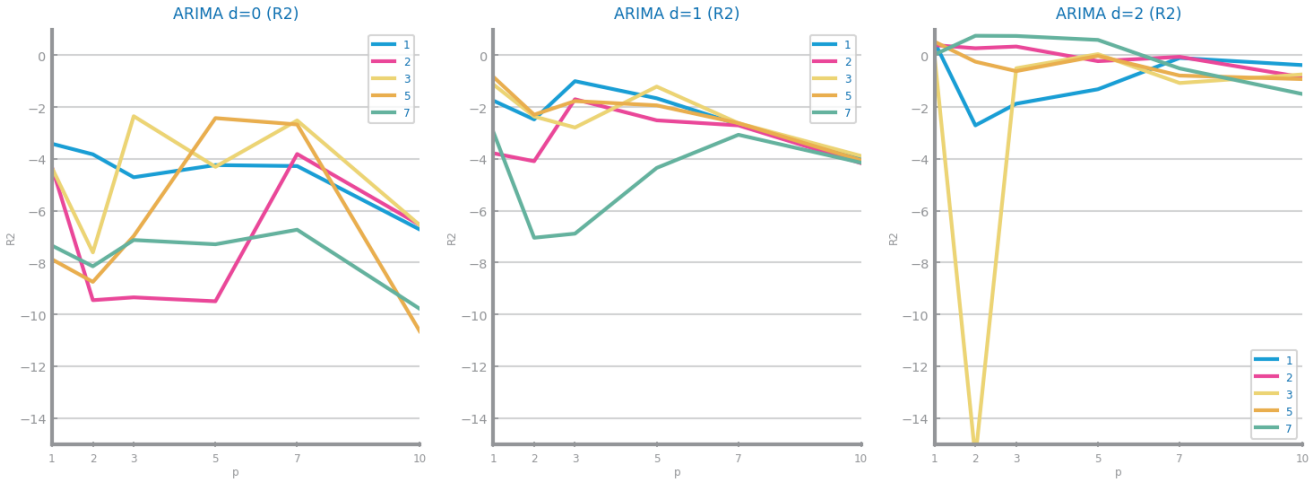

Figure 138: Forecasting study over different parameterisations of the ARIMA algorithm with multiple variables over time series 2

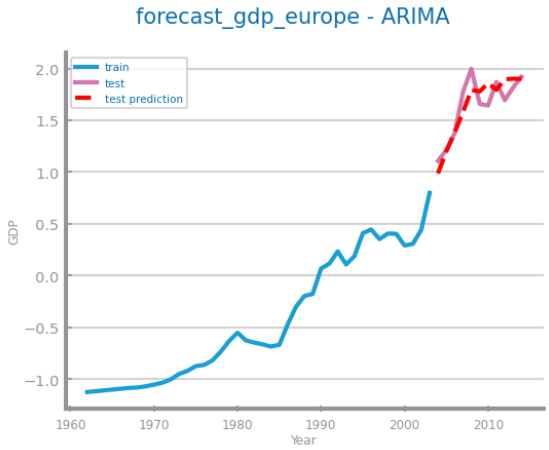

Figure 139: Forecasting plots obtained with the best parameterisation of ARIMA algorithm with multiple variables over time series 2

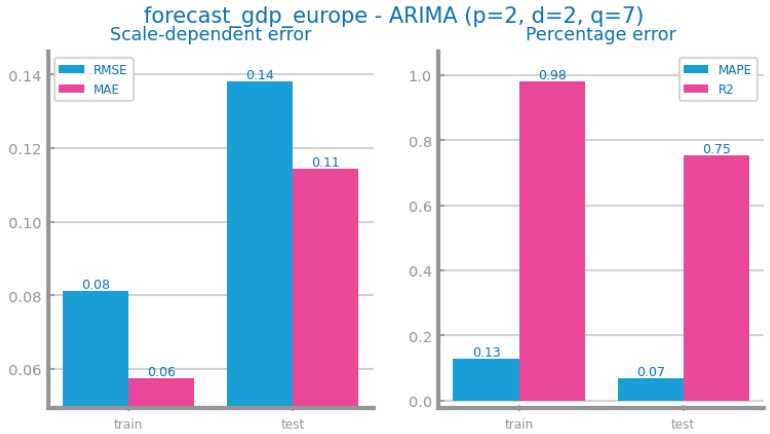

Figure 140: Forecasting results obtained with the best parameterisation of ARIMA algorithm with multiple variables over time series 2

LSTMs Model

Both - Using only the target series, the best LSTM yields higher R2 and moderate errors, showing it captures temporal patterns more effectively. Adding extra features may introduce noise or insufficient signals, reducing multi-variable performance.

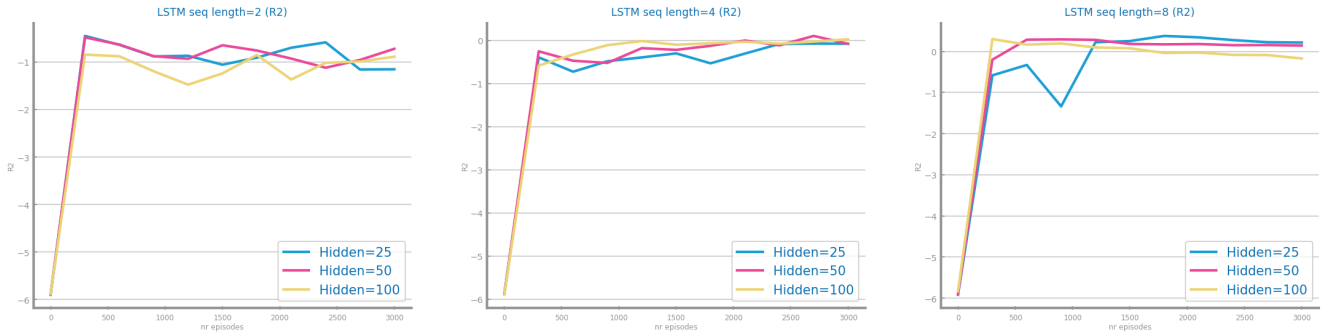

Figure 141: Forecasting study over different parameterisations of the LSTMs over time series 1, only with the target variable

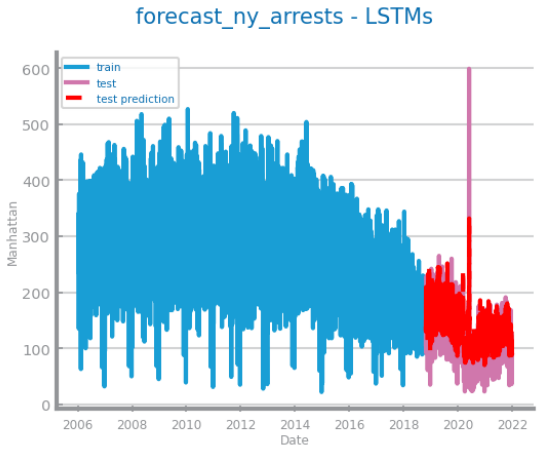

Figure 142: Forecasting plots obtained with the best parameterisation of LSTMs, over time series 1, only with the target variable

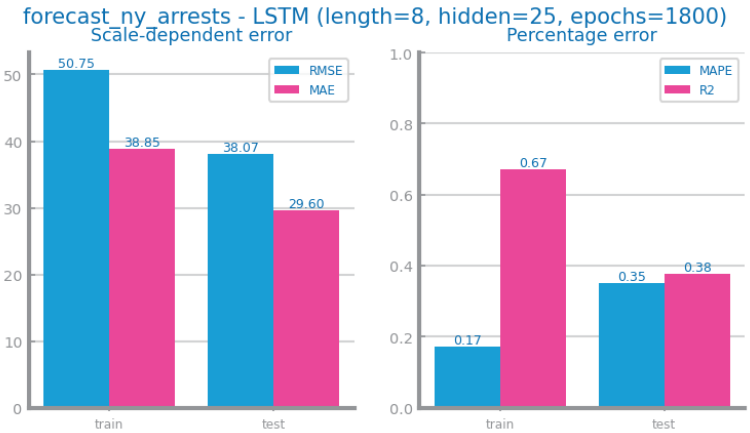

Figure 143: Forecasting results obtained with the best parameterisation of LSTMs, over time series 1, only with the target variable

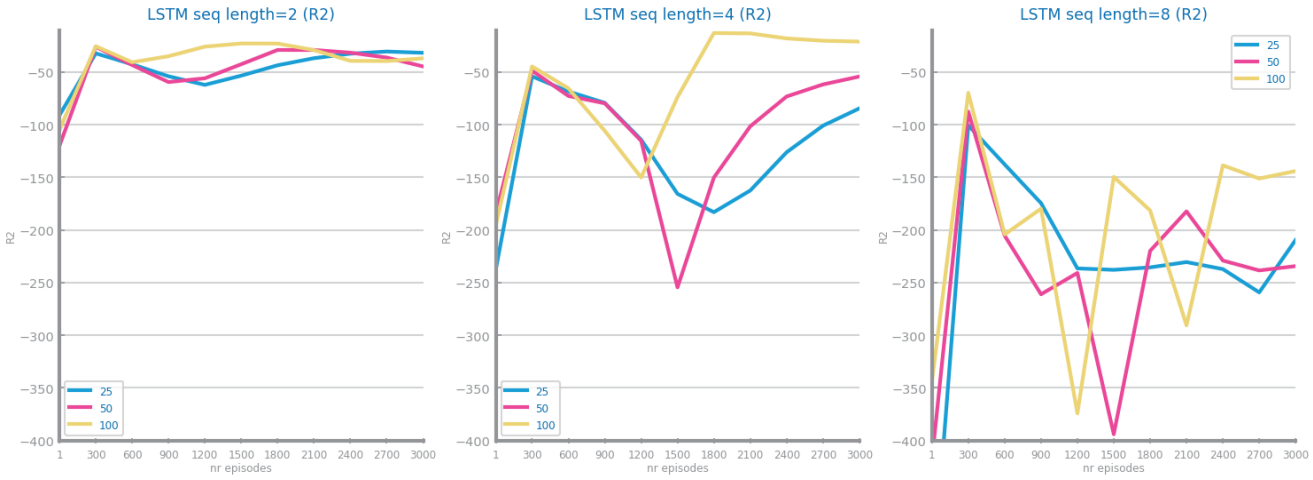

Figure 144: Forecasting study over different parameterisations of the LSTMs over time series 2, only with the target variable

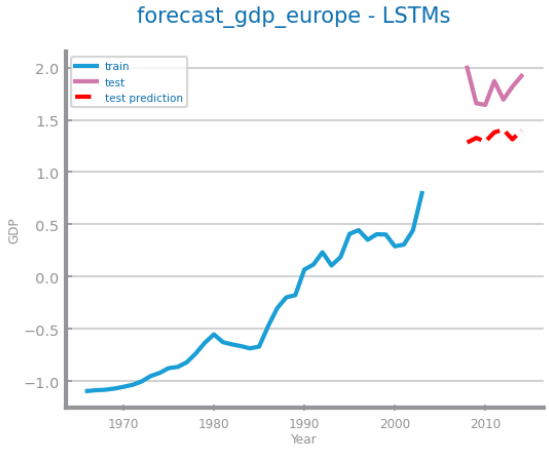

Figure 145: Forecasting plots obtained with the best parameterisation of LSTMs, over time series 2, only with the target variable

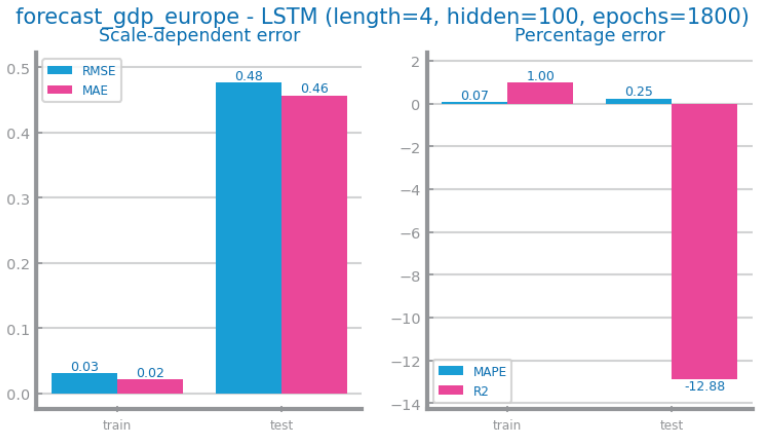

Figure 146: Forecasting results obtained with the best parameterisation of LSTMs, over time series 2, only with the target variable

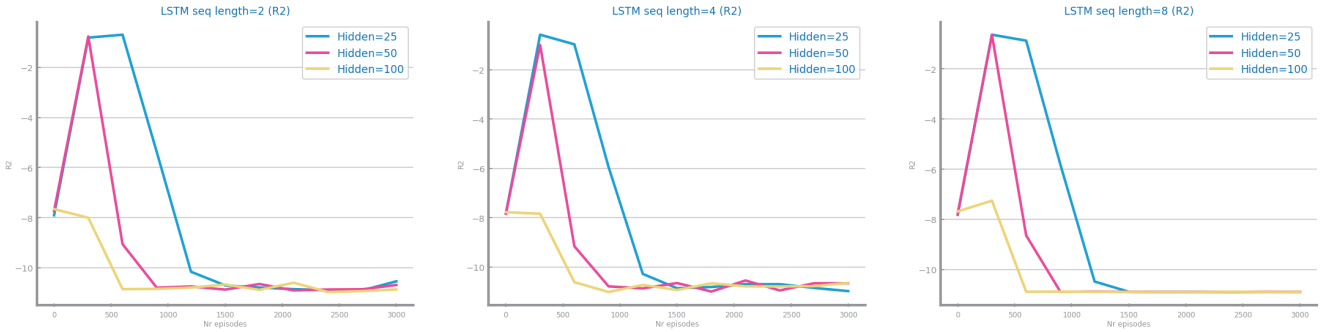

Figure 147: Forecasting study over different parameterisations of LSTMs with multiple variables over time series 1

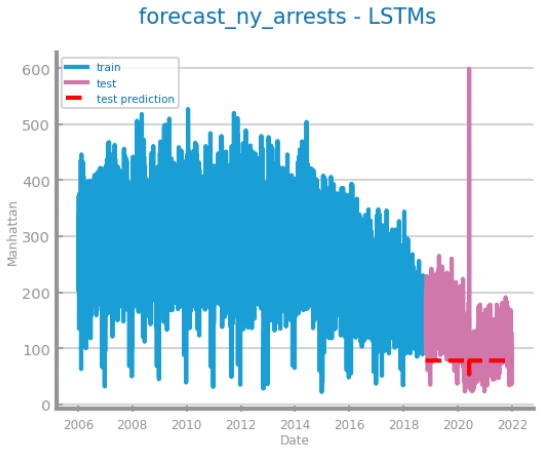

Figure 148: Forecasting plots obtained with the best parameterisation of LSTMs with multiple variables over time series 1

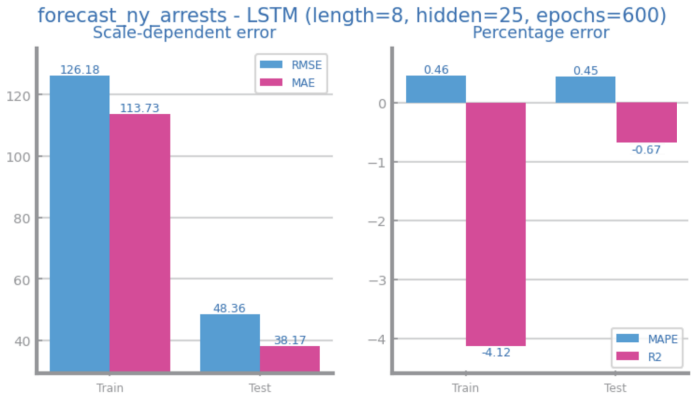

Figure 149: Forecasting results obtained with the best parameterisation of LSTMs with multiple variables over time series 1

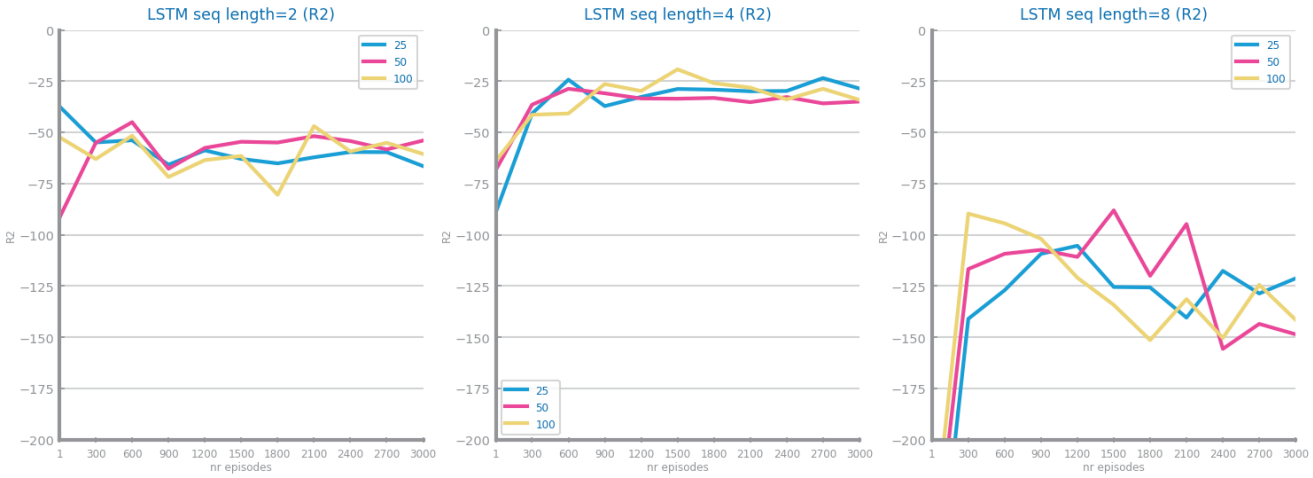

Figure 150: Forecasting study over different parameterisations of LSTMs with multiple variables over time series 2

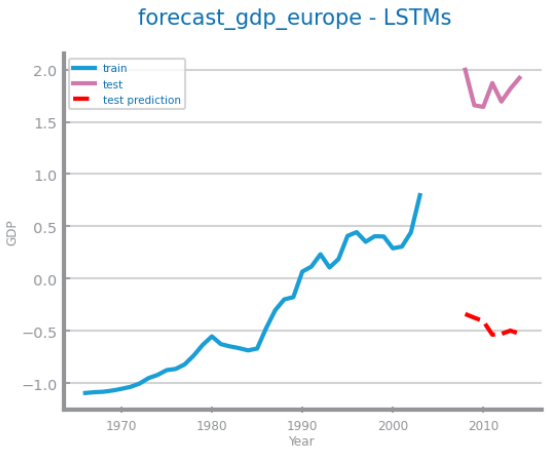

Figure 151: Forecasting plots obtained with the best parameterisation of LSTMs with multiple variables over time series 2

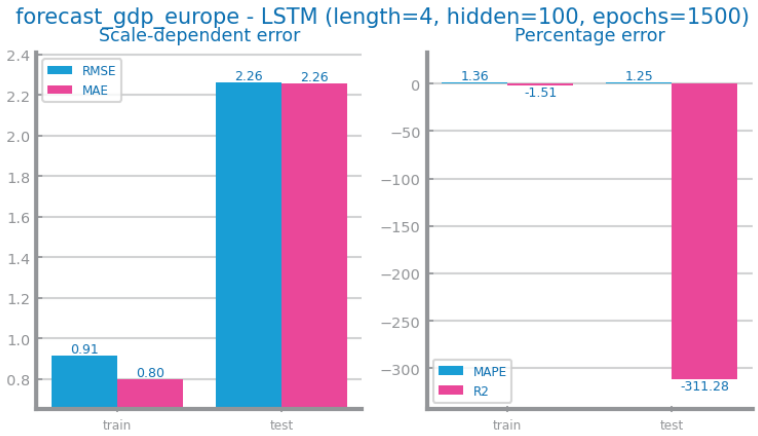

Figure 152: Forecasting results obtained with the best parameterisation of LSTMs with multiple variables over time series 2

8 CRITICAL ANALYSIS

Set 1 - The modeling process utilized a dataset with smoothing and daily aggregation, keeping many of the general characteristics of the initial dataset. Both the Rolling Mean and Exponential Smoothing methods yielded similar results, likely due to the large dataset size and minimal oscillations in the values. The ARIMA models, both single and multi-variable, produced very good R^2 values because the mean, variance, and autocorrelation of the series remained stable over time, even though the dataset was non-stationary. On the other hand, the LSTM model, when considering only one variable, performed well, however, when multiple variables were included, its performance deteriorated; **Set 2** - A 5-year aggregation smoothed out noisy fluctuations while preserving the main upward trajectory and complexity of the data. However, it complicated ARIMA, given the smaller dataset size, so the original yearly aggregation was used with the standard scaler applied. Analysis of data distribution revealed consistently high autocorrelation, which guided the modeling approach. Rolling Mean smoothing erased nuances critical for accurate long-term GDP forecasts. Both Linear Regression and Exponential Smoothing produced large negative R^2 values, indicating their failure to capture rapid growth. Although LSTMs offer advanced capabilities in sequence modeling, they underutilized the strong autocorrelation present in the data and consequently underperformed. In contrast, ARIMA successfully leveraged the nonstationary nature of the GDP data by directly incorporating autocorrelation through its lag-based structure, yielding superior predictive accuracy; **Both** - As a result, ARIMA emerged as the most effective model, accurately capturing the evolving trends in the data while maintaining computational efficiency.

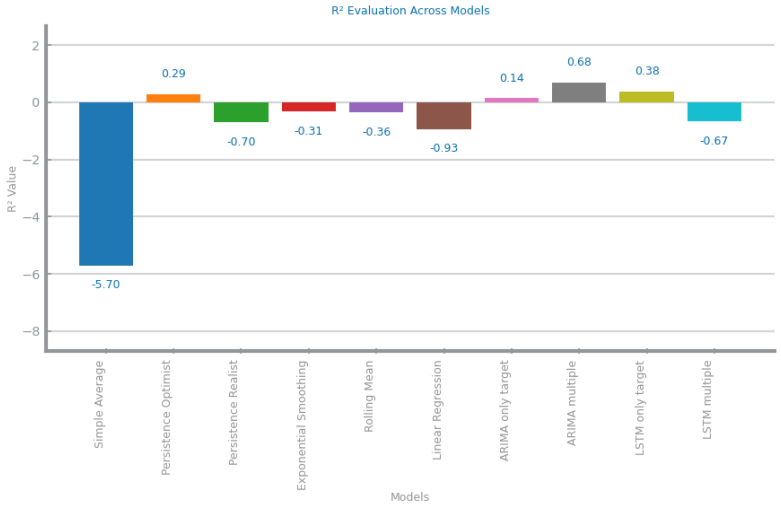

Figure 153: Models' R^2 evaluation metric for time series 1

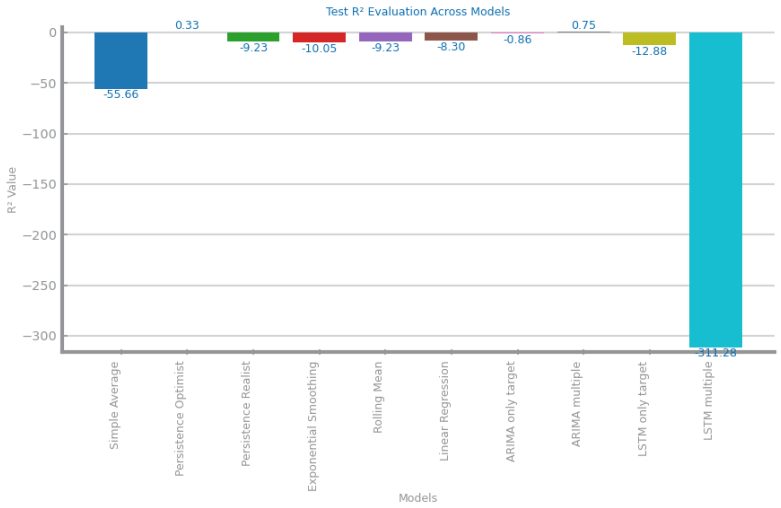

Figure 154: Models' R^2 evaluation metric for time series 2