TÉCNICO LISBOA

DS 2023

Data Science Project

Team nr: 13	**Student 1:**	Gonçalo Gonçalves	**IST nr:**	99226
	Student 2:	José Cruz	**IST nr:**	99260
	Student 3:	Jorge Santos	**IST nr:**	99258
	Student 4:	Matilde Heitor	**IST nr:**	99284

The present document presents a template for the Data Science Project report. It specifies the mandatory format and suggests the structure to follow. All text with grey background shall be replaced with the analysis made over the datasets. Put your charts in the *images* folder, and set the name of the file in the *includegraphics* command, after uncommenting it.

CLASSIFICATION

1 DATA PROFILING

Ds 1 has a lot of records, impacting future decisions.

Not much preprocessing was done.

The *Age* variable in ds 2 had values with the character "_", the character was removed.

Other anomalies were kept. **Shall not exceed 200 characters.**

Data Dimensionality

Both datasets have much more records than variables, avoiding curse of dimensionality.

There are no date variables - no time frame associated . The health dataset predominantly features symbolic variables, particularly binary ones, reflecting the challenge of quantifying clinical observations. The services one is more numerical, demonstrating the precision of its financial context.

In both datasets no variable has more than 20% missing values, therefore all variables are reasonably interesting. **Shall not exceed 500 characters.**

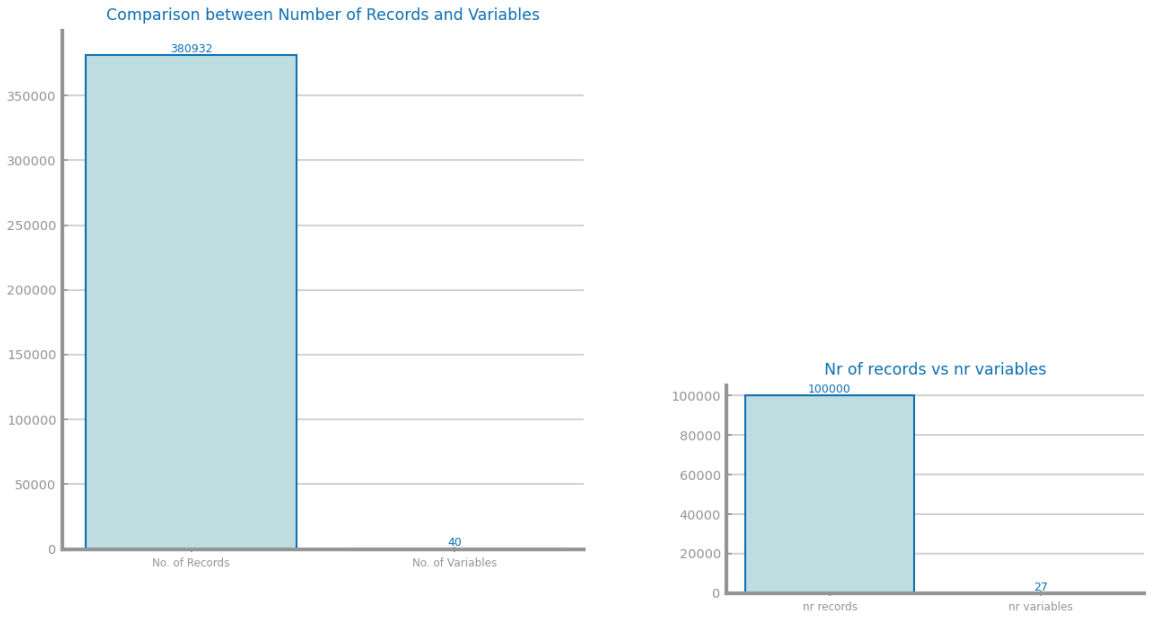


Figure 1: Nr Records x Nr variables for dataset 1 (left) and dataset 2 (right)

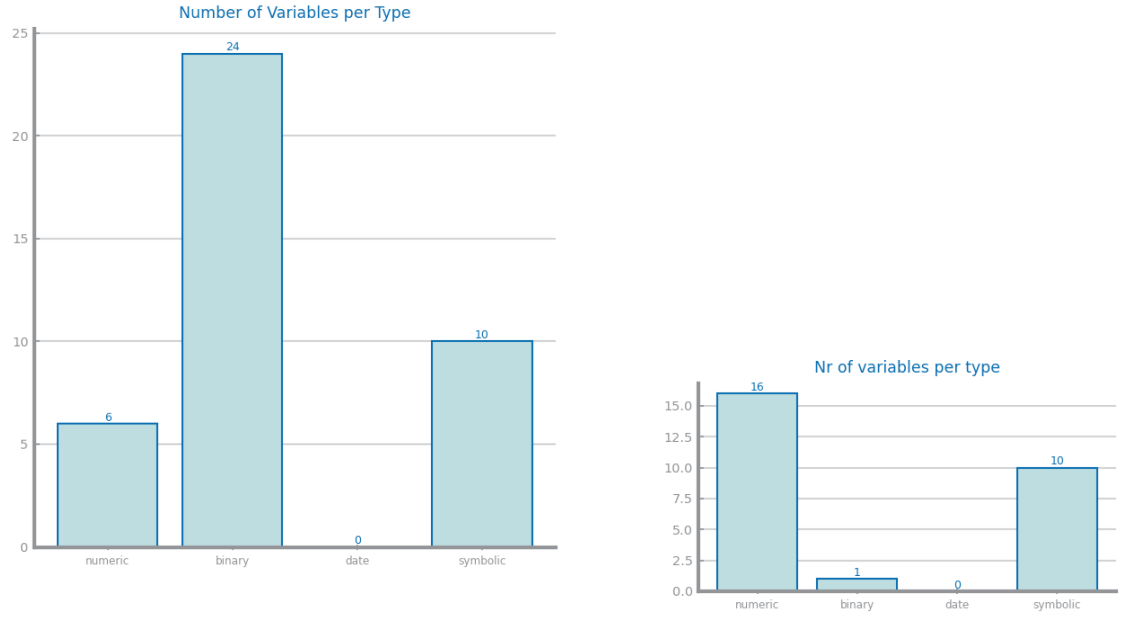


Figure 2: Nr variables per type for dataset 1 (left) and dataset 2 (right)

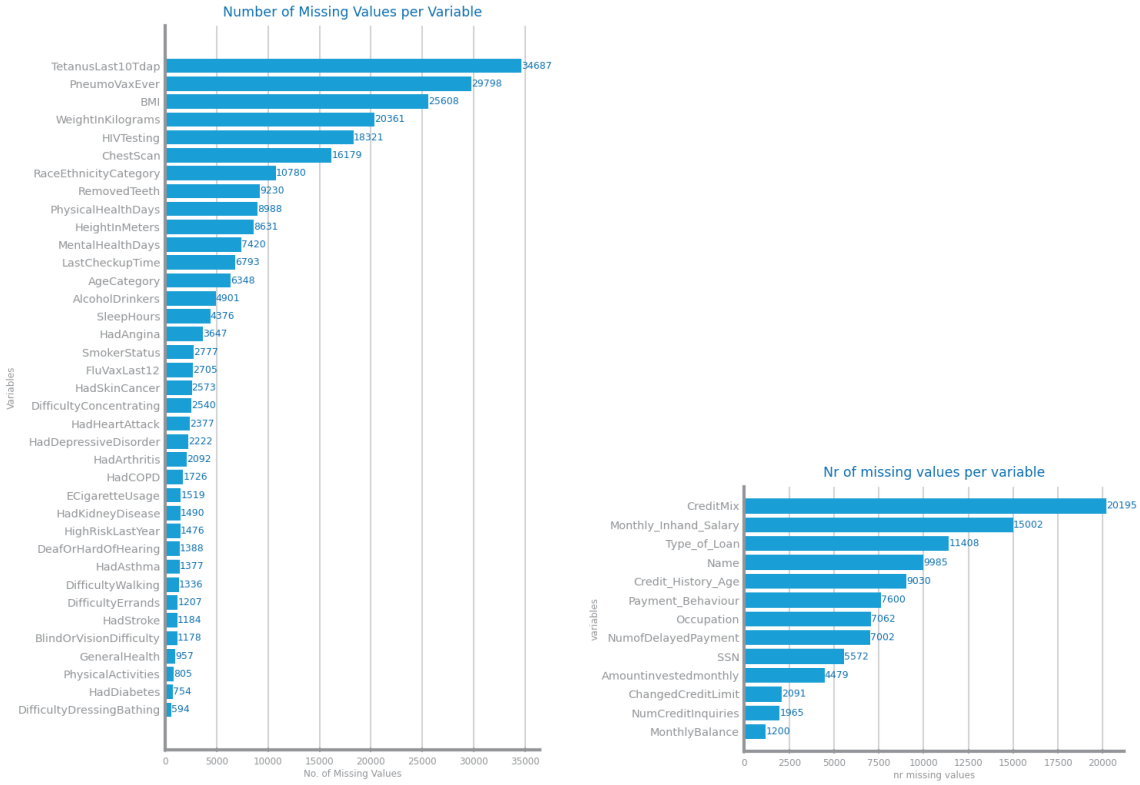


Figure 3: Nr missing values for dataset 1 (left) and dataset 2 (right)

Data Distribution

Global boxplots depend on the scaling of variables, in our cases, they are pretty much useless

All numeric variables have quite high variability in both datasets.

The majority of numerical variables follow a normal distribution.

IQR is more robust to extreme values, more desirable for the study of the second dataset. The stddev is better suited for variables that follow a normal distribution.

The class distribution is similar in both datasets, a 25%-75%. Some balancing might be required. **Shall not exceed 500 characters.**

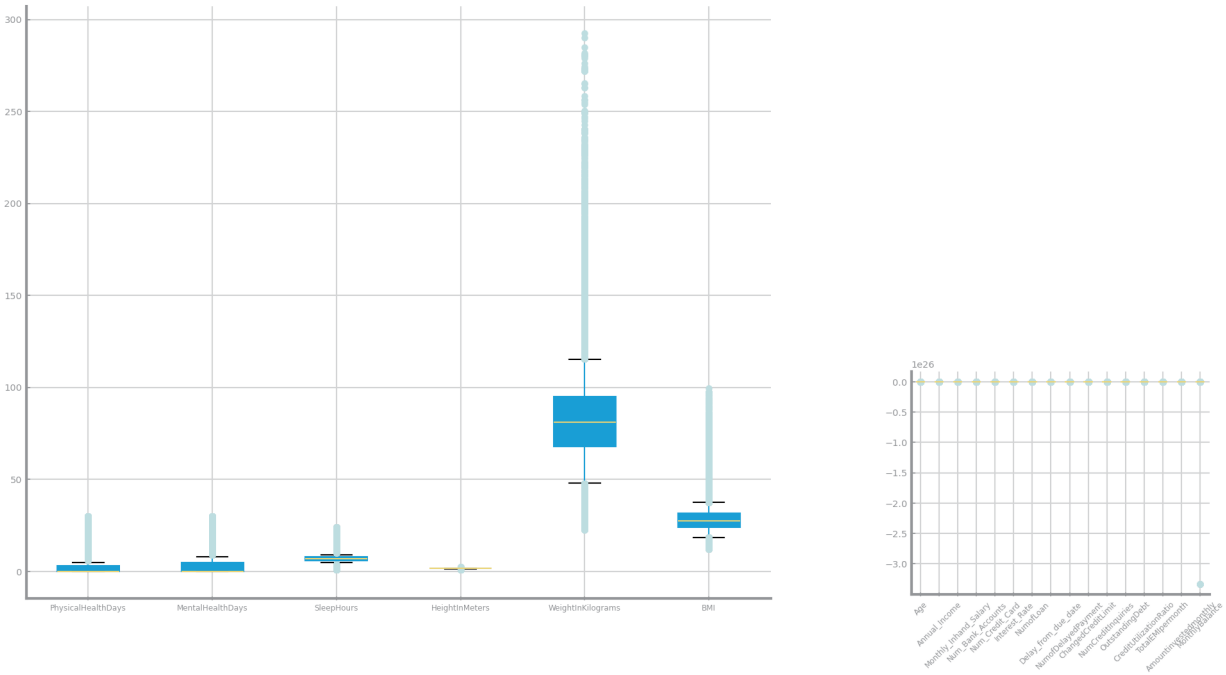


Figure 4: Global boxplots dataset 1 (left) and dataset 2 (right)

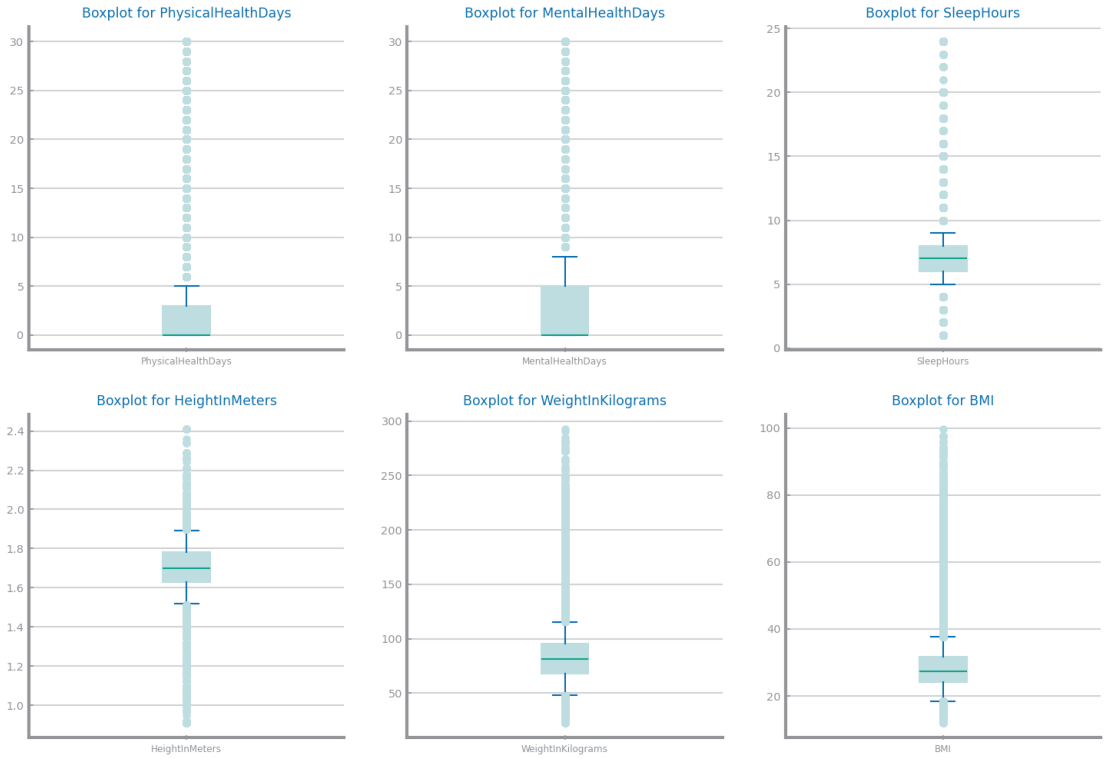


Figure 5: Single variables boxplots for dataset 1

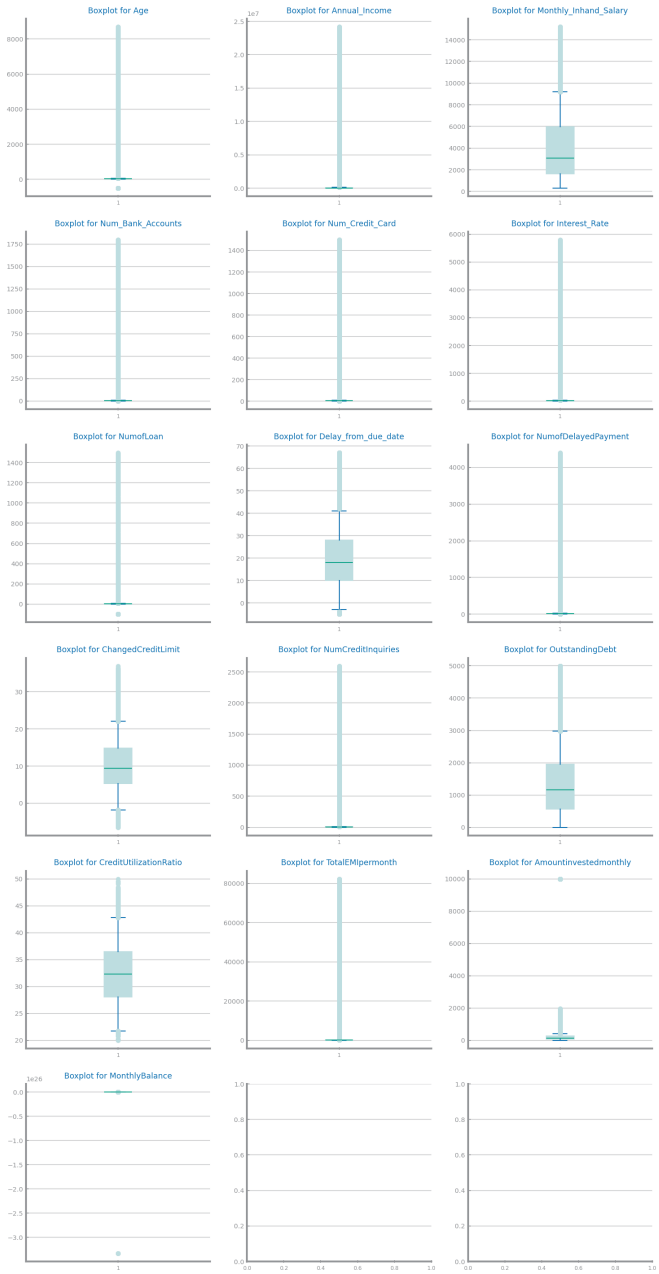


Figure 6: Single variables boxplots for dataset 2

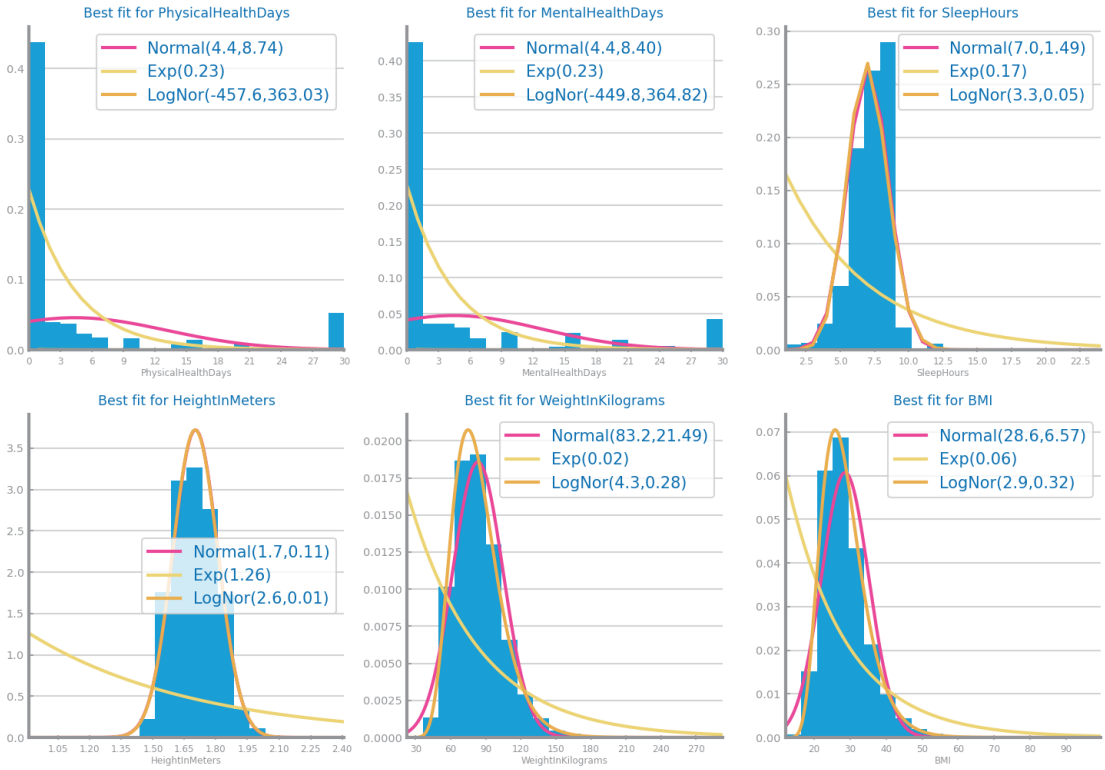


Figure 7: Histograms for dataset 1

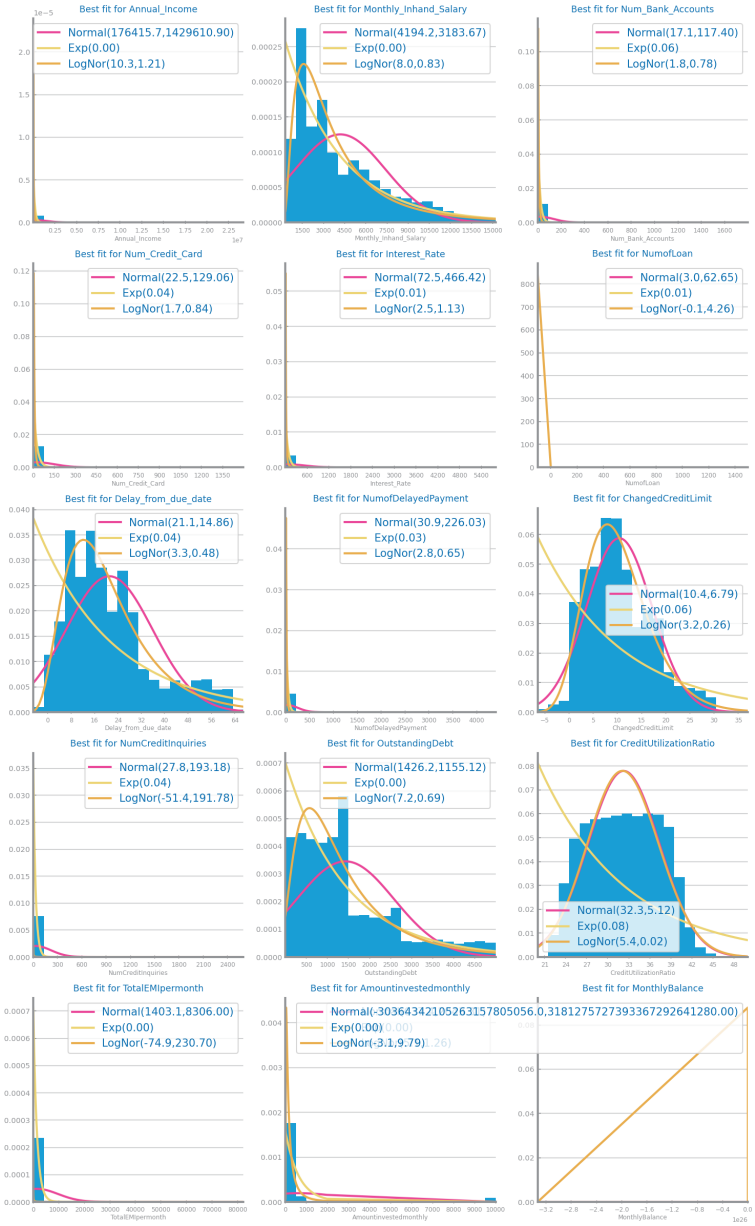


Figure 8: Histograms for dataset 2

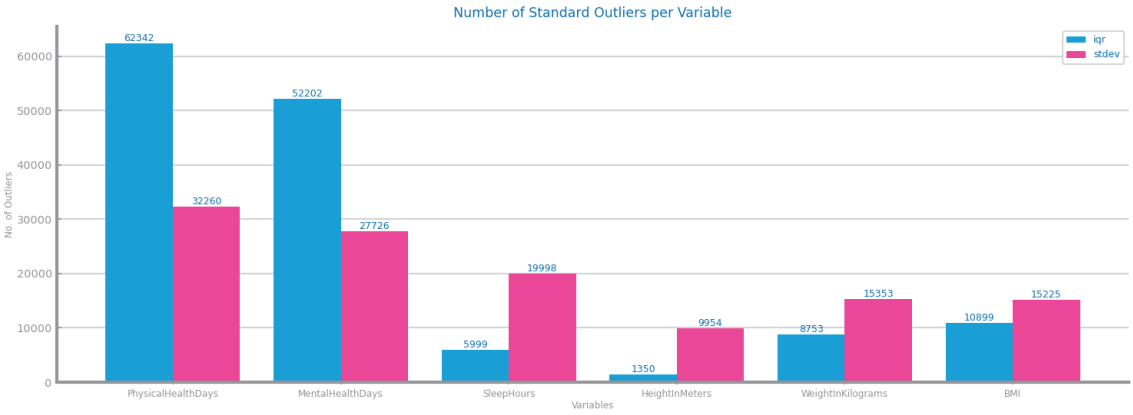


Figure 9: Outliers study dataset 1

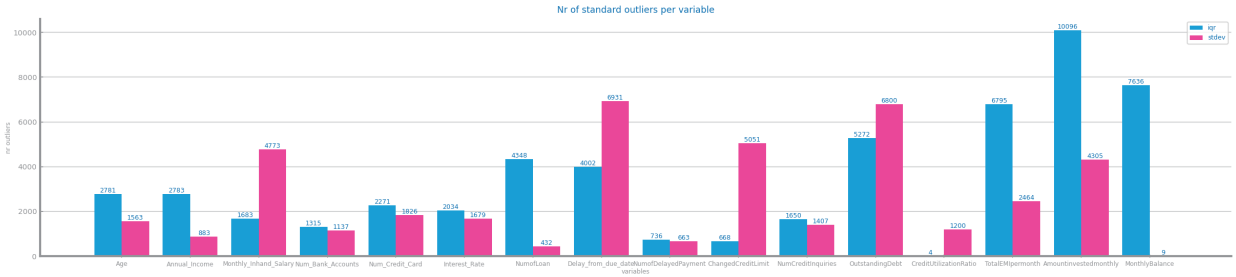


Figure 10: Outliers study dataset 2

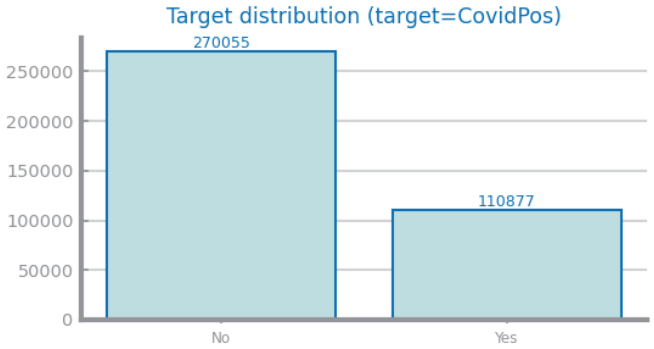


Figure 11: Class distribution for dataset 1

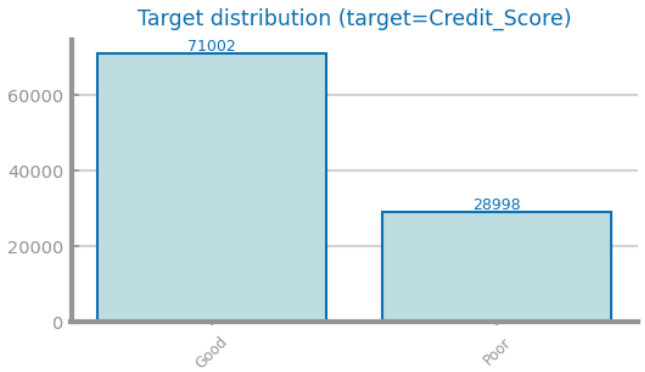


Figure 12: Class distribution for dataset 2

Data Granularity

In dataset 1, the variables *LastCheckupTime*, *State* and *AgeCategory* have interesting taxonomies, allowing for reduction of variance without harming the distribution and information of them.

For dataset 2. Some of the variables clearly had too much differente values, this study allowed us to know if descending in the taxonomy hierachy would not signifincantly alter the distribution of these variables. Notable examples of this are the *Type_of_Loan* and *Credit_History_Age* variables. **Shall not exceed 500 characters.**

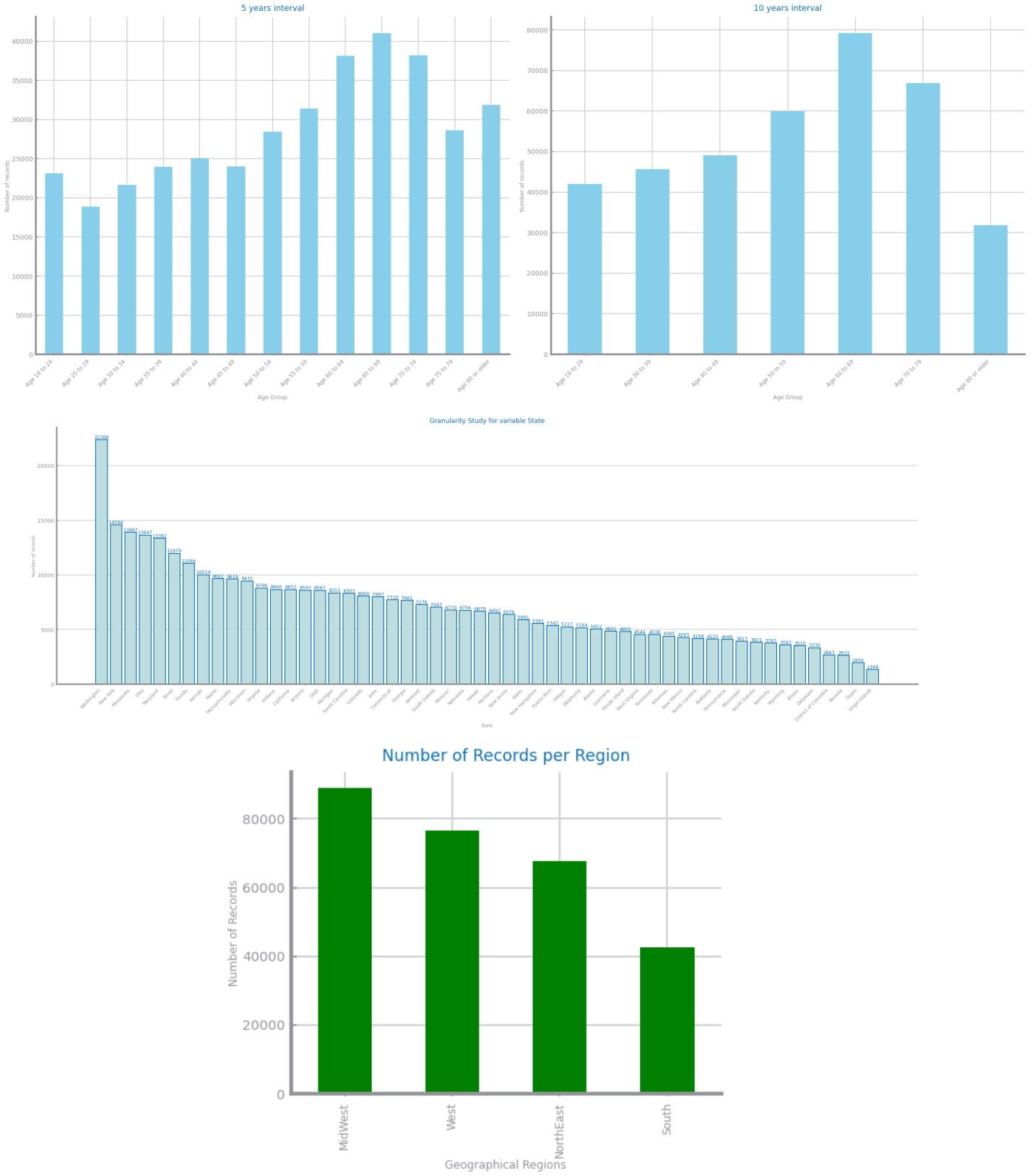


Figure 13: Granularity analysis for dataset 1

Figure 14: Granularity analysis for dataset 2

Data Sparsity

Binary on binary is not interesting for sparsity evaluation. We can see which variables have more impact on the class and the correlation between them.

We encoded the symbolic variables into numeric for the correlation analysis. In dataset 1, there are not many variables with high correlation. Even when there are it might not be interesting - gender/height, BMI/weight, Name/ID.

In dataset 2 we can infer more information about the behaviour of people and what leads to their credit score. **Shall not exceed 500 characters.**

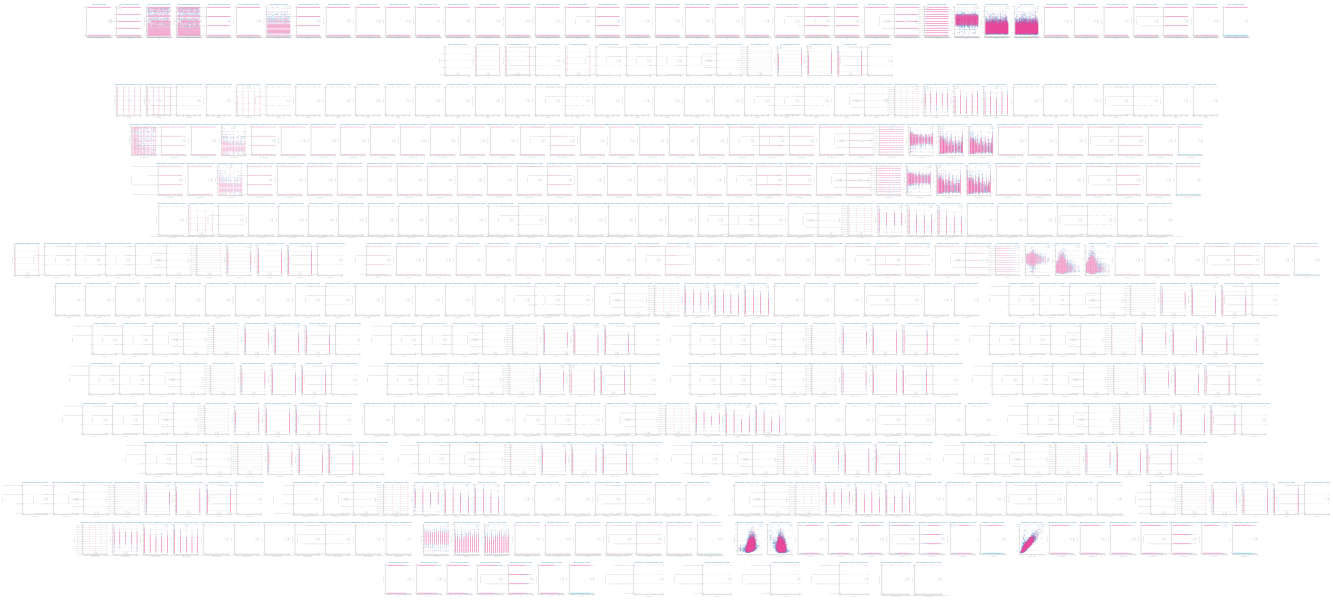


Figure 15: Sparsity analysis for dataset 1

Figure 16: Sparsity analysis for dataset 2

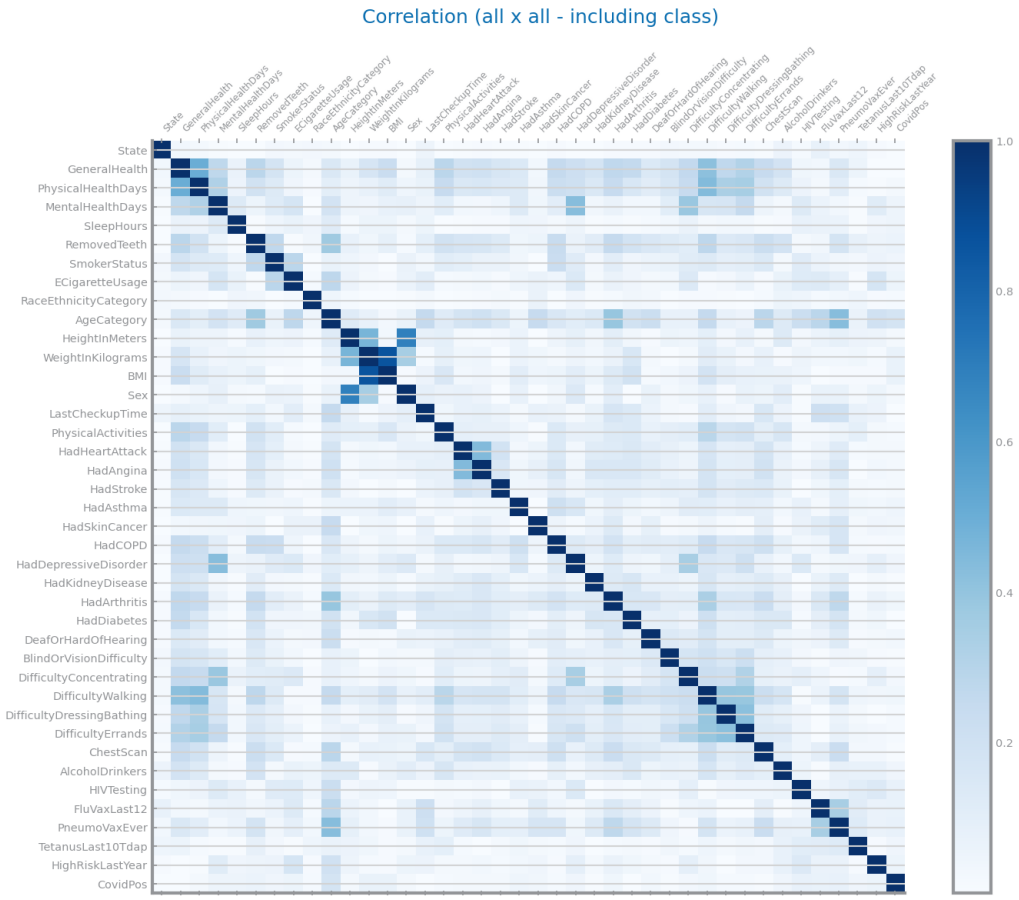


Figure 17: Correlation analysis for dataset 1

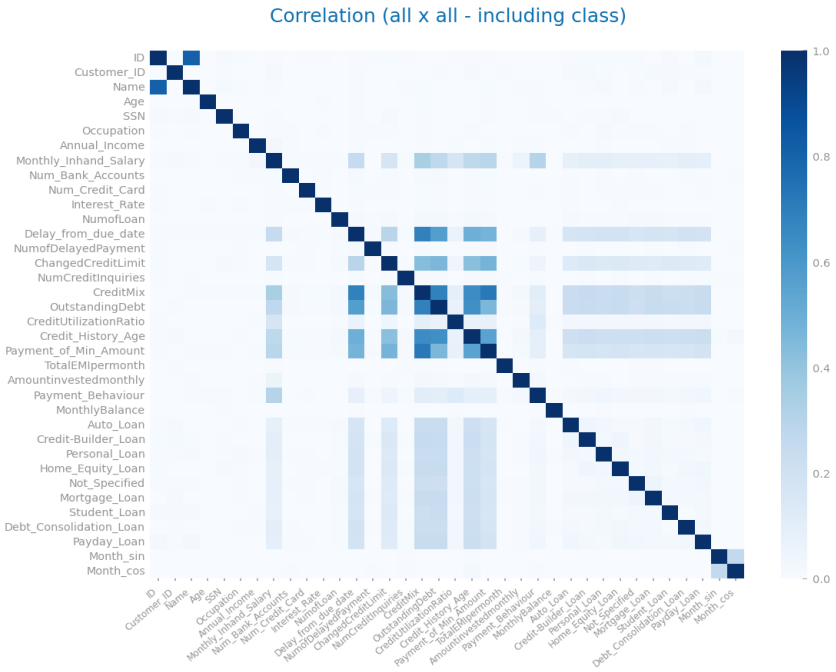


Figure 18: Correlation analysis for dataset 2

2 DATA PREPARATION

Variables Encoding

Dataset 1 - Health

In this step all symbolic variables were converted to numeric. All the binary got turned into zeros and ones. Ordinal variables preserved their order - *GeneralHealth*, *LastCheckupTime*, *RemovedTeeth*, *HadDiabetes*, *SmokerStatus*, *ECigaretteUsage*, *AgeCategory* and *TetanusLast10Tdap*. The *State* got its granularity reduced into regions according to the previously done study.

Dataset 2 - Services

In this step we converted all non numerical variables into numerical. For the *ID*, *Customer_ID*, *SSN* and *Name* were transformed into numbers, preserving their uniqueness. For variables that had order to their values we used ordinal linear encoding, these were *Credit_Score*, *Credit_Mix*, *Payment_of_Min_Amount*, *Credit_History_Age* and *Payment_Behaviour*. For the *Month* we used cyclic encoding. We reduced the granularity of the *Occupation* variable, turning it more broad. For the *Type_of_Loan* variable we used dummification mixed with the results of the taxonomy evaluation. **Shall not exceed 500 characters *for each dataset.***

Missing Value Imputation

Firstly, we dropped all records with more than 90% missing values. We then applied two different filling strategies, frequence and KNN, which had very similiar results after evaluation (we didn't bother experimenting the constant strategy as it changes the data distribution). The frequence strategy (using mean and mode) came out on top by little, hence our

choice to use it to replace the missing values in both datasets. **Shall not exceed 500 characters.**

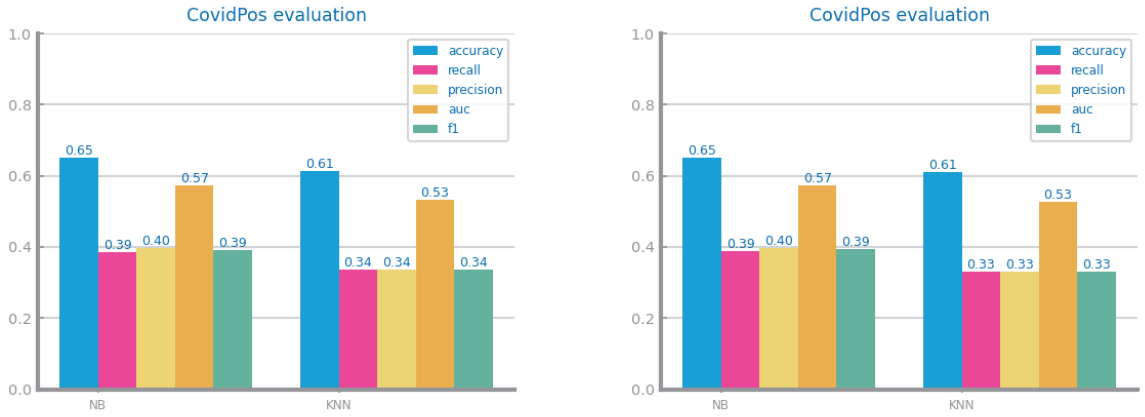


Figure 19: Missing values imputation results with different approaches for dataset 1. Frequence (left) and KNN (right) strategies

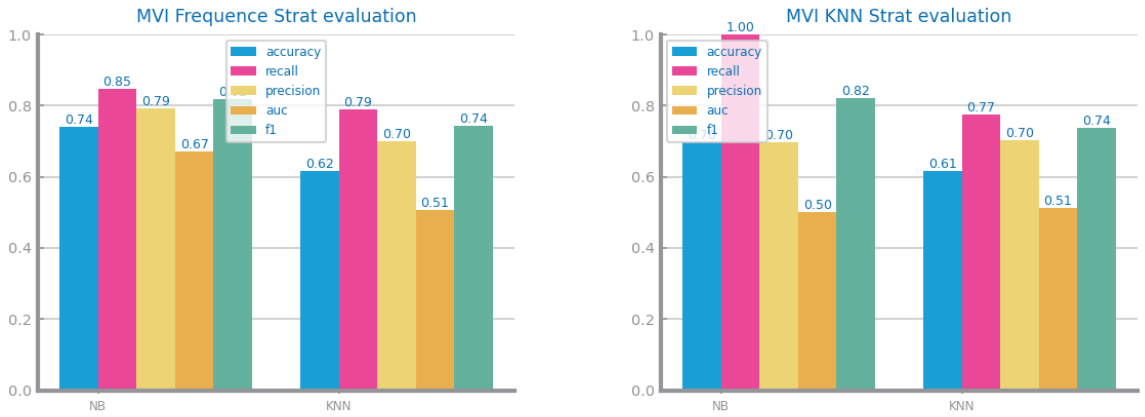


Figure 20: Missing values imputation results with different approaches for dataset 2

Outliers Treatment

For ds1 the tested strategies significantly worsened the evaluation, particularly the recall, for which we give more attention due to the domain of the data. No modification was applied for ds1.

For ds2 the strategies attained similar results but the best evaluation values were by std based dropping (mainly looking at accuracy), this was our choice moving forward. Also, we knew there were significant outliers from the profiling study, motivating us to apply a measure to deal with them. **Shall not exceed 500 characters.**

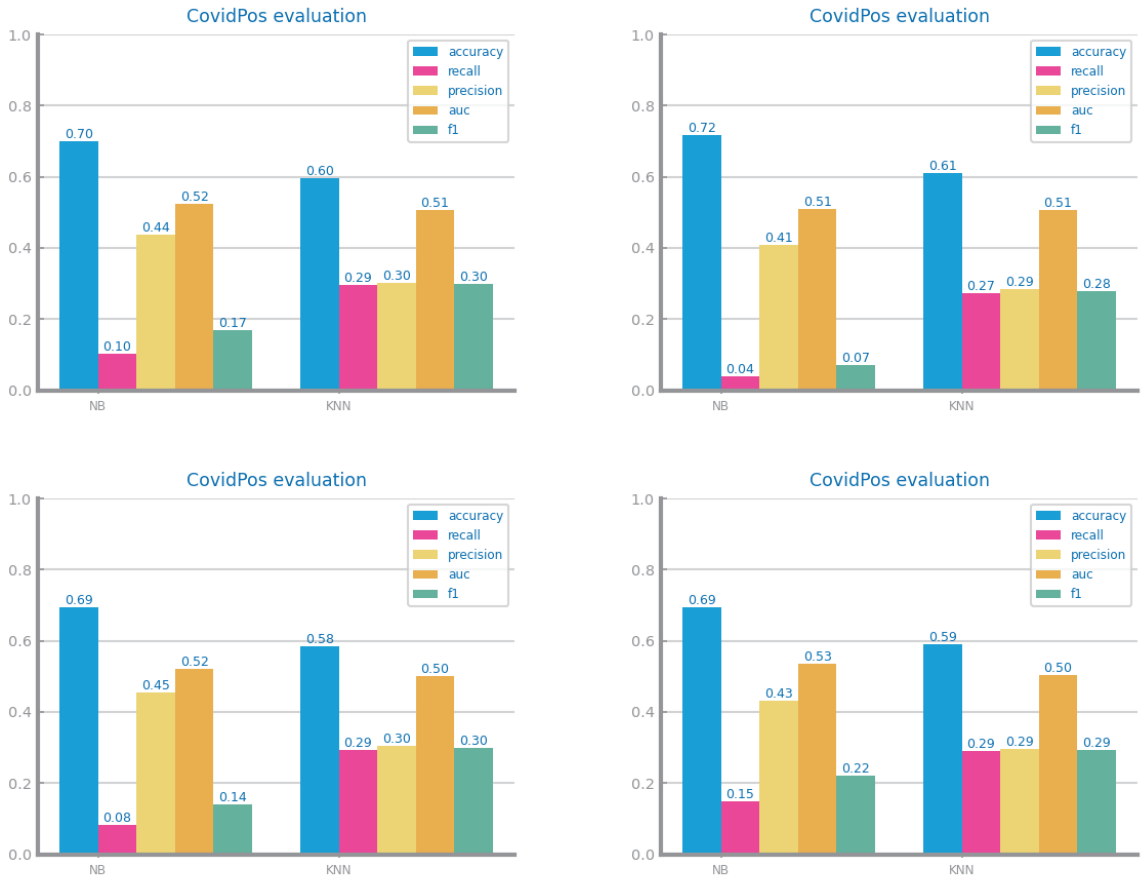


Figure 21: Outliers imputation results with different approaches for dataset 1. Rep_fixed_median (top left), rowDrop_NotStdBased (top right), rowDrop_StdBased (bottom left) and truncating_minmax (bottom right)

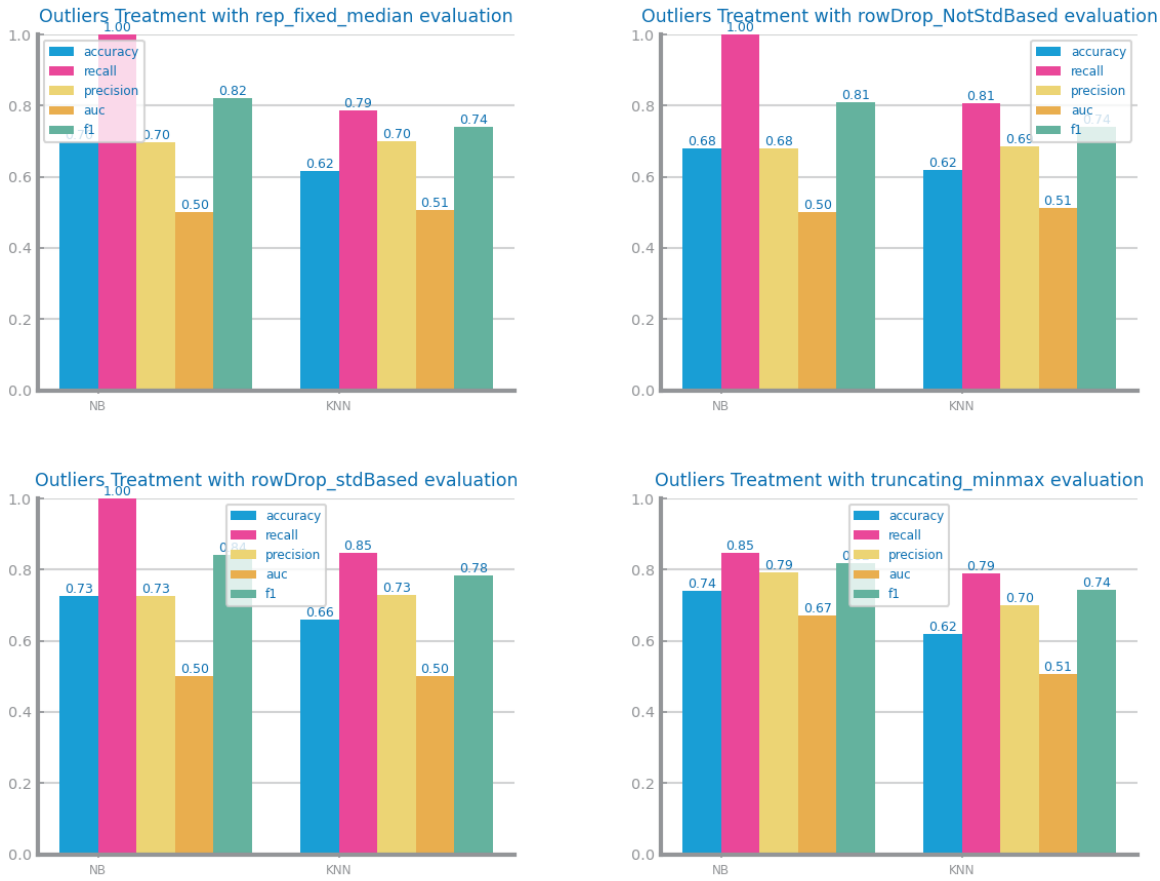


Figure 22: Outliers imputation results with different approaches for dataset 2

Scaling

Comparing the MinMax and Z-Score approaches with the original there is little change. MinMax does not handle outliers very well. The scaling chosen was Z-Score as it slightly edges the original evaluation results.

For dataset 2, we compared the same approaches as in 1, given that both had better accuracys than the original but similar between each other, we opted to scale using Z-Score as it very slightly outperforms MinMax (0-1). **Shall not exceed 200 characters.**

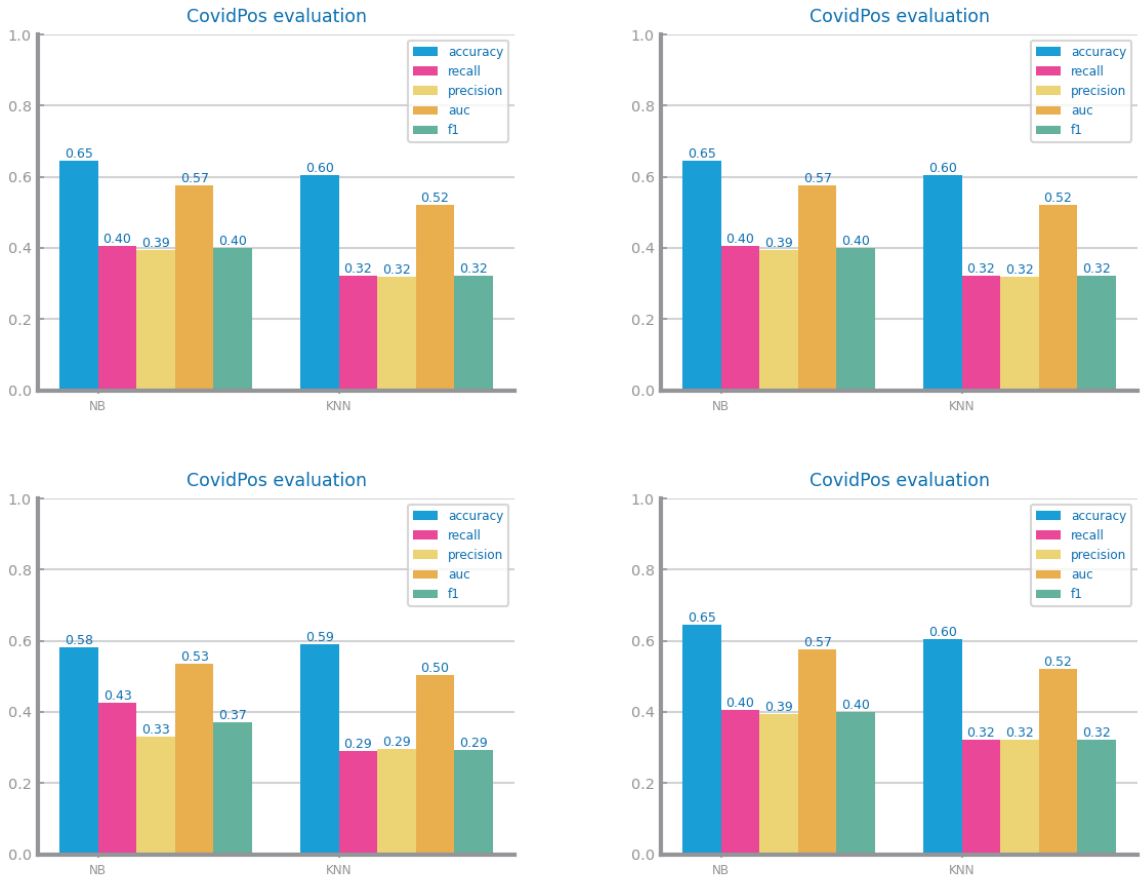


Figure 23: Scaling results with different approaches for dataset 1. MinMax[0,1] (top left), MinMax[0,10] (top right), Original (bottom left), Z-Score(bottom right)

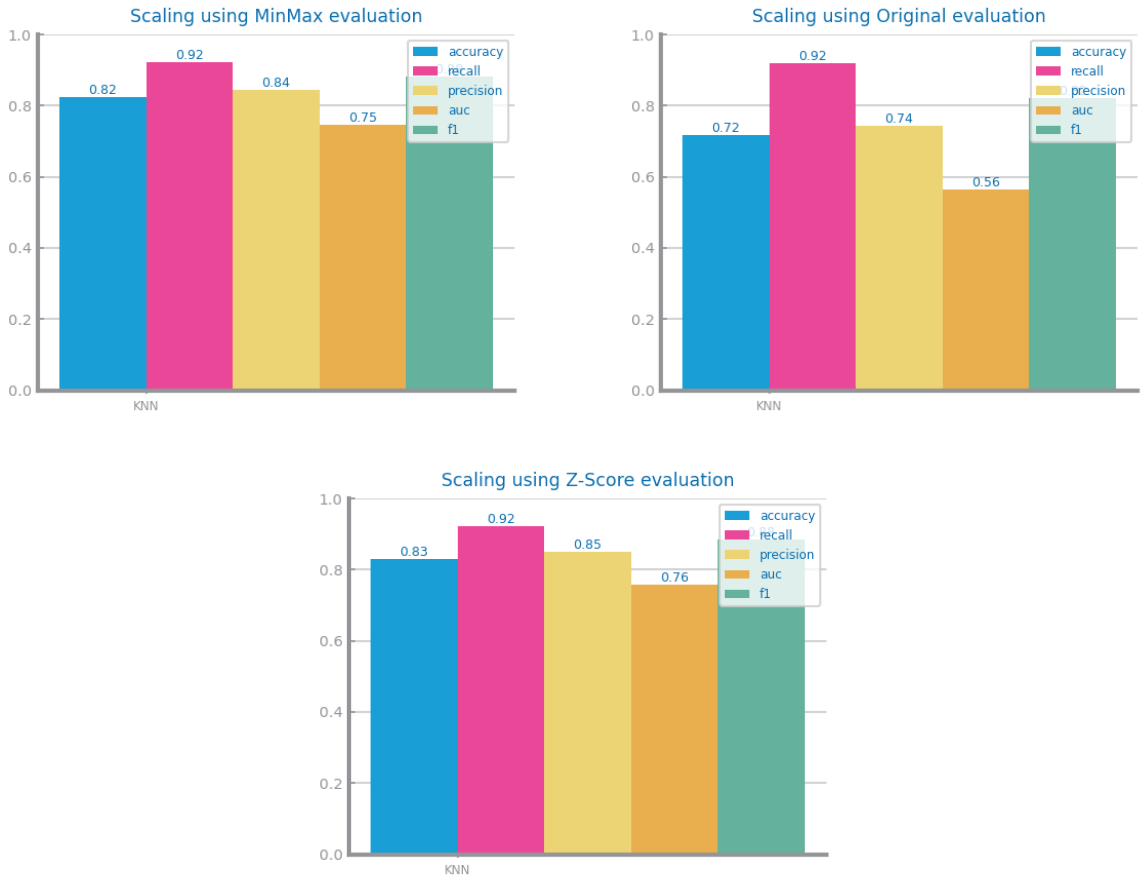


Figure 24: Scaling results with different approaches for dataset 2

Feature Selection

For ds 1 recall was chosen due to the data domain, we want false negatives to be more influencial than false positives. Z-Score turns all variance to 1, chosing a threshold of 1 worsens the results, 22 variables were too litlle relevant. Correlation changes don't affect the results, dropped 4 redundant variables.

In ds 2 no variable is to be dropped due to their relevancy, threshold lower than 1. In terms of redundacy the best result is as well to drop no variables, correlation threshold of 0.57. **Shall not exceed 500 characters.**

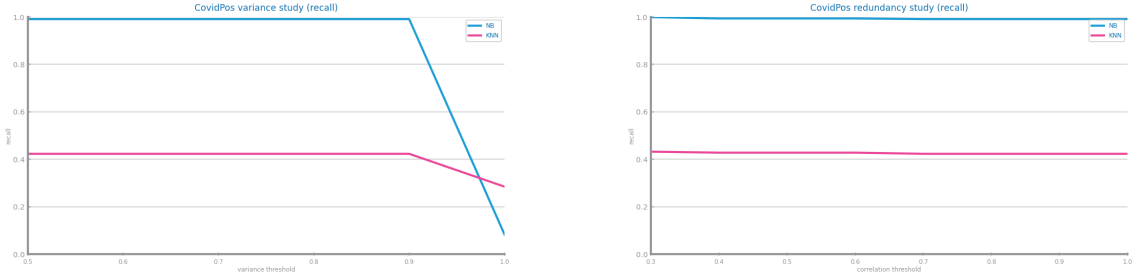


Figure 25: Feature selection of redundant variables results with different parameters for dataset 1

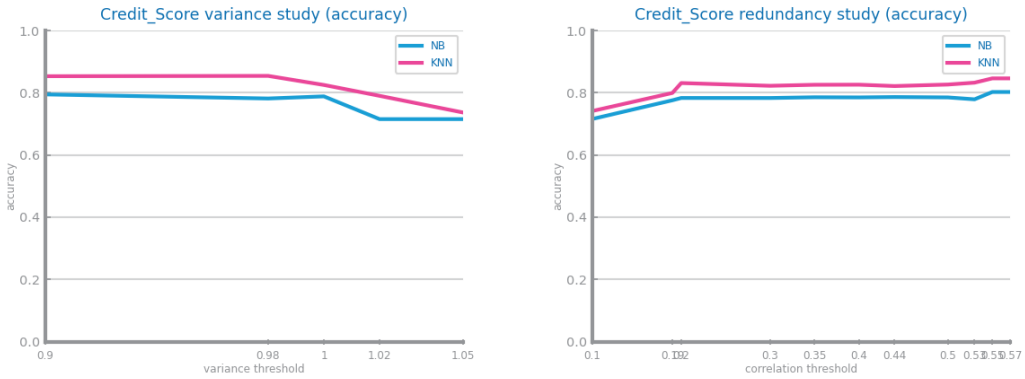


Figure 26: Feature selection of redundant variables results with different parameters for dataset 2

Balancing

For dataset 1, the SMOTE presented interesting results for recall, but the model was evidently flawed. So we chose to balance the data with undersampling, as the set contains a lot of noise (at least 16% of the records have an outlier value).

We chose SMOTE balancing transformation for ds 2. **Shall not exceed 500 characters.**

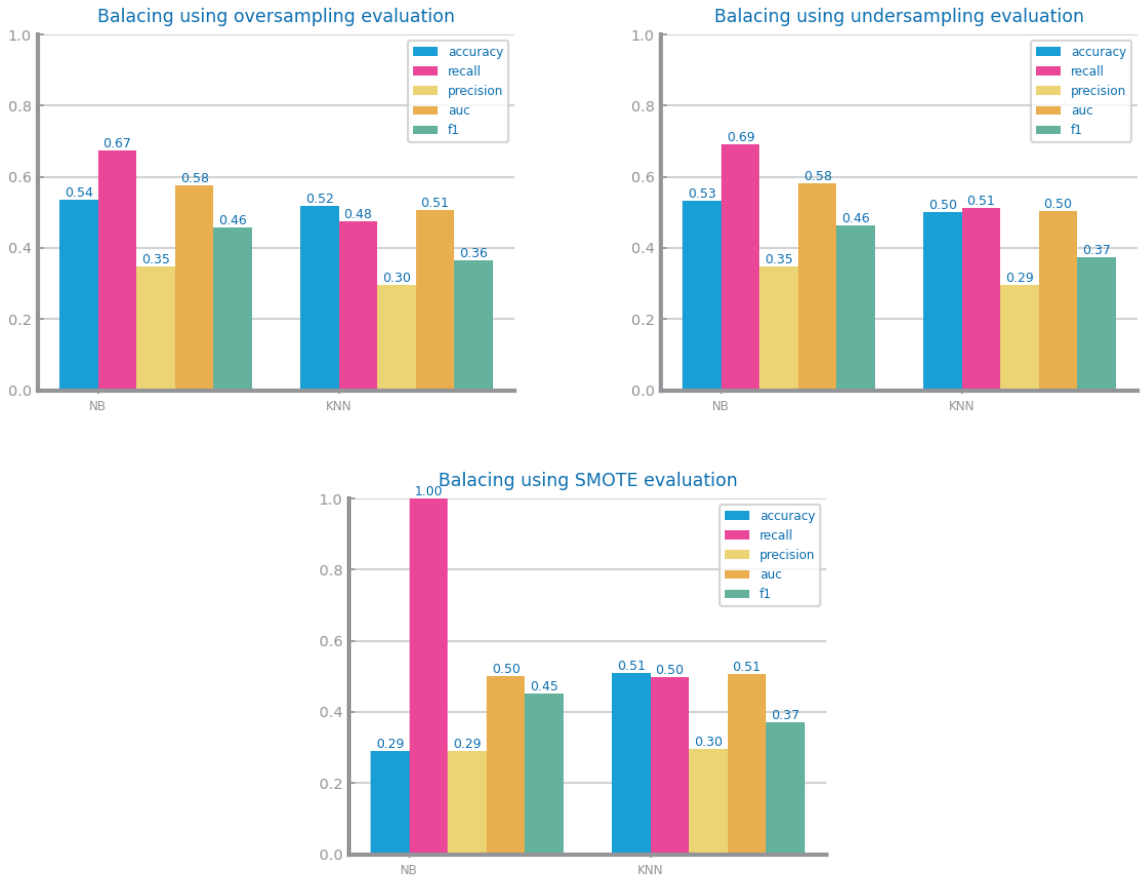


Figure 27: Balancing results with different approaches for dataset 1

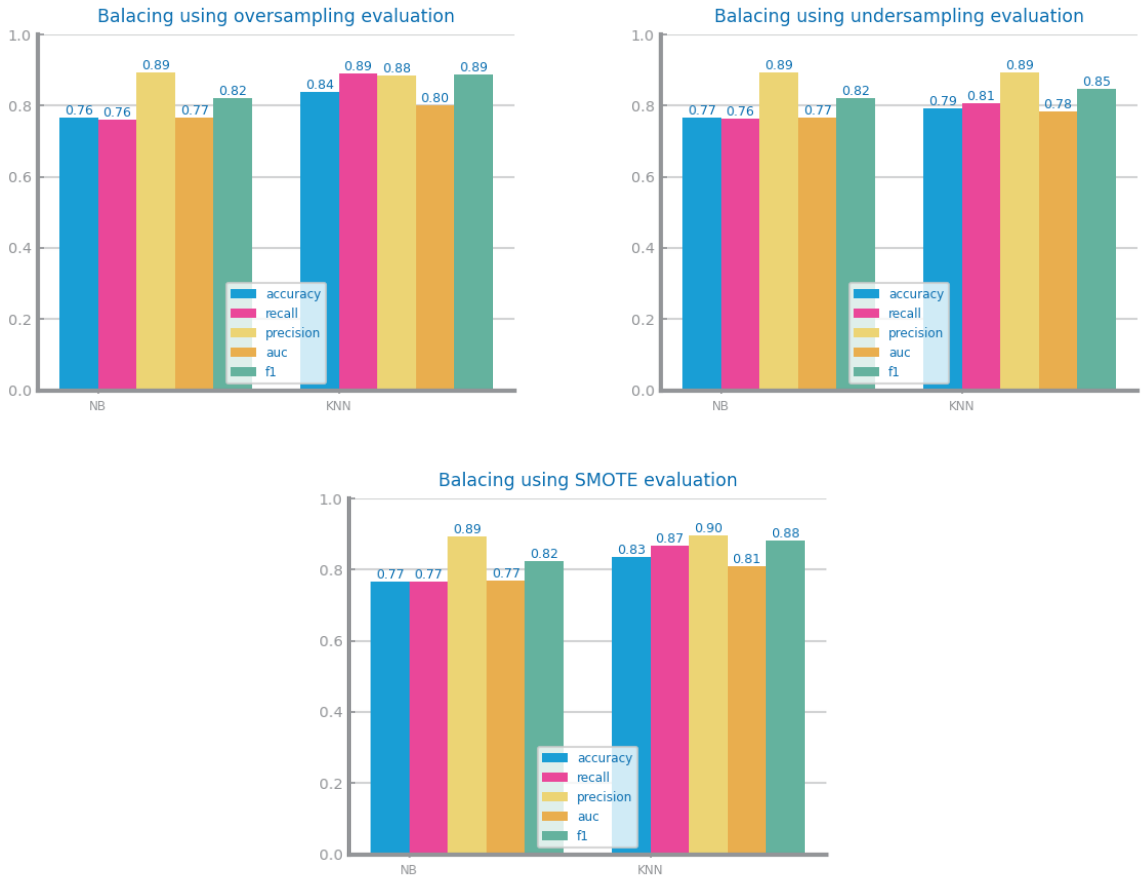


Figure 28: Balancing results with different approaches for dataset 2

3 MODELS' EVALUATION

For metric, we pursued recall for the dataset 1, since this means the system values more avoiding FN and maximizing TP. This is ideal because, as we are talking about health, it is always better to be safe than sorry.

For the dataset 2, we chose accuracy as the statistic, because, in this case, it makes sense to have system where we aim to get maximize TP and TN classifications, since a bad decision has a very high cost for the business.

In both datasets we used the hold-out strategy for training.

Shall be used to point out any important decision taken during the training, including training strategy and evaluation measures used. **Shall not exceed 500 characters.**

Naïve Bayes

For the dataset 1, the Gaussian NB presents the better value, since the majority of numerical variables follow a normal distribution.

For the dataset 2, the Bernoulli NB provides the best results, as the dataset is pretty sparse overall, thus making it harder to follow a normal distribution.

Shall be used to present the results achieved with each one of Naïve Bayes implementations, comparing and proposing explanations for them. If any of the implementations is not used, a justification for it shall be presented. Shall be used to present the evaluation of the best model achieved. **Shall not exceed 300 characters.**

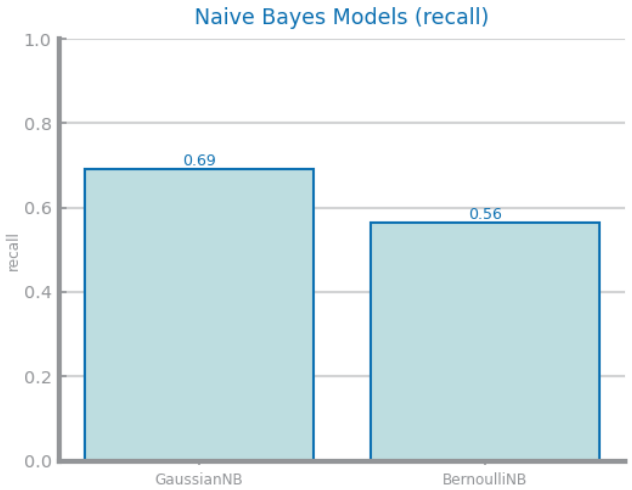


Figure 29: Naïve Bayes alternatives comparison for dataset 1

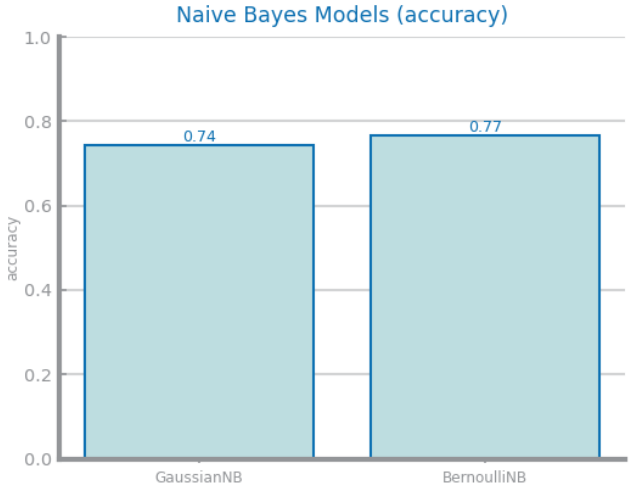


Figure 30: Naïve Bayes alternative comparison for dataset 2

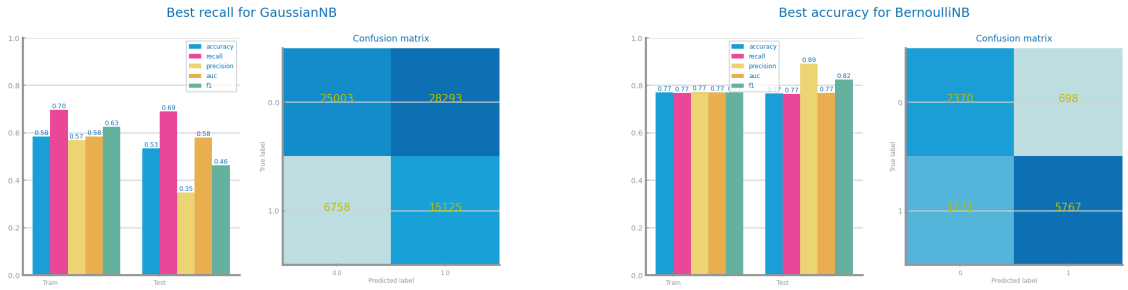


Figure 31: Naïve Bayes best model results for dataset 1 (left) and dataset 2 (right)

KNN

For both datasets we decided to limit the search for the best model to a maximum of k=25, due to the computation complexity of the algorithm.

The best parametrization for the dataset 1 is the Chebyshev, and the model chosen was the k=1, although model is more unstable and overfitted.

For dataset 2, the best parametrization was the Manhattan with model chosen was k=3. This model is more stable and less specialized than it's previous iterations, resulting in a lower probability for overfitting.

Shall be used to present the results achieved through different similarity measures and KNN parameterisations. The results shall be compared and explanations for them shall be presented. The justification for the chosen similarity measures shall be presented. Shall be used to address the overfitting phenomenon, studying the conditions under which models face it. Shall be used to present the evaluation of the best model achieved. **Shall not exceed 500 characters.**

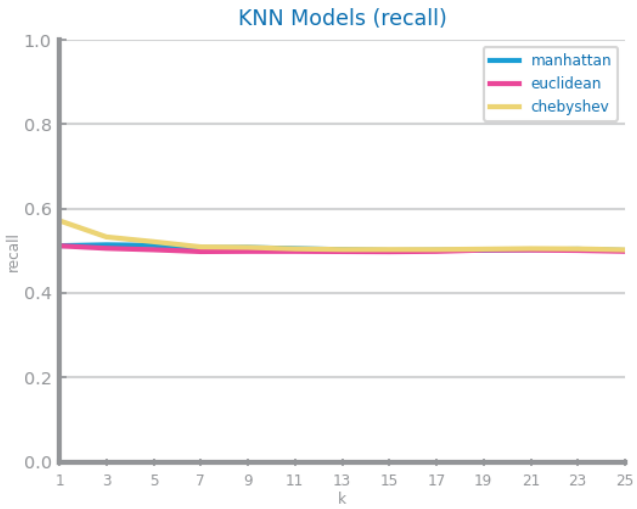


Figure 32: KNN different parameterisations comparison for dataset 1

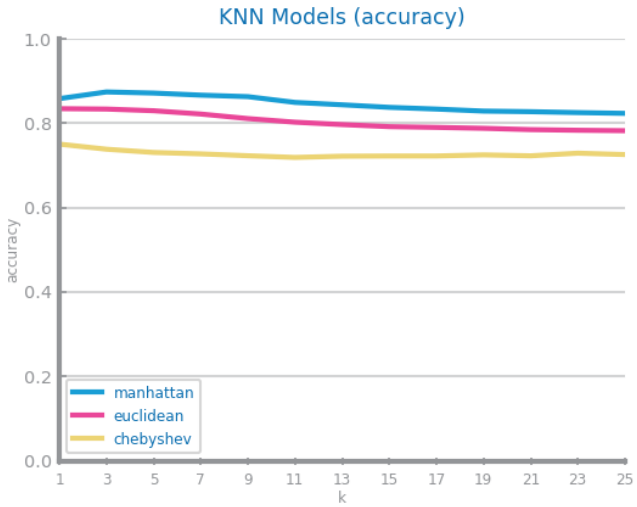


Figure 33: KNN different parameterisations comparison for dataset 2

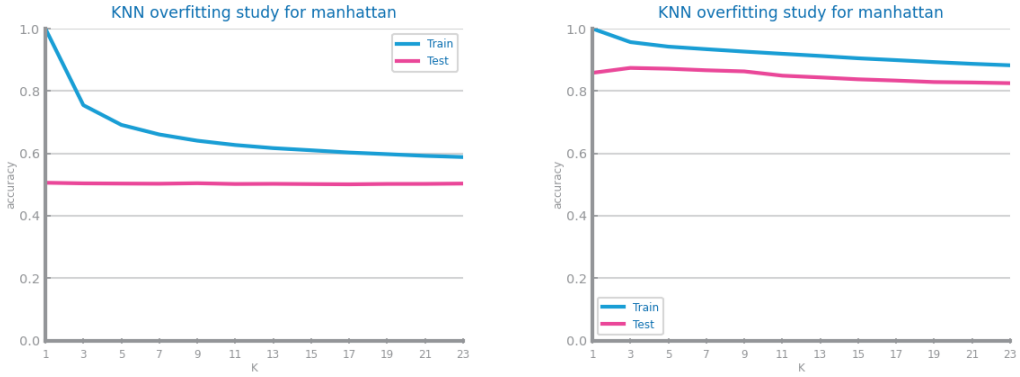


Figure 34: KNN overfitting analysis for dataset 1 (left) and dataset 2 (right)

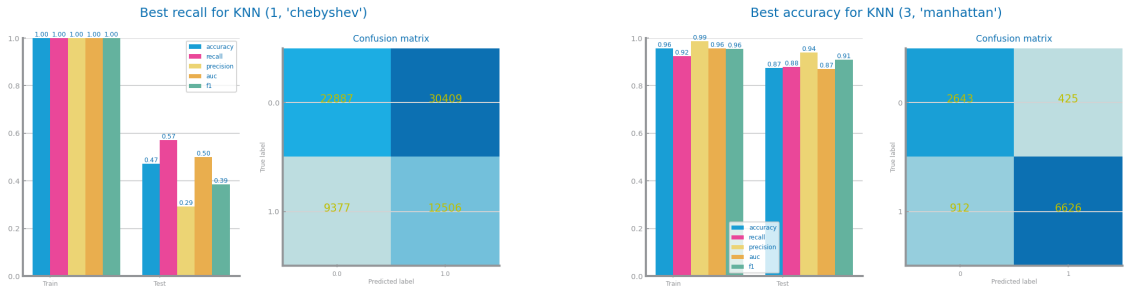


Figure 35: KNN best model results for dataset 1 (left) and dataset 2 (right)

Decision Trees

For both datasets we limited the search to a maximum depth of 25 due to the computation complexity of the algorithm.

For dataset 1, the models stopped improving the test accuracy after depth=9, with the best model using entropy criterion and depth=6.

For dataset 2, the best model is gini criterion with depth=14, having the test accuracy peaked at depth=14.

For both datasets, within our limited testing, there is no overfitting, but there is an upwards trend suggesting it may happen after depth=25.

Shall be used to present the results achieved through different parameterisations for the train of decision trees. The results shall be compared and explanations for them shall be presented. Shall be used to address the overfitting phenomenon, studying the conditions under which models face it. Shall be used to present the evaluation of the best model achieved. Shall be used to present the best tree achieved and its succinct description. **Shall not exceed 500 characters.**

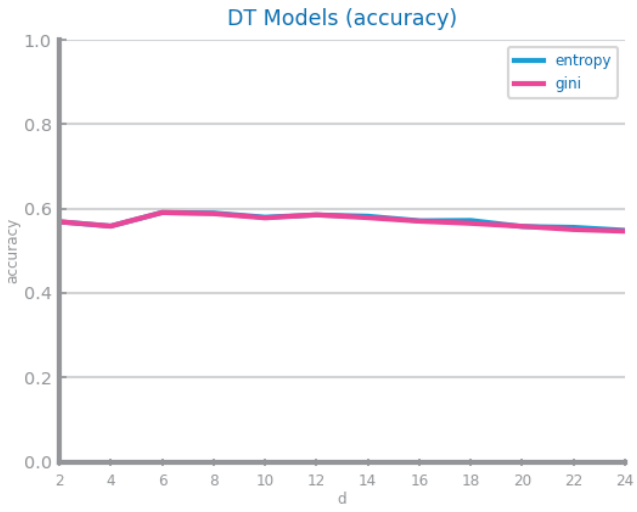


Figure 36: Decision Trees different parameterisations comparison for dataset 1

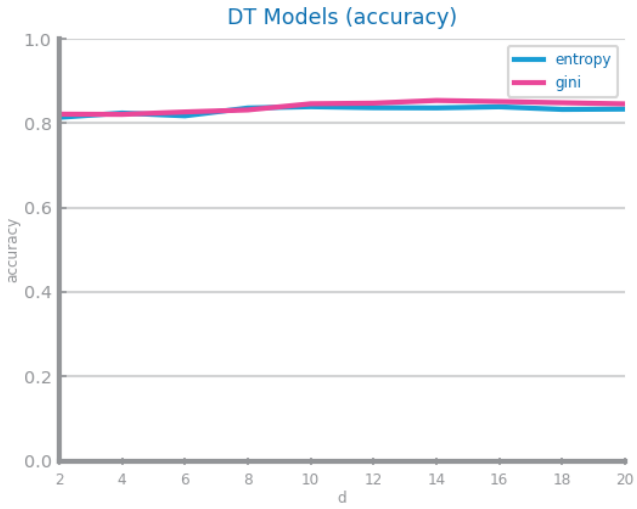


Figure 37: Decision Trees different parameterisations comparison for dataset 2

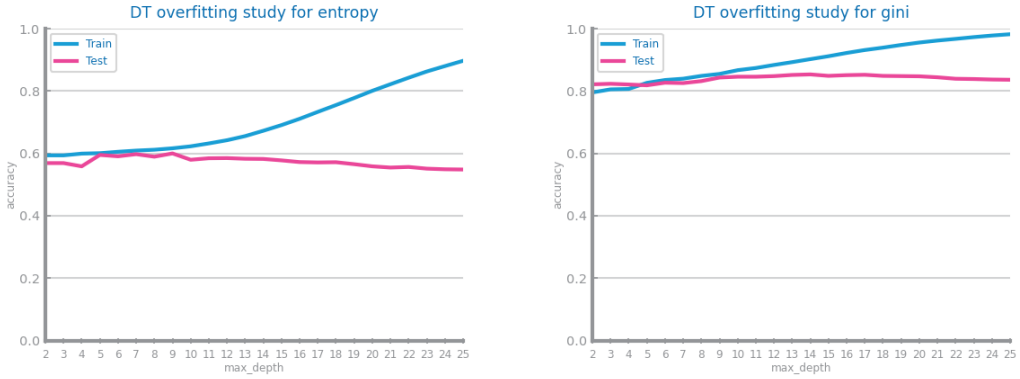


Figure 38: Decision Trees overfitting analysis for dataset 1 (left) and dataset 2 (right)

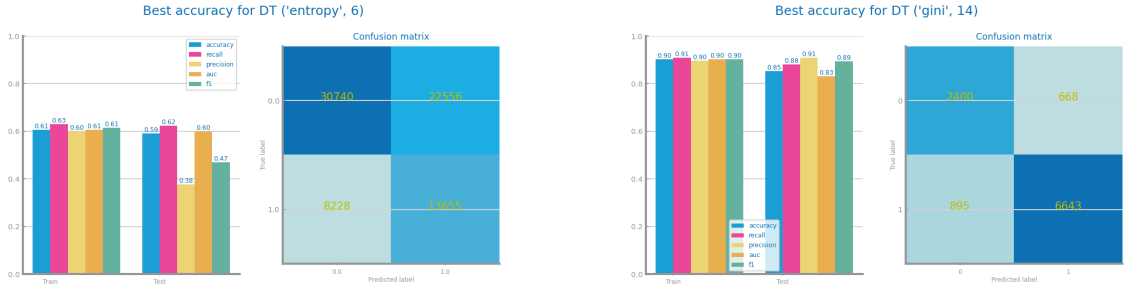


Figure 39: Decision trees best model results for dataset 1 (left) and dataset 2 (right)

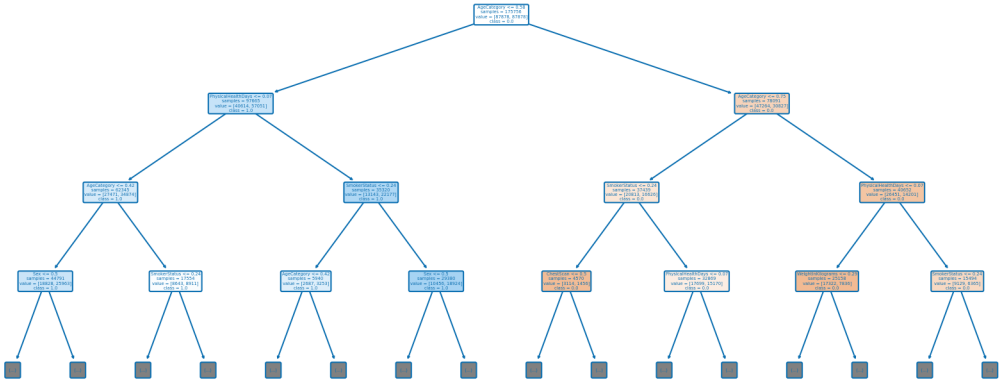


Figure 40: Best tree for dataset 1

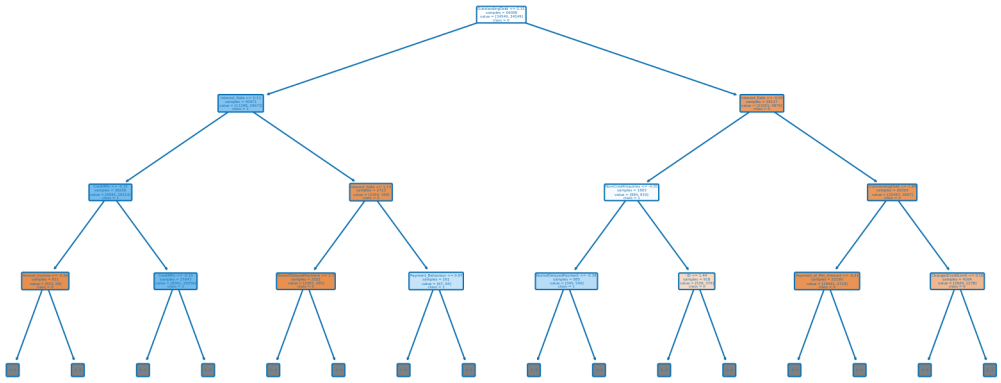
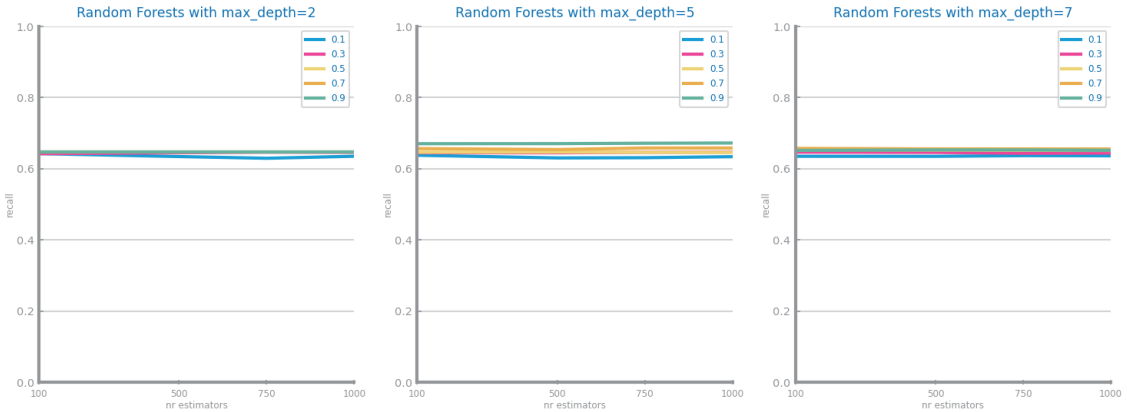
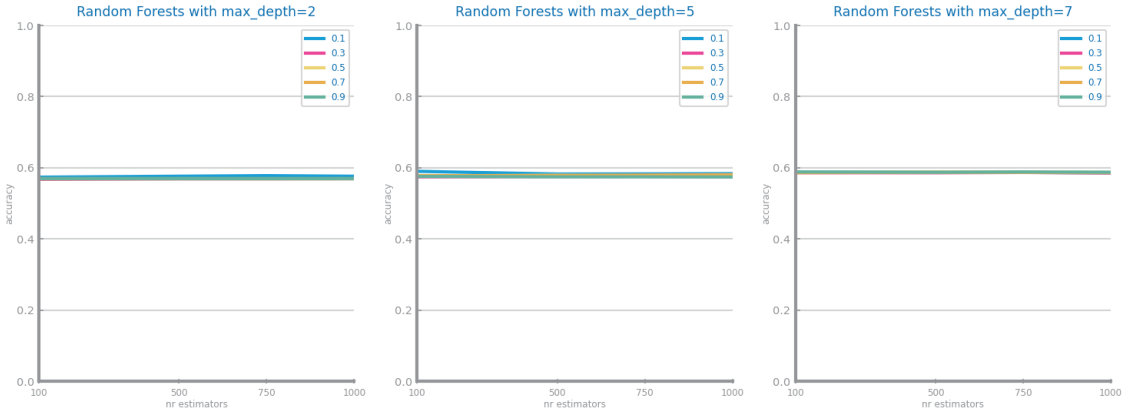


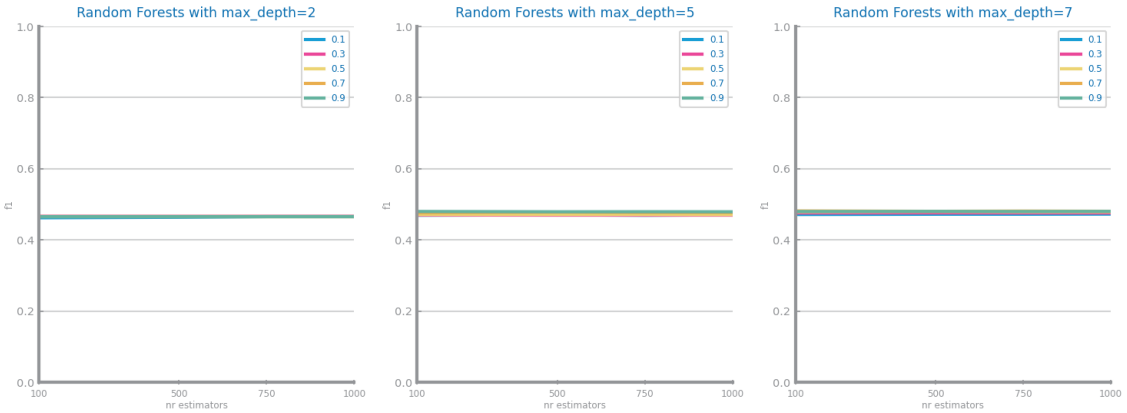
Figure 41: Best tree for dataset 2

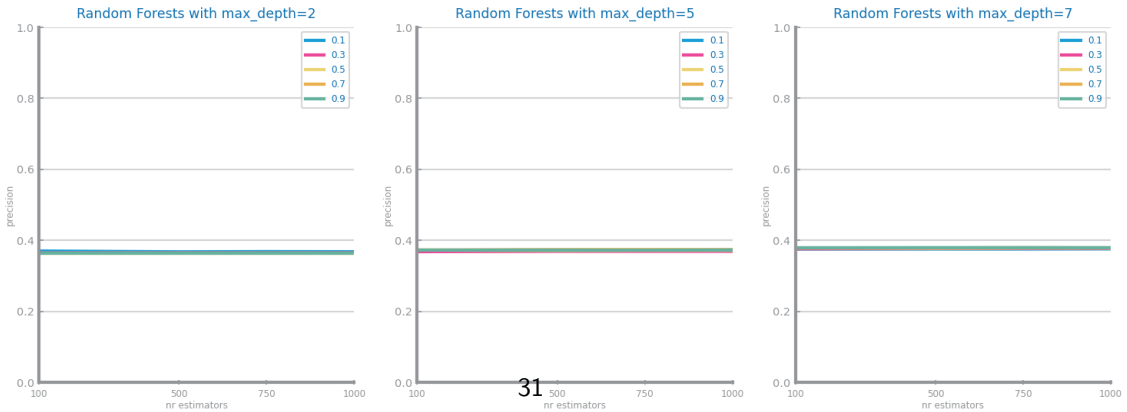
Random Forests

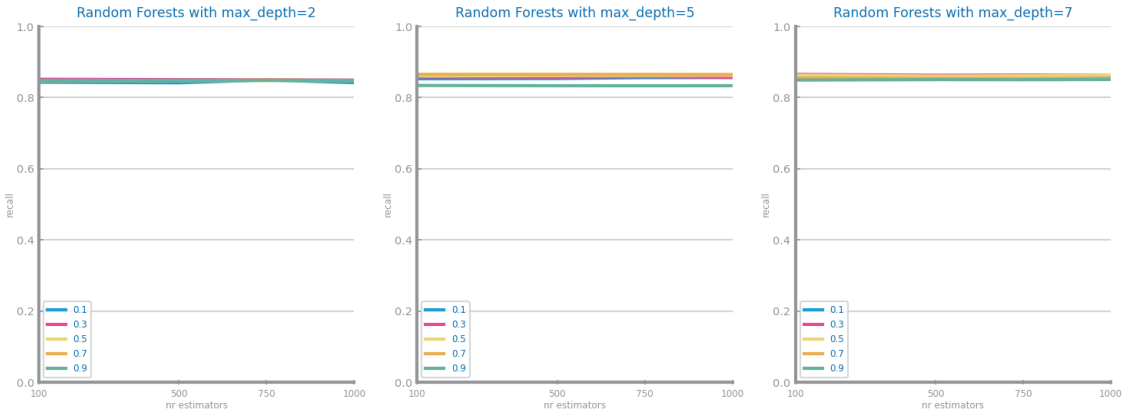
Shall be used to present the results achieved through different parameterisations for the train of random forests. The results shall be compared and explanations for them shall be presented. Shall be used to address the overfitting phenomenon, studying the conditions under which models face it. Shall be used to present the evaluation of the best model achieved. May be used to present the most important variables in the model. **Shall not exceed 500 characters.**

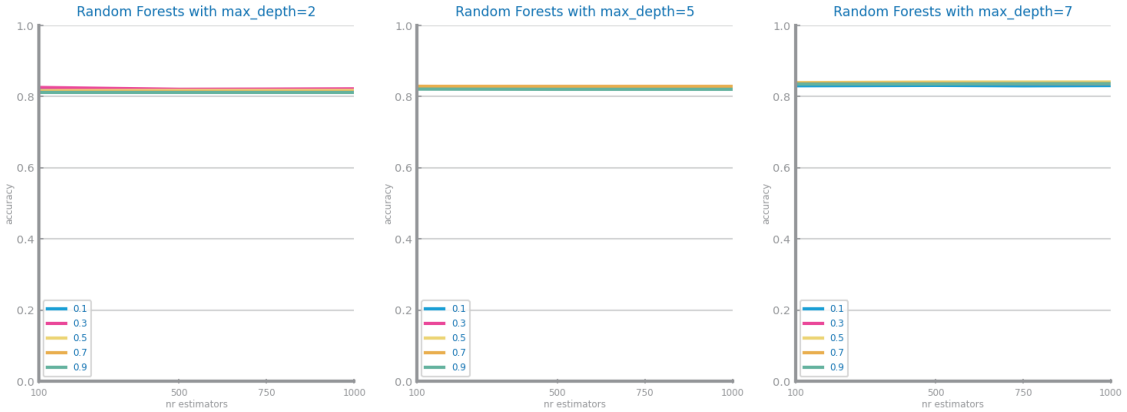

Random Forests with max_depth=2
Random Forests with max_depth=5
Random Forests with max_depth=7
recall
nr estimators
0.1
0.3
0.5
0.7
0.9

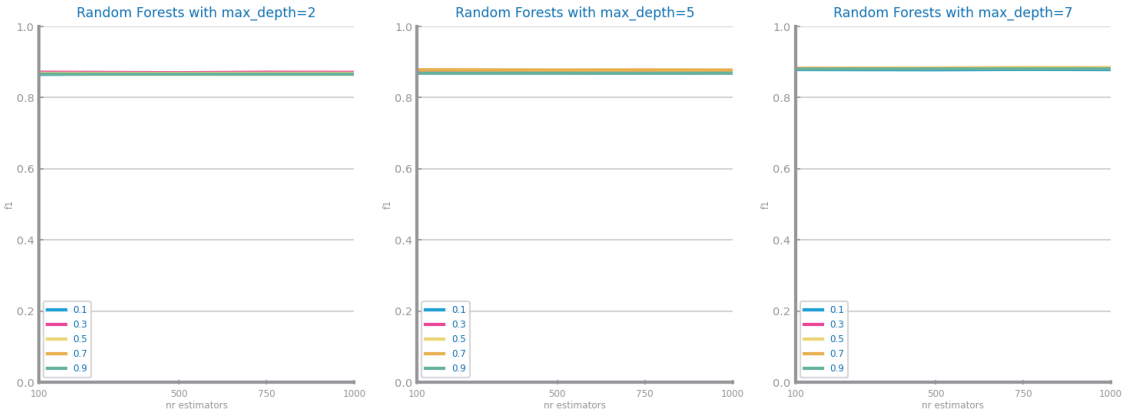

Random Forests with max_depth=2
Random Forests with max_depth=5
Random Forests with max_depth=7
accuracy
nr estimators
0.1
0.3
0.5
0.7
0.9

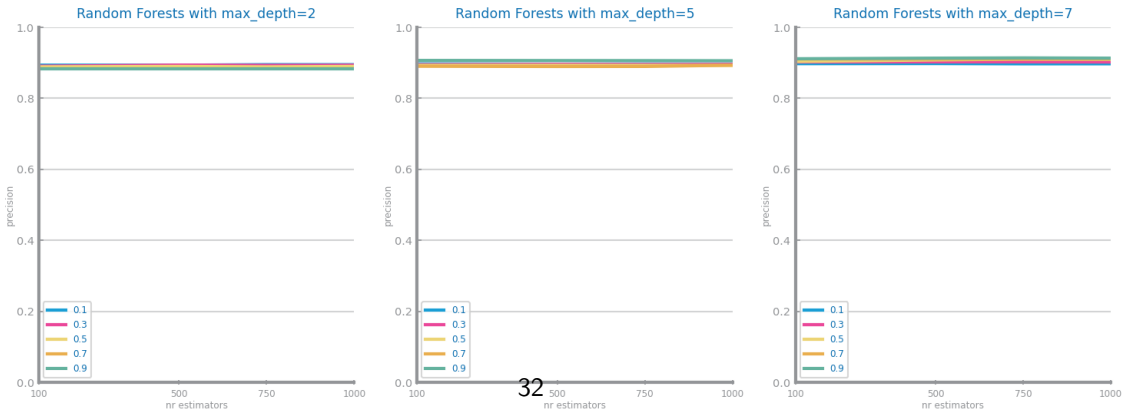

Random Forests with max_depth=2
Random Forests with max_depth=5
Random Forests with max_depth=7
f1
nr estimators
0.1
0.3
0.5
0.7
0.9


Random Forests with max_depth=2
Random Forests with max_depth=5
Random Forests with max_depth=7
precision
nr estimators
0.1
0.3
0.5
0.7
0.9


Random Forests with max_depth=2
Random Forests with max_depth=5
Random Forests with max_depth=7
recall
nr estimators
0.1
0.3
0.5
0.7
0.9


Random Forests with max_depth=2
Random Forests with max_depth=5
Random Forests with max_depth=7
accuracy
nr estimators
0.1
0.3
0.5
0.7
0.9


Random Forests with max_depth=2
Random Forests with max_depth=5
Random Forests with max_depth=7
f1
nr estimators
0.1
0.3
0.5
0.7
0.9


Random Forests with max_depth=2
Random Forests with max_depth=5
Random Forests with max_depth=7
precision
nr estimators
0.1
0.3
0.5
0.7
0.9

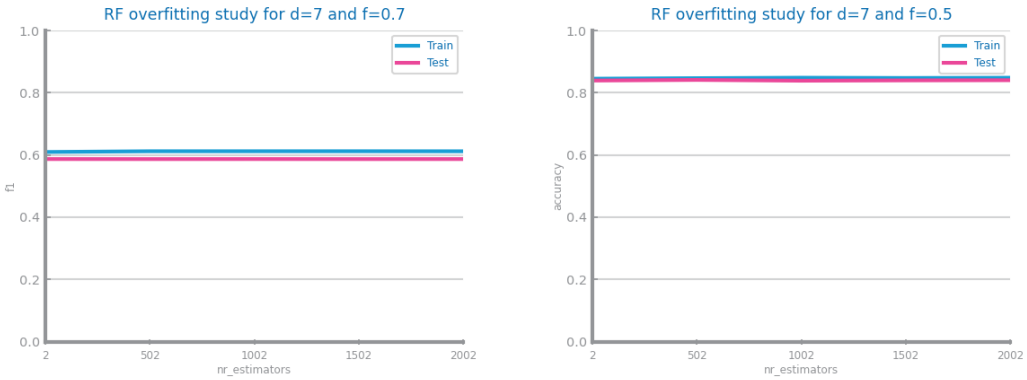


Figure 44: Random Forests overfitting analysis for dataset 1 (left) and dataset 2 (right)

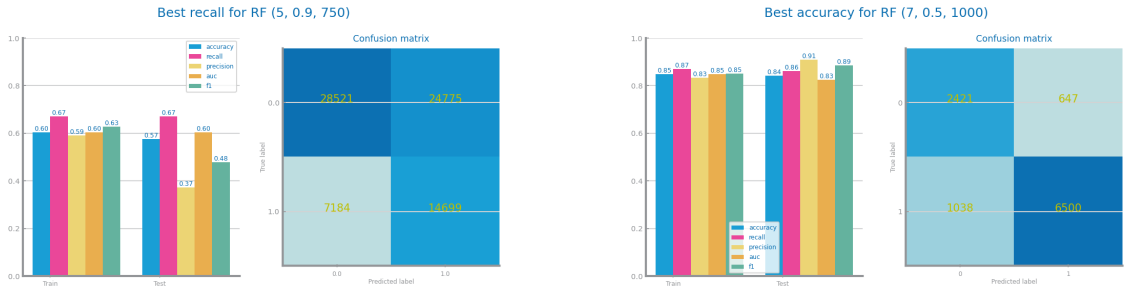


Figure 45: Random Forests best model results for dataset 1 (left) and dataset 2 (right)

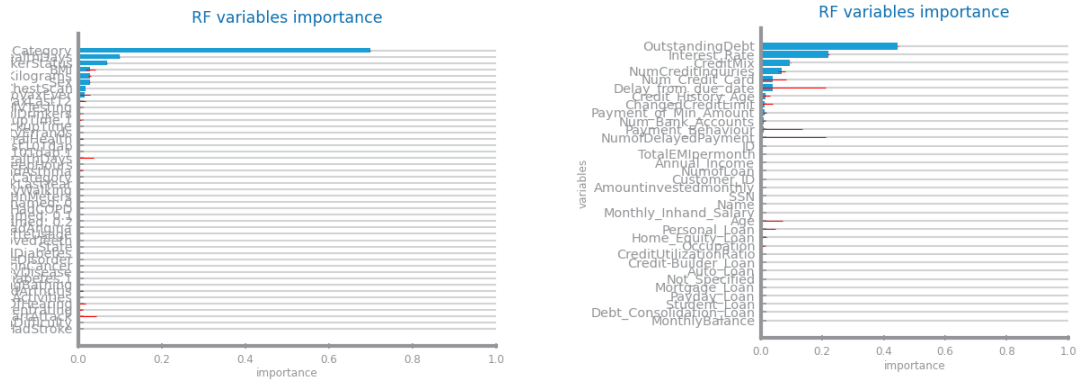
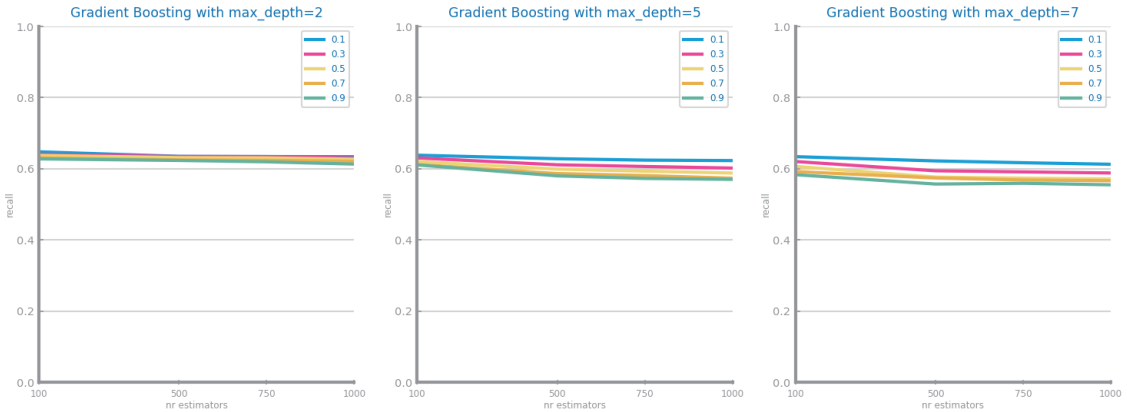
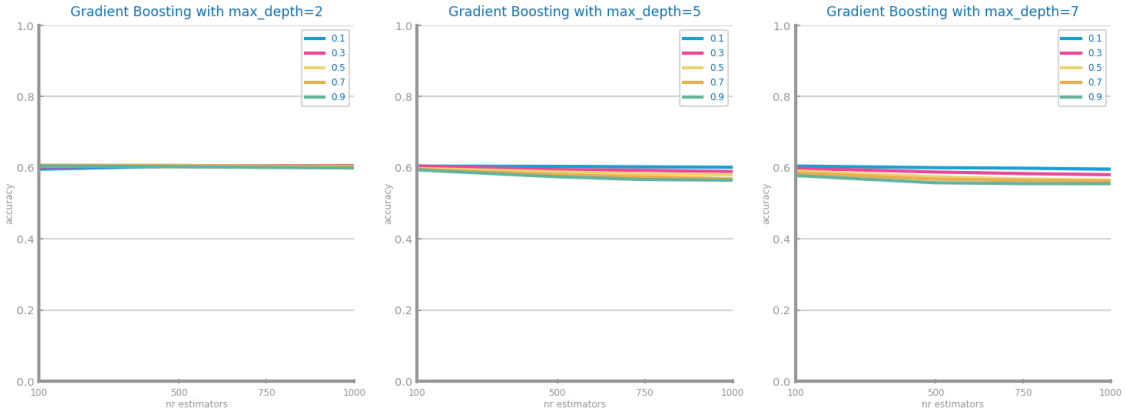


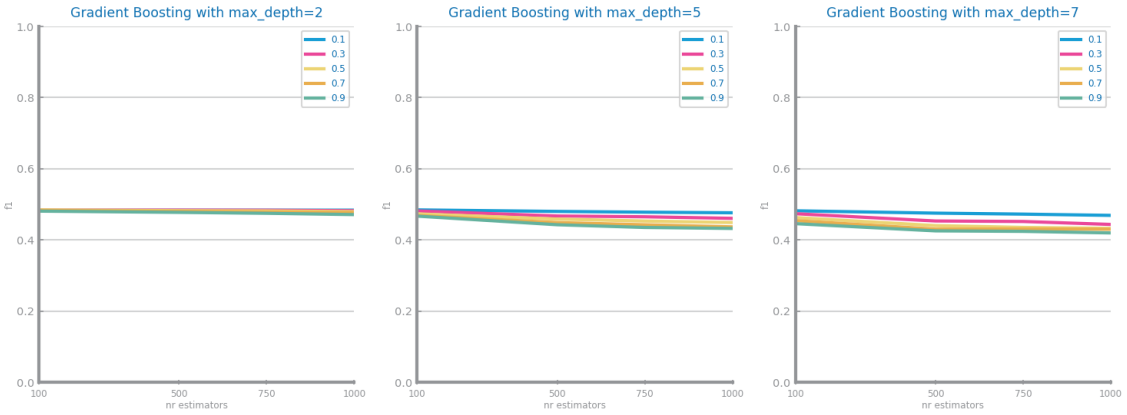
Figure 46: Random Forests variables importance for dataset 1 (left) and dataset 2 (right)

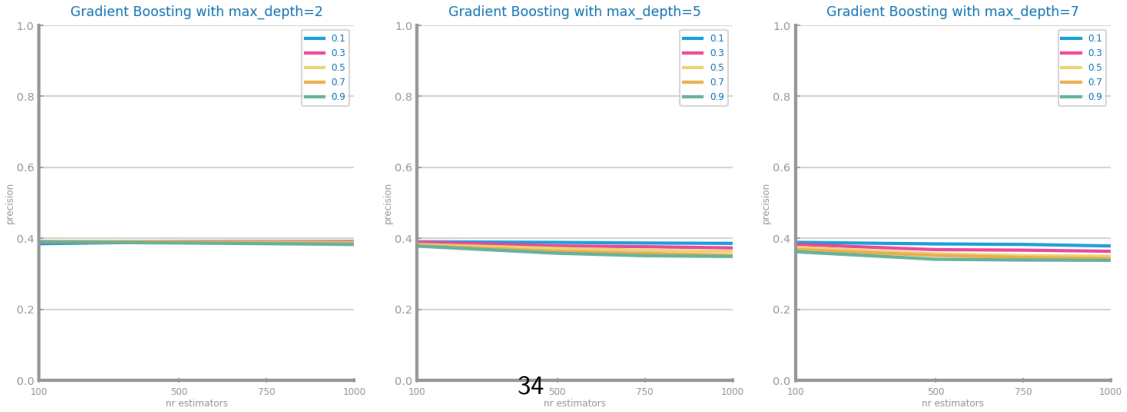
Gradient Boosting

Shall be used to present the results achieved through different parameterisations for the train of gradient boosting. The results shall be compared and explanations for them shall be presented. Shall be used to address the overfitting phenomenon, studying the conditions under which models face it. Shall be used to present the evaluation of the best model achieved. May be used to present the most important variables in the model. **Shall not exceed 500 characters.**

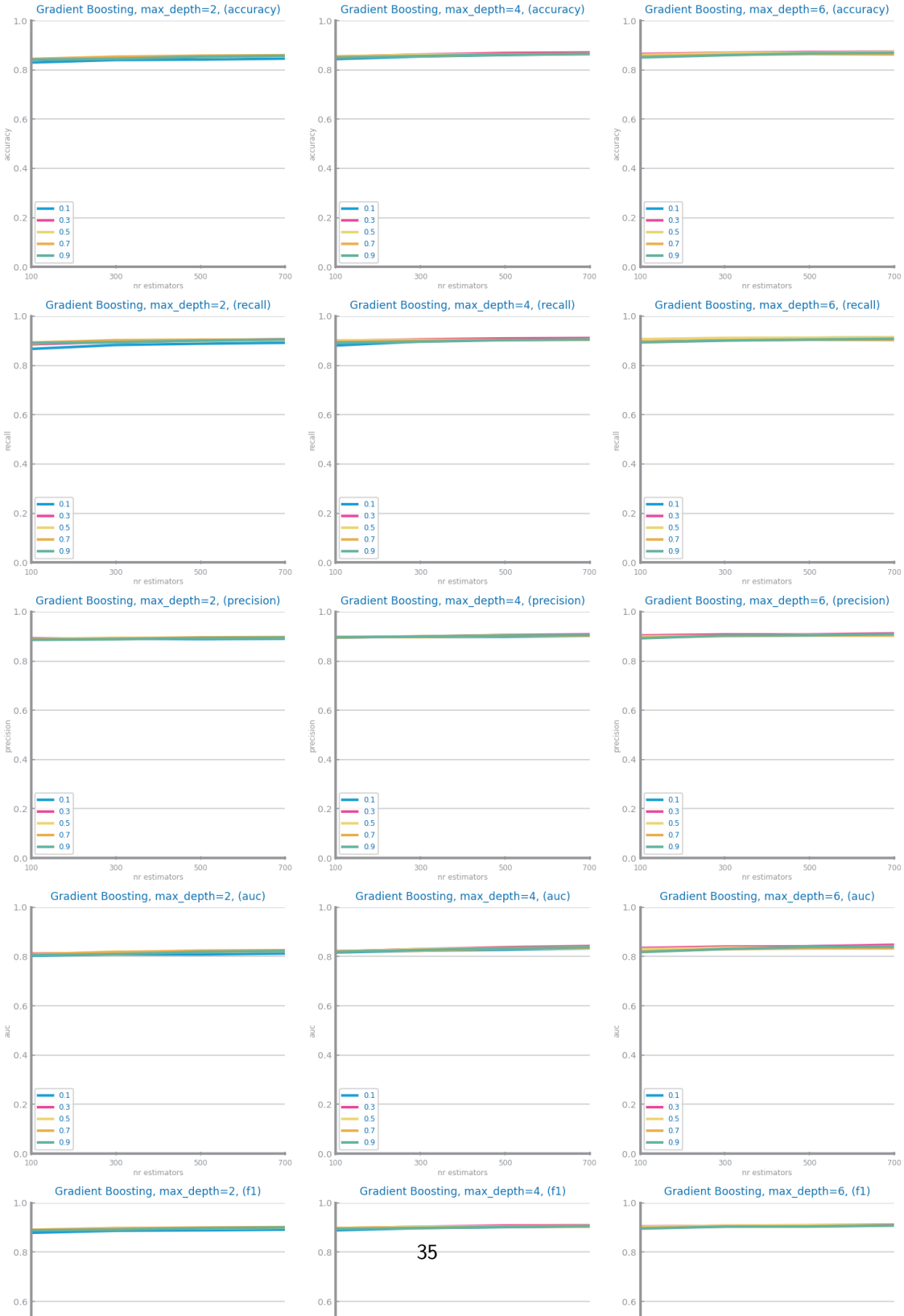

Gradient Boosting with max_depth=2
Gradient Boosting with max_depth=5
Gradient Boosting with max_depth=7
recall
nr estimators
0.1
0.3
0.5
0.7
0.9


Gradient Boosting with max_depth=2
Gradient Boosting with max_depth=5
Gradient Boosting with max_depth=7
accuracy
nr estimators
0.1
0.3
0.5
0.7
0.9


Gradient Boosting with max_depth=2
Gradient Boosting with max_depth=5
Gradient Boosting with max_depth=7
f1
nr estimators
0.1
0.3
0.5
0.7
0.9


Gradient Boosting with max_depth=2
Gradient Boosting with max_depth=5
Gradient Boosting with max_depth=7
precision
nr estimators
0.1
0.3
0.5
0.7
0.9

Gradient Boosting study for different parameters



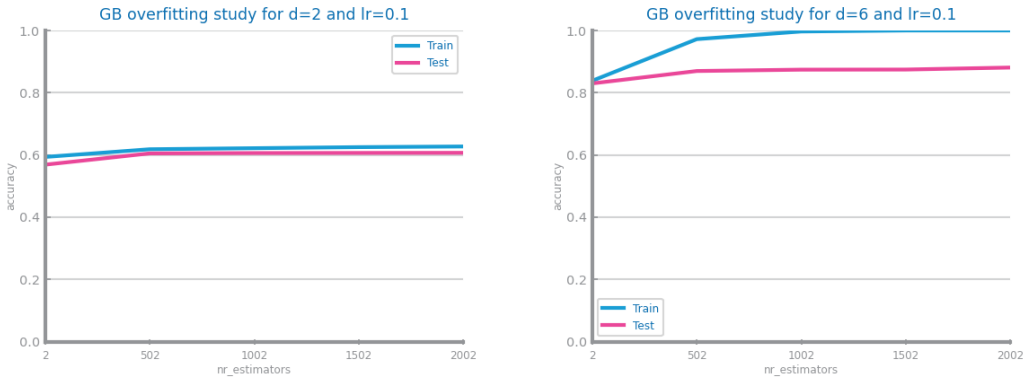


Figure 49: Gradient boosting overfitting analysis for dataset 1 (left) and dataset 2 (right)

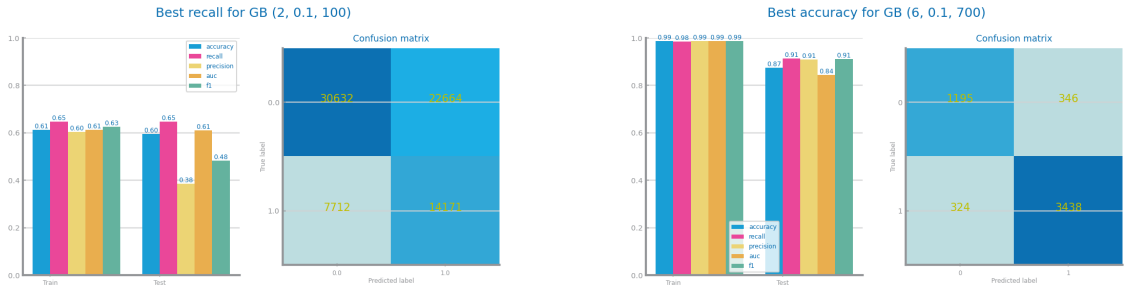


Figure 50: Gradient boosting best model results for dataset 1 (left) and dataset 2 (right)

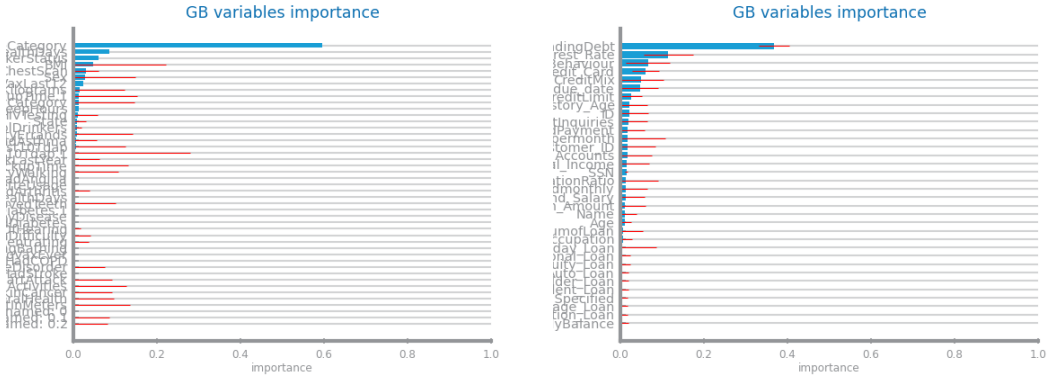


Figure 51: Gradient boosting variables importance for dataset 1 (left) and dataset 2 (right)

Multi-Layer Perceptrons

For dataset 1, the model learnt is not useful, with 1.0 recall, every predicted value is positive. This means that recall is clearly not the right measure to optimize. While the recall over the training set is constant, on the test it oscilates periodically with no evident improvement or decrease, could be due to high amounts of noise.

For dataset 2, although the model might be on a slight overfit, especially from the 800th iteration, the high values on all metrics indicate that it's a good model. **Shall not exceed 500 characters.**

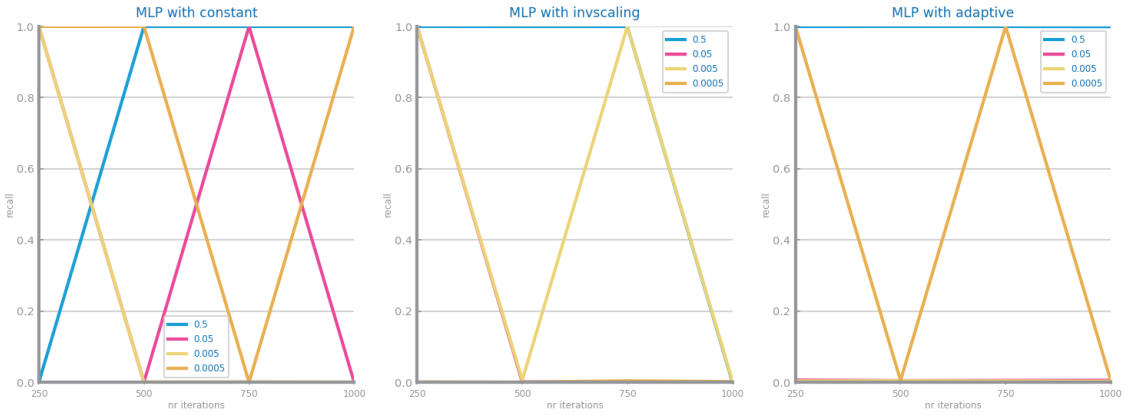


Figure 52: MLP different parameterisations comparison for dataset 1

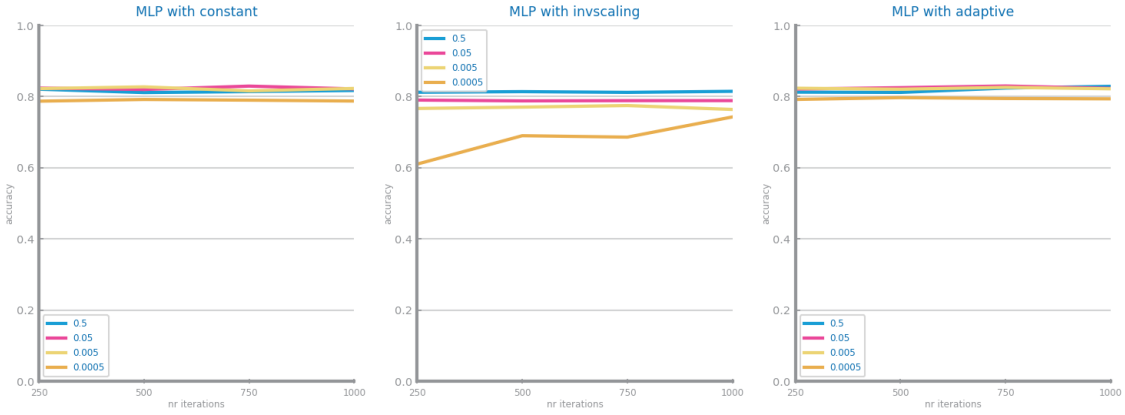


Figure 53: MLP different parameterisations comparison for dataset 2

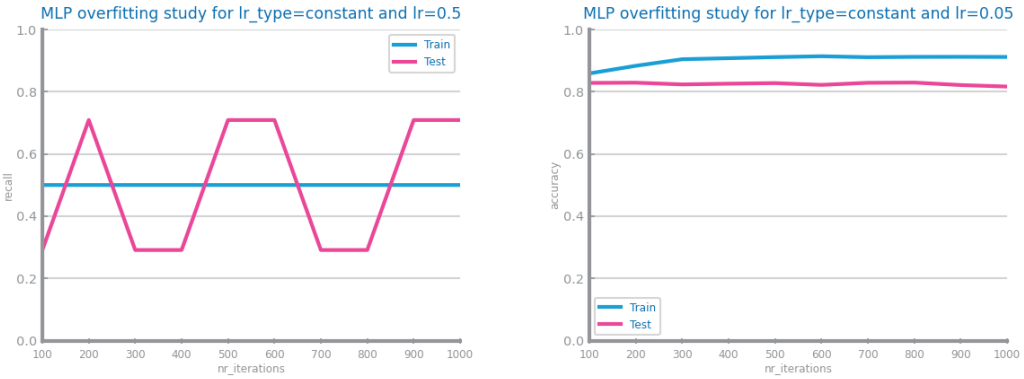


Figure 54: MLP overfitting analysis for dataset 1 (left) and dataset 2 (right)

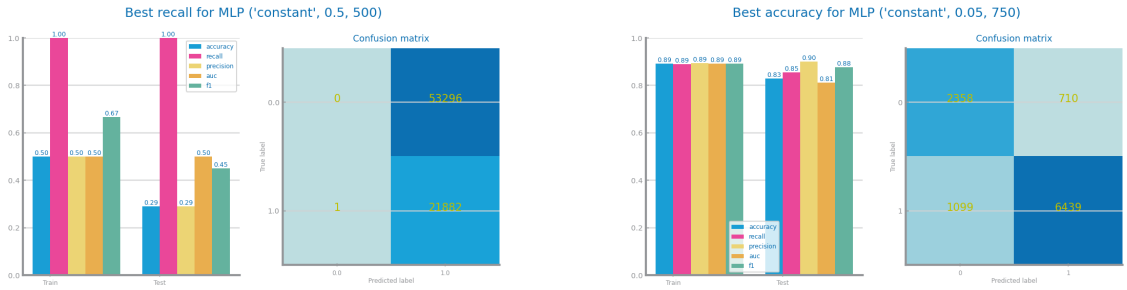


Figure 55: MLP best model results for dataset 1 (left) and dataset 2 (right)

4 CRITICAL ANALYSIS

Shall be used to present a summary of the results achieved with the different modelling techniques, and the impact of the different preparation tasks on their performance. A cross-analysis of the different models may also be presented, identifying the most relevant variables common to all of them (when possible) and the relation among the patterns identified within the different classifiers. A critical assessment of the best models shall be presented, clearly stating if the models seem to be good enough for the problem at hand. **Additional charts may be presented here. Shall not exceed 2000 characters.**

remember that an outlier is a point that is far away from the regular ones. Given that for all but the *MonthlyBalance* variable all the points are continuous, some care must be taken in their treatment.

TIME SERIES ANALYSIS

5 DATA PROFILING

Data Dimensionality and Granularity

We used the "sum" function as "agg_fun" for both datasets. We studied the granularity at three different levels, for dataset 1, weekly (atomic), monthly and quarterly with an upwards trend but no seasonality or cyclical behaviour.

For dataset 2 by 15 minutes (atomic), hourly and daily, with no visible trend, daily seasonality on each morning and evening and also weekly cyclicality with a busier day a week (usually Mondays except the first spike) both corresponding to heavier traffic flows. **Shall not exceed 500 characters.**

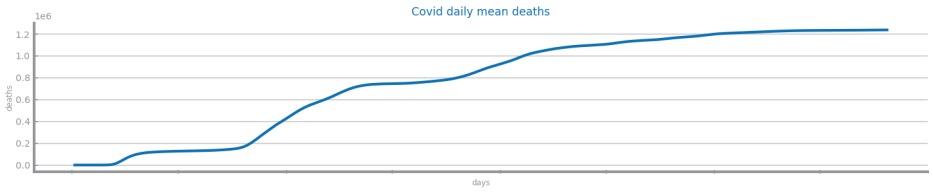


Figure 56: Time series 1 at the most granular detail

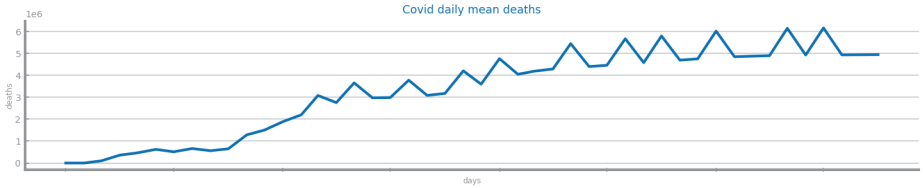


Figure 57: Time series 1 at the second chosen granularity

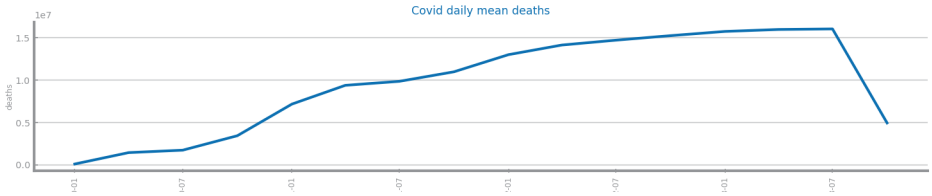


Figure 58: Time series 1 at the third chosen granularity

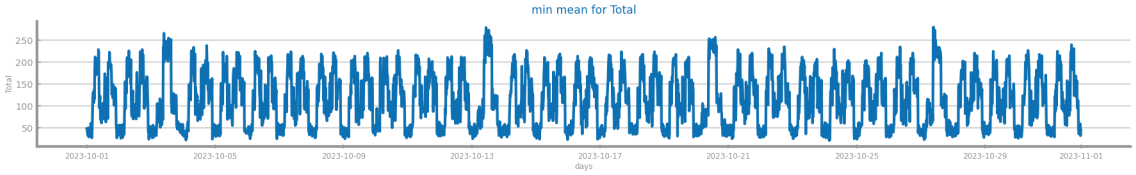


Figure 59: Time series 2 at the most granular detail

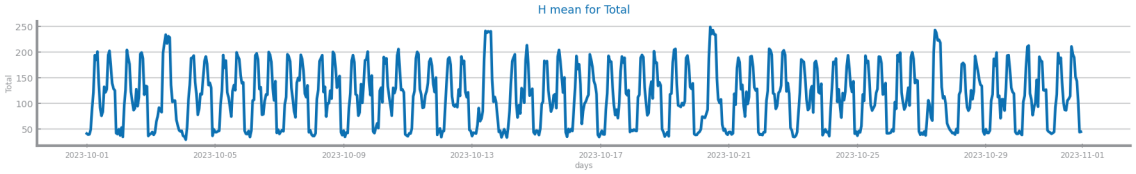


Figure 60: Time series 2 at the second chosen granularity

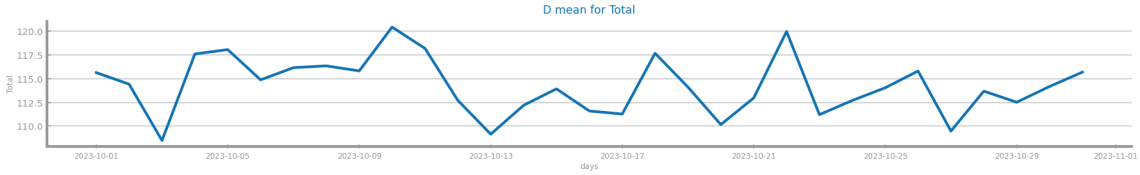
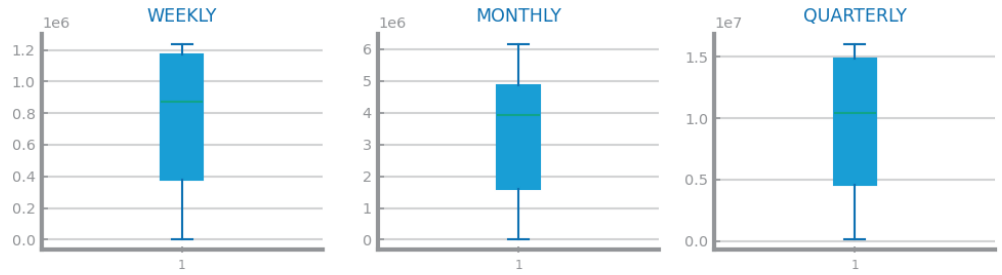


Figure 61: Time series 2 at the third chosen granularity

Data Distribution

Shall be used to perform the data analysis at those three different granularities, concerning the series distribution. **Shall not exceed 500 characters.**



count	1.990000e+02
mean	7.743876e+05
std	4.356312e+05
min	0.000000e+00
25%	3.775975e+05
50%	8.691990e+05
75%	1.172049e+06
max	1.238650e+06

Name: deaths, dtype: float64

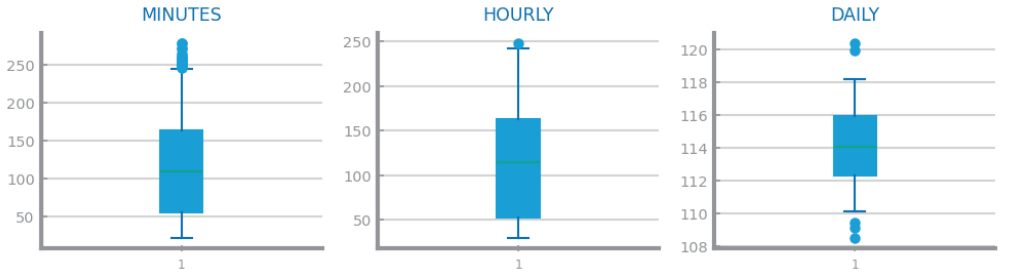
count	1.990000e+02
mean	7.743876e+05
std	4.356312e+05
min	0.000000e+00
25%	3.775975e+05
50%	8.691990e+05
75%	1.172049e+06
max	1.238650e+06

Name: deaths, dtype: float64

count	4.600000e+01
mean	3.350068e+06
std	1.937462e+06
min	1.000000e+00
25%	1.602803e+06
50%	3.919796e+06
75%	4.874868e+06
max	6.172614e+06

Name: deaths, dtype: float64

Figure 62: Boxplot(s) for time series 1



count	2976.000000
mean	114.218414
std	60.190627
min	21.000000
25%	55.000000
50%	109.000000
75%	164.000000
max	279.000000

Name: Total, dtype: float64

count	744.000000
mean	114.218414
std	56.144258
min	29.250000
25%	52.687500
50%	114.875000
75%	162.687500
max	248.750000

Name: Total, dtype: float64

count	31.000000
mean	114.218414
std	3.043082
min	108.489583
25%	112.348958
50%	114.104167
75%	115.973958
max	120.406250

Name: Total, dtype: float64

Figure 63: Boxplot(s) for time series 2

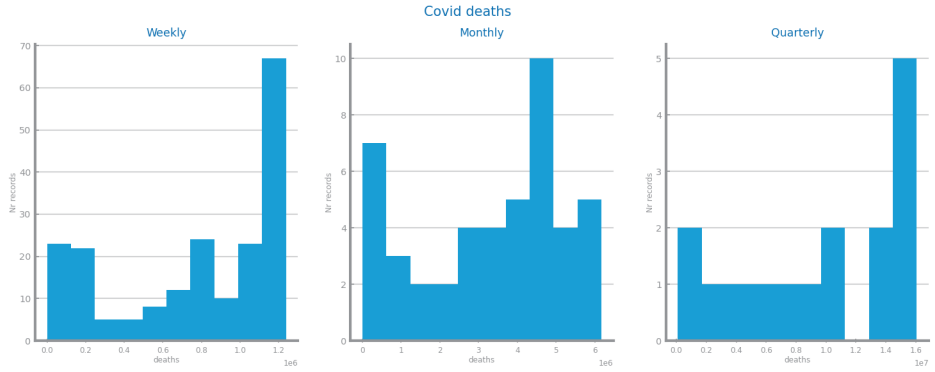


Figure 64: Histogram(s) for time series 1

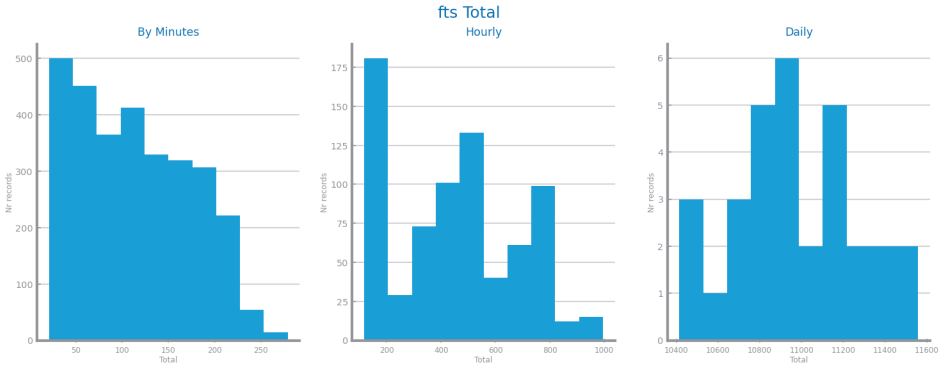


Figure 65: Histogram(s) for time series 2

Data Stationarity

For the 1st dataset we obtained a p-value of 0.223. Looking at the graphs, we can see that there is a trend in the first 2 and that there is evidence of a seasonal trend in the 3rd.

For the 2nd dataset we obtained a p-value of 0. and that there is no trend but there is some seasonality. **Shall not exceed 300 characters.**

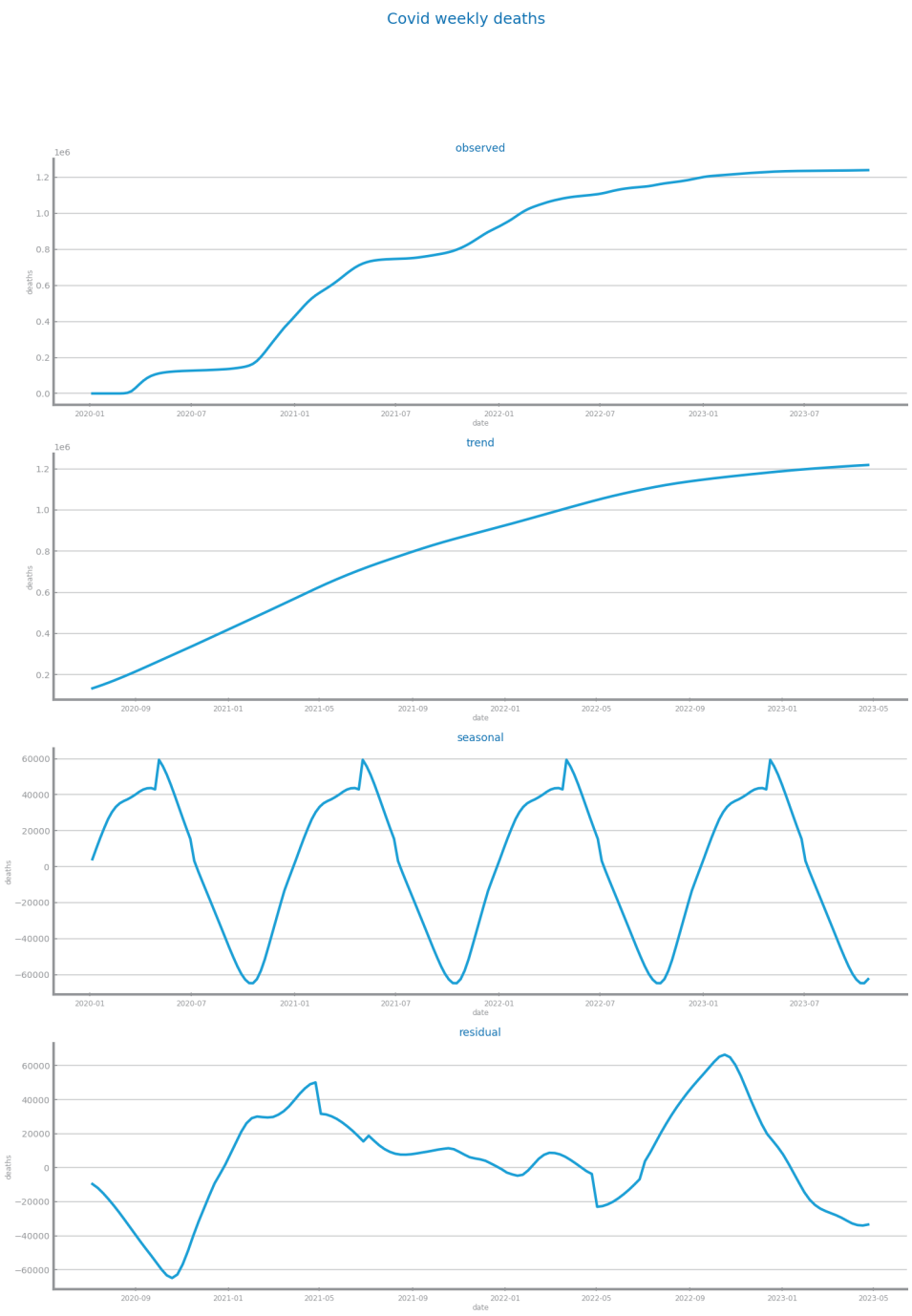


Figure 66: Components study for time series 1

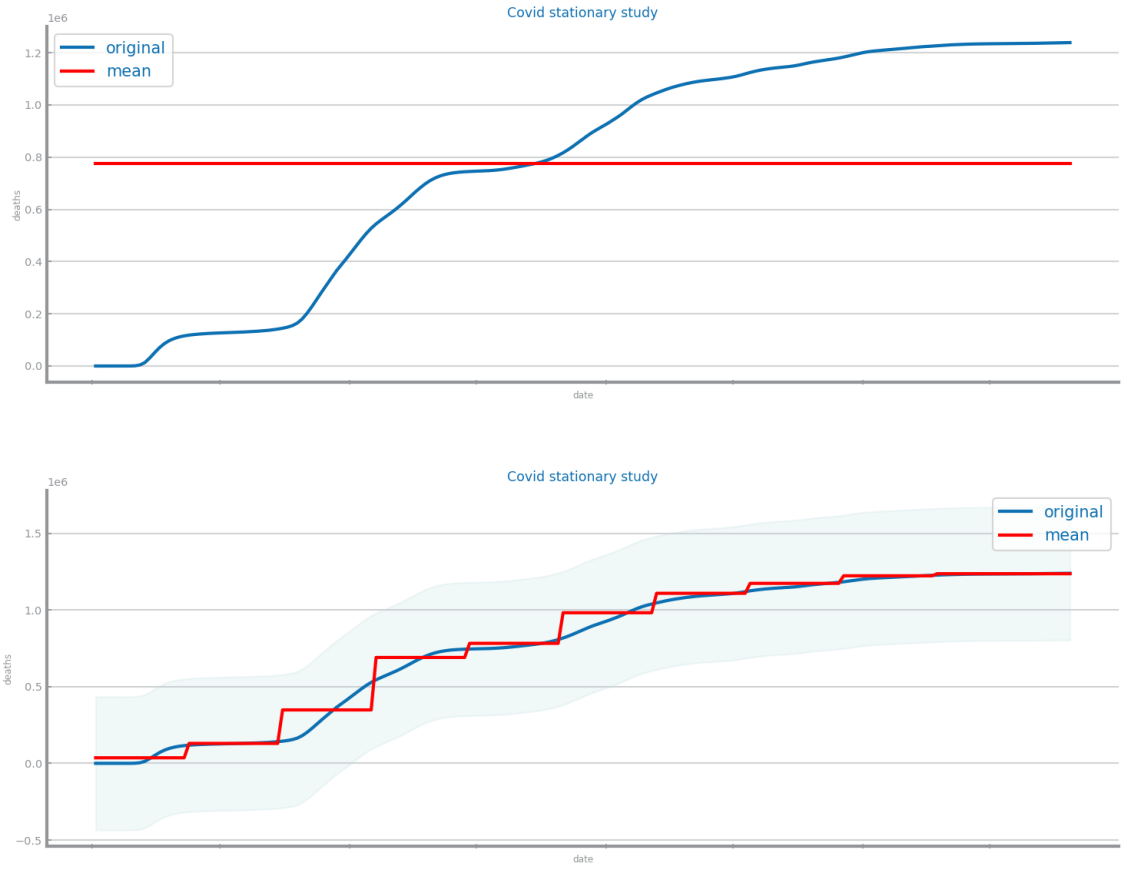


Figure 67: Stationarity study for time series 1

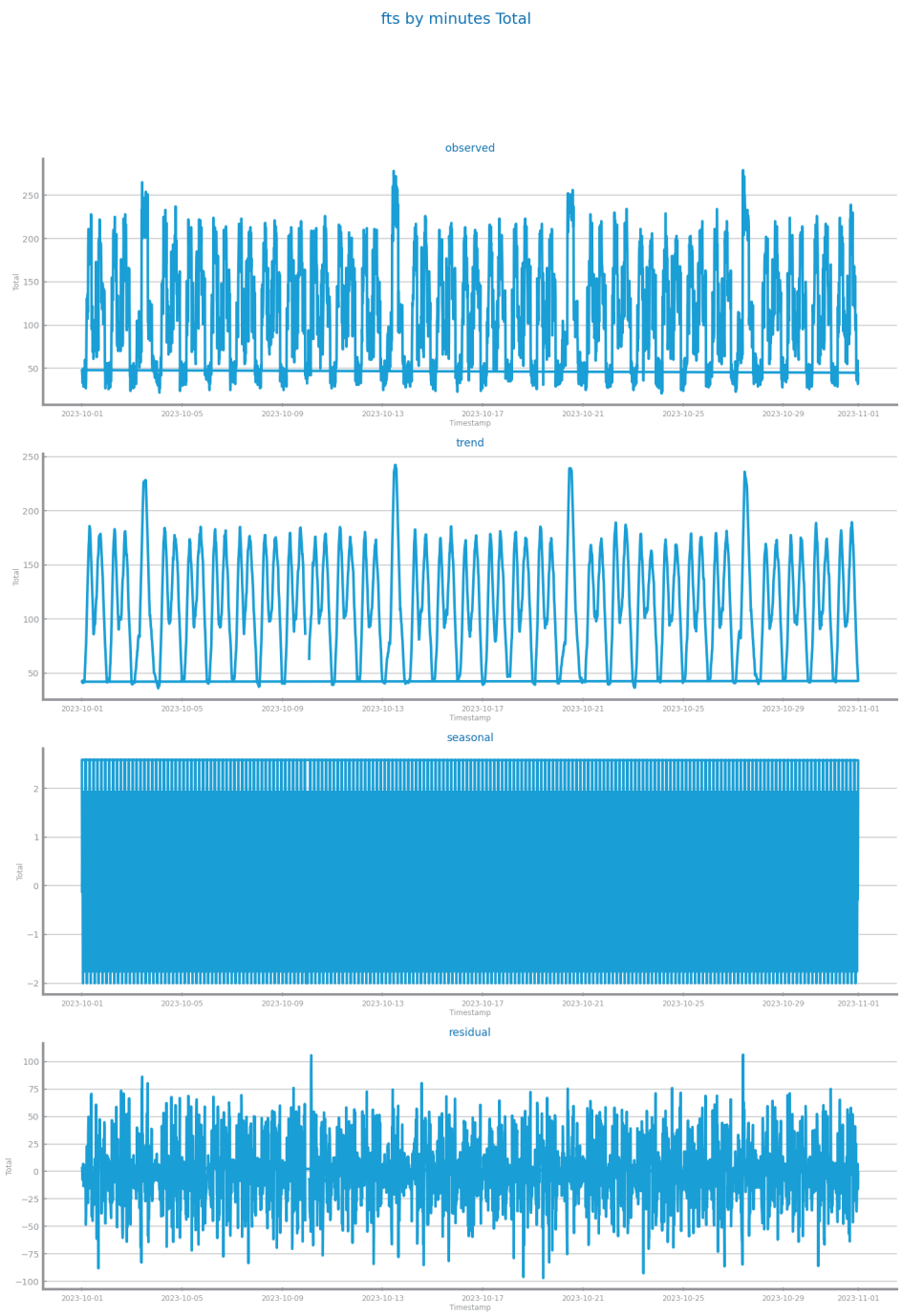


Figure 68: Components study for time series 2

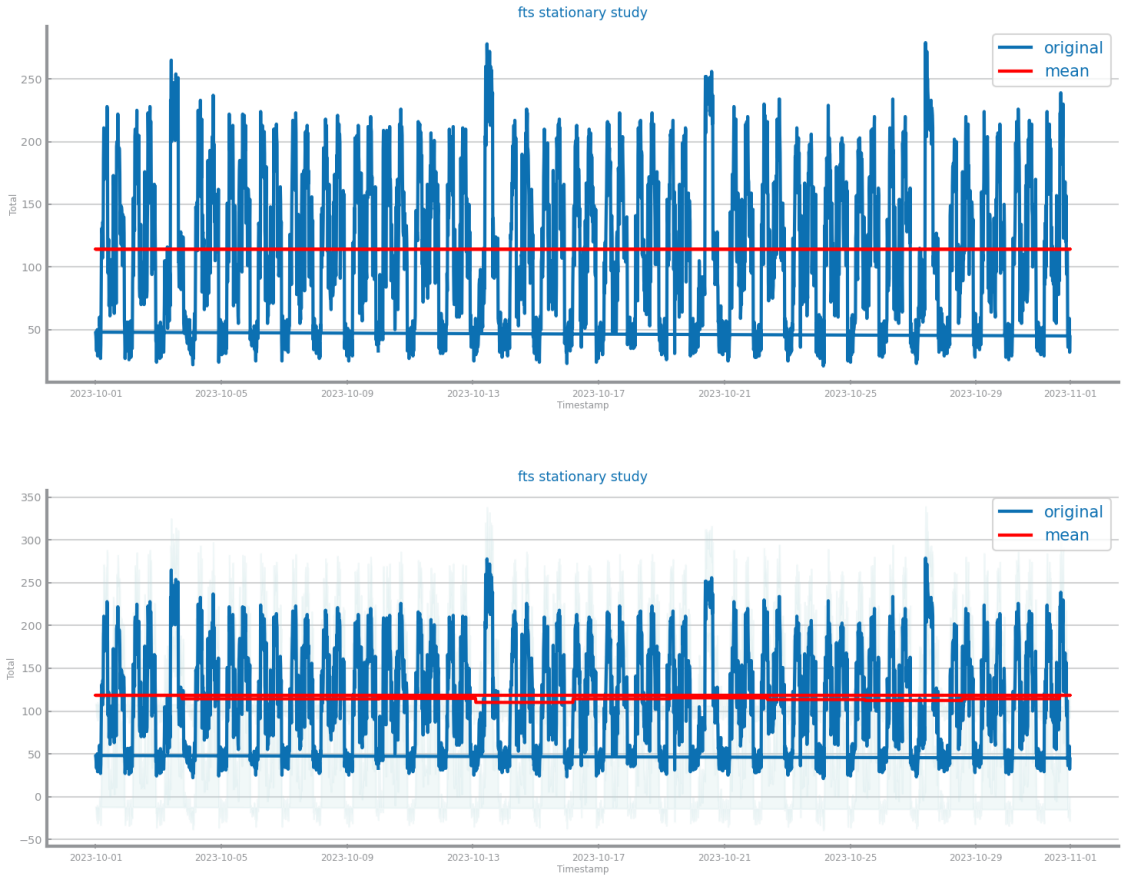


Figure 69: Stationarity study for time series 2

6 DATA TRANSFORMATION

Aggregation

To study the best aggregation, we applied the model at 3 different levels studied in Data Profiling.

In dataset 1, chose the weekly aggregation and for dataset 2, hourly aggregation as they obtained lower values for the different errors, simplifying the model without losing information or context. **Shall not exceed 300 characters.**

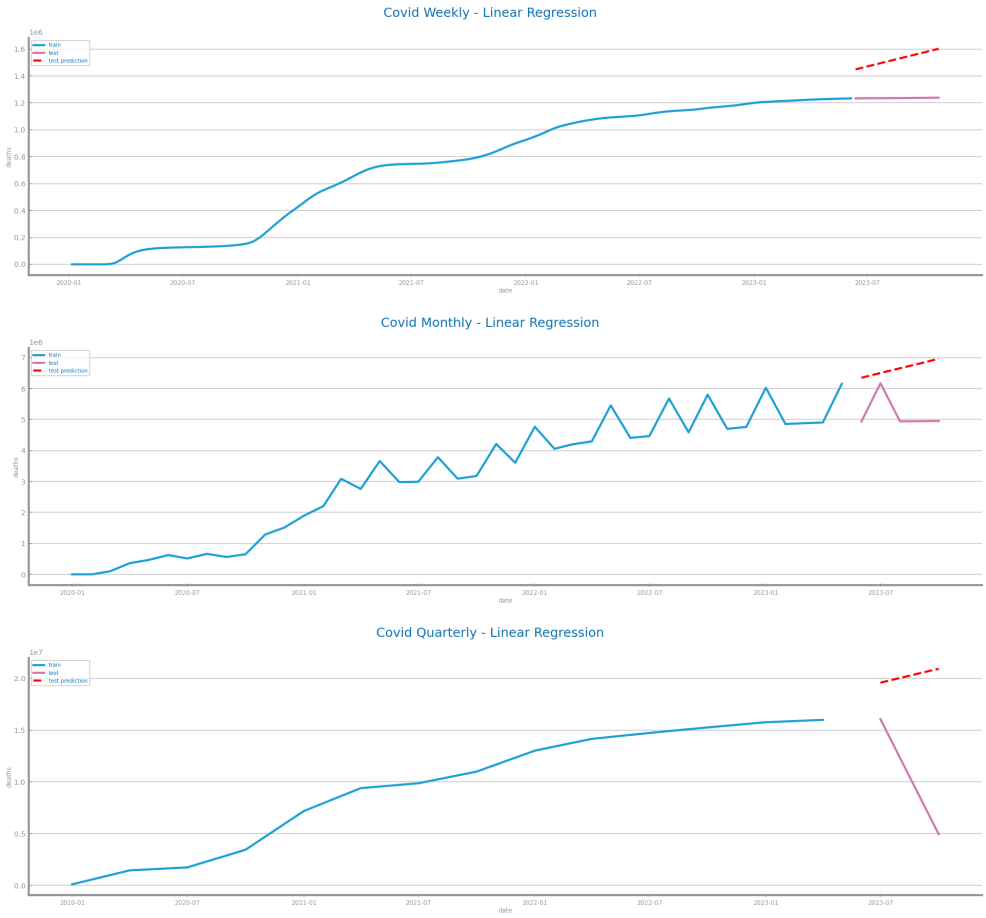


Figure 70: Forecasting plots after different aggregations on time series 1

Figure 71: Forecasting results after different aggregations on time series 1

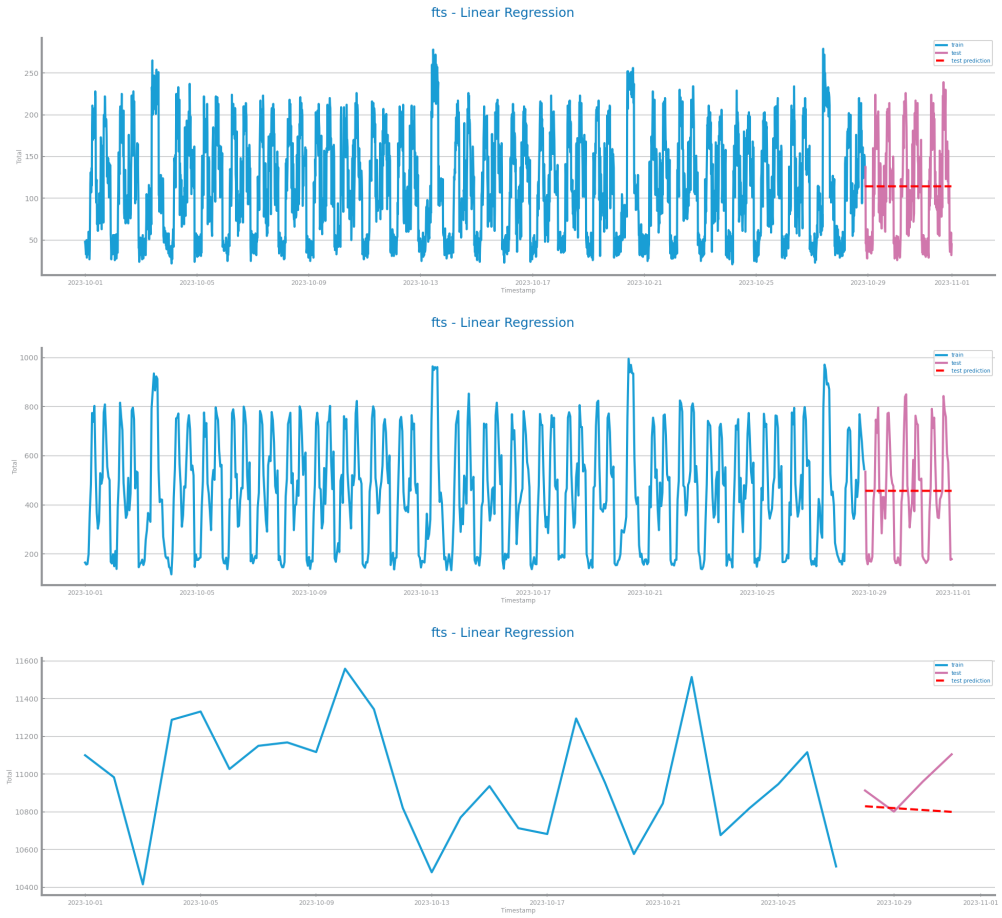


Figure 72: Forecasting plots after different aggregations on time series 2

Figure 73: Forecasting results after different aggregations on time series 2

Smoothing

To study the best Window Size, we applied the model to 4 different values (25, 50, 75 and 100).

In this case we chose 100 for both datasets since it was where we obtained the lowest values for the different errors. **Shall not exceed 300 characters.**

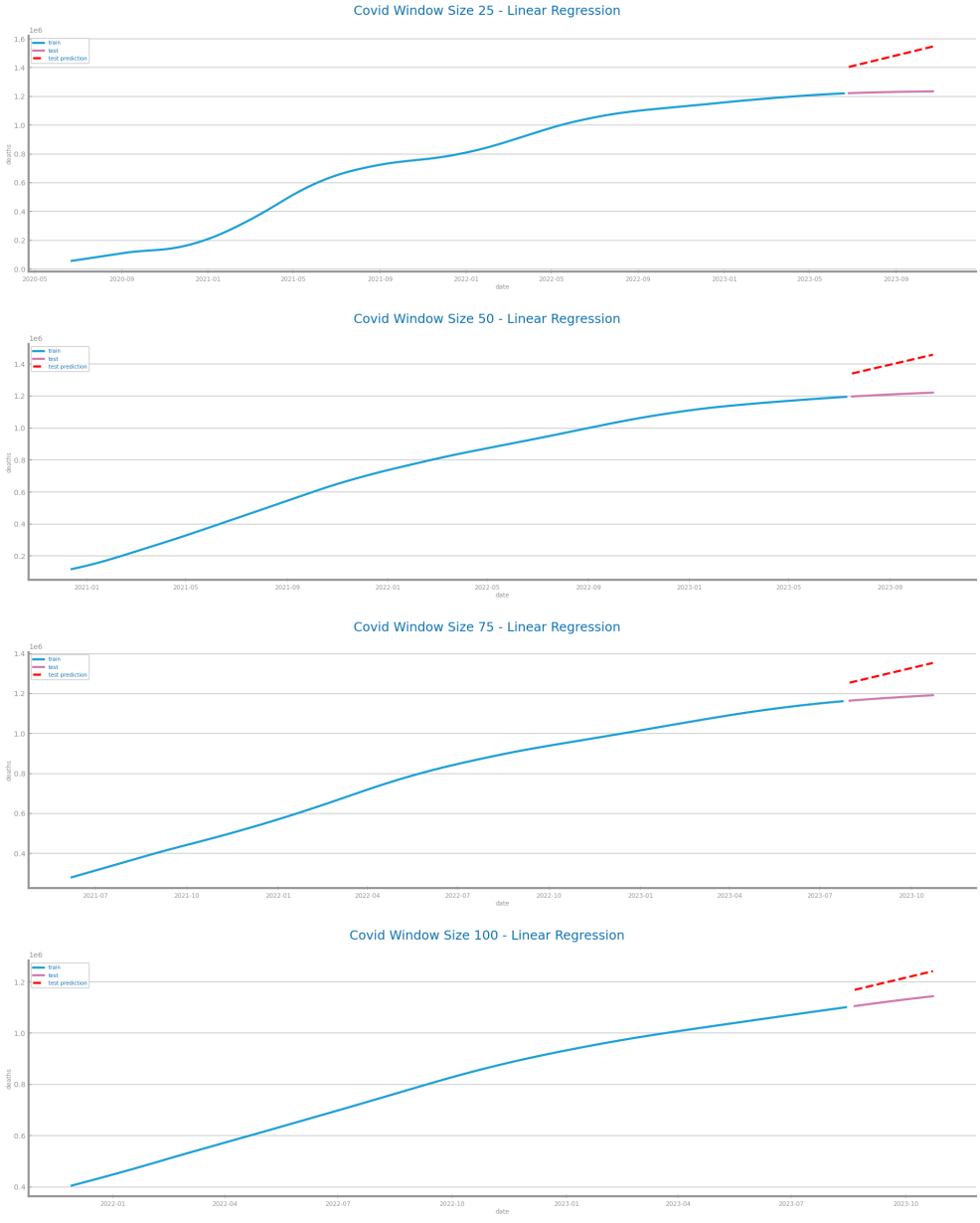


Figure 74: Forecasting plots after different smoothing parameterisations on time series 1

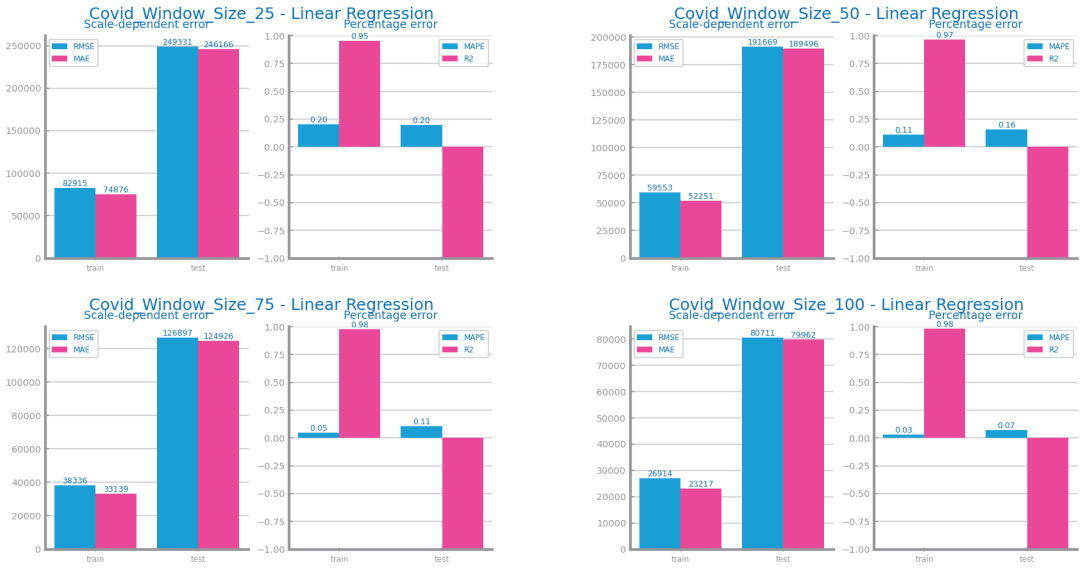


Figure 75: Forecasting results after different smoothing parameterisations on time series 1

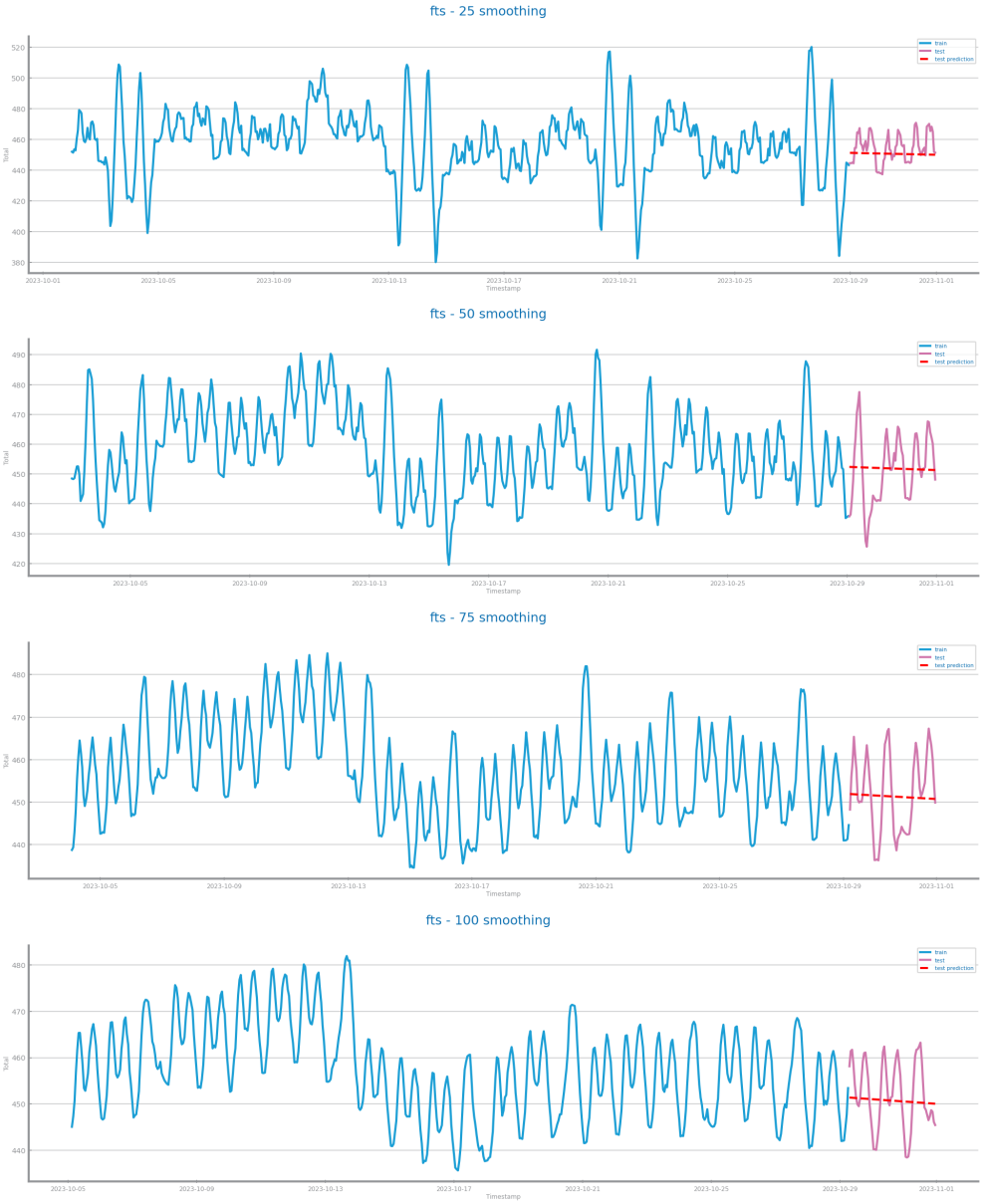


Figure 76: Forecasting plots after different smoothing parameterisations on time series 2

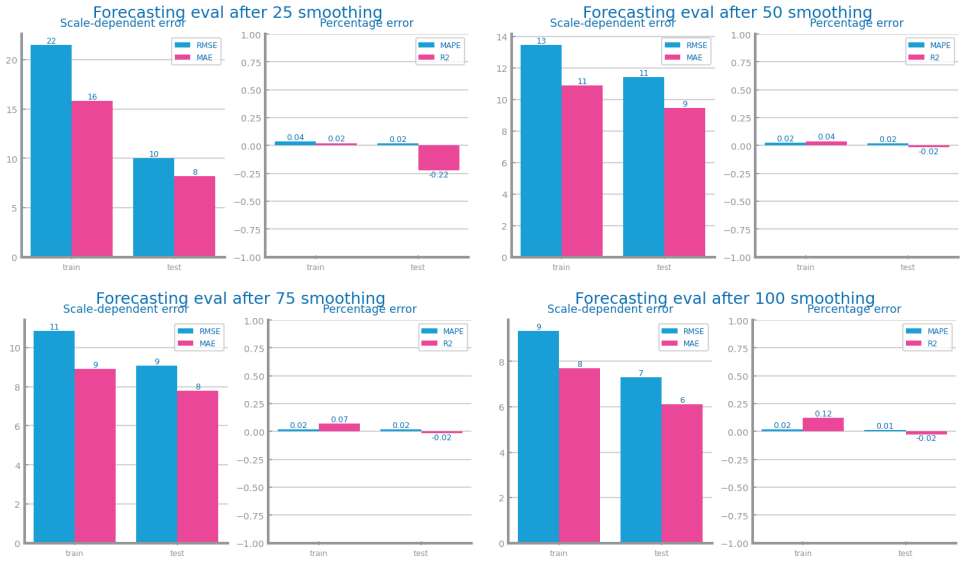


Figure 77: Forecasting results after different smoothing parameterisations on time series 2

Differentiation

To study differentiation, we applied the first two derivatives, favoring the first one for both datasets. In dataset 1, it helped remove quadratic trends and minimize the errors and for dataset 2, both derivatives removed seasonality, but the second added complexity, making it harder to predict. **Shall not exceed 300 characters.**

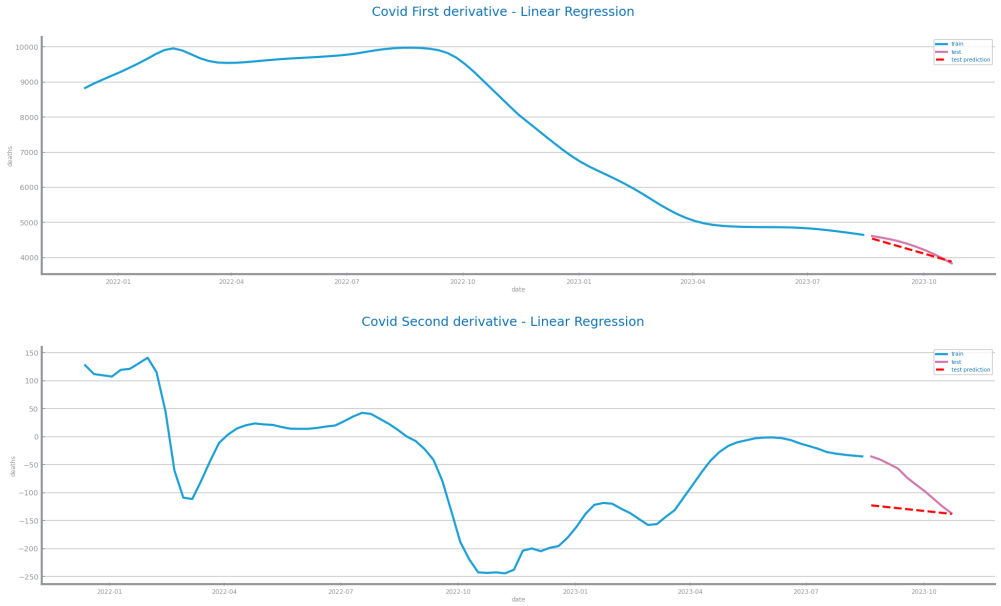


Figure 78: Forecasting plots after first and second differentiation of time series 1

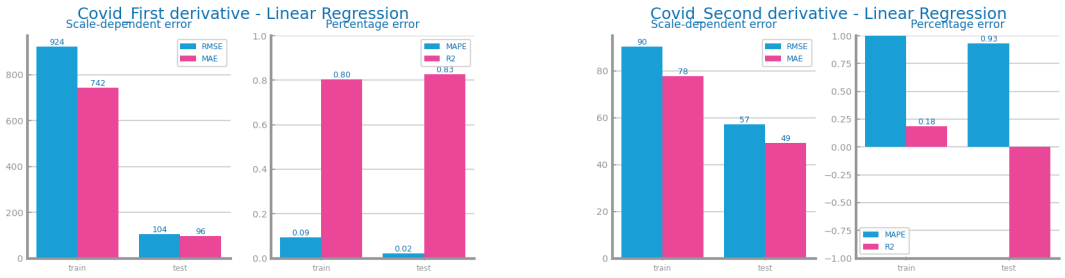


Figure 79: Forecasting results after first and second differentiation of time series 1

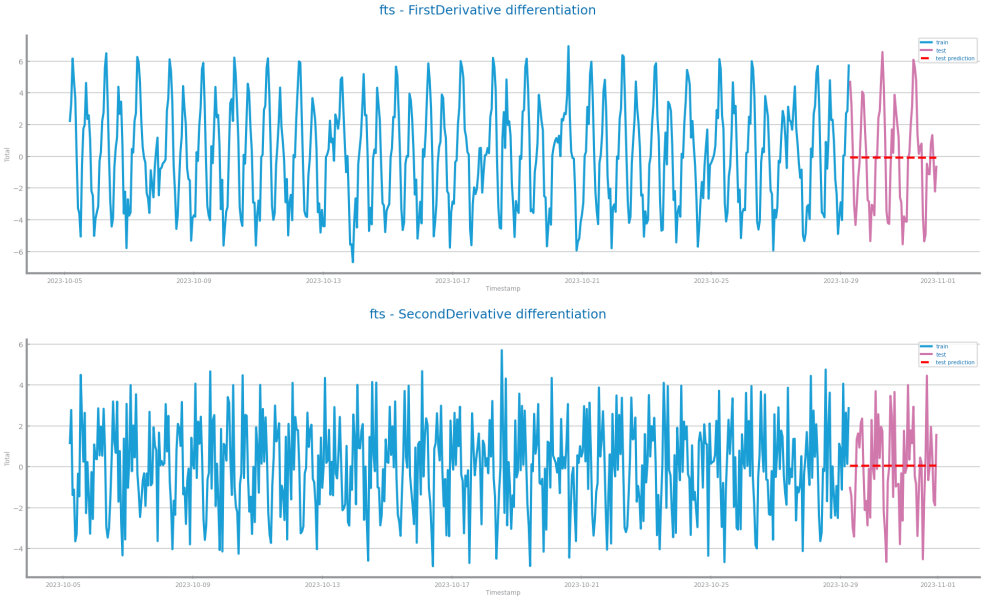


Figure 80: Forecasting plots after first and second differentiation of time series 2

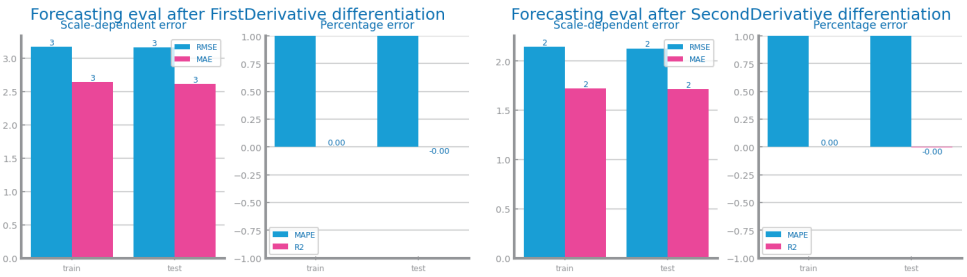


Figure 81: Forecasting results after first and second differentiation of time series 2

Other transformations (optional)

Finally, we applied scaling in both datasets in order to have best values to use in the models' evaluation specifically in the LSTM model. **Shall not exceed 500 characters.**

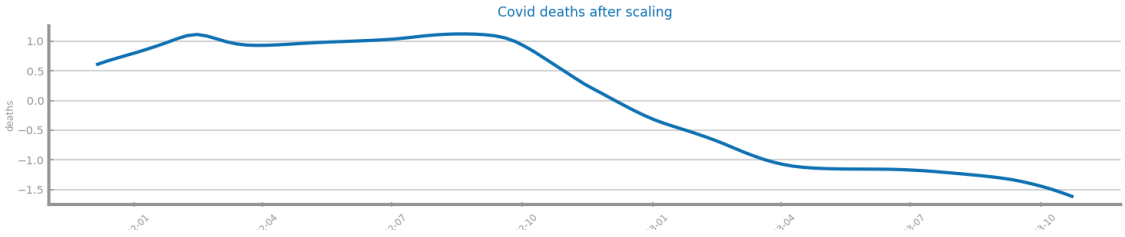


Figure 82: Forecasting plots after applying scaling over time series 1

Figure 83: Forecasting results after applying other transformations over time series 1

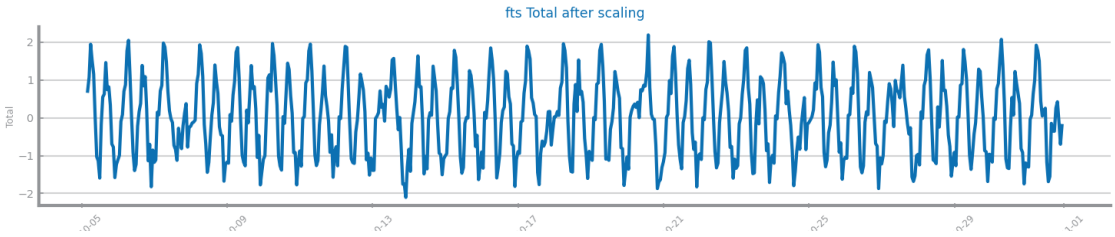


Figure 84: Forecasting plots after applying other transformations over time series 2

Figure 85: FForecasting results after applying other transformations over time series 2

7 MODELS' EVALUATION

For dataset 1 we used the weekly aggregation with window size=100, the first derivative and then applied scaling.

For dataset 2 we selected minutely aggregation, window size=100, first derivative and scaling. Results were surprisingly positive for dataset 1 when applying the linear regression compared to dataset 2 as the last is closer to the shape of a cosine function instead of linear. For the aggregation study, higher levels were not selected due to high loss of information.

Simple Average Model

Although this metric doesn't approximate any of the datasets correctly, the error and R2 values for dataset 2 seem better because there are some contact points between the real and predicted values.

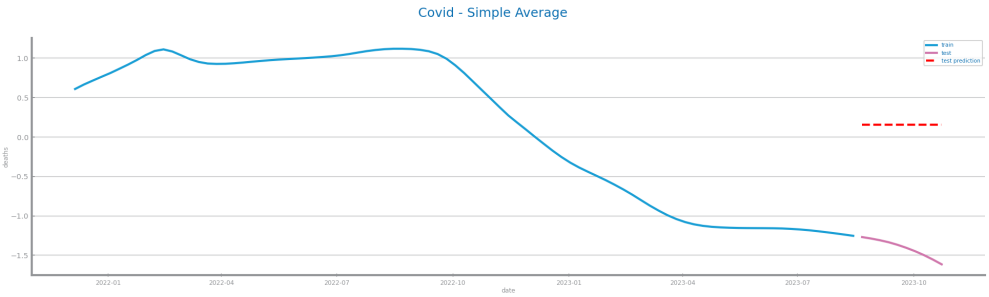


Figure 86: Forecasting plots obtained with Simple Average model over time series 1

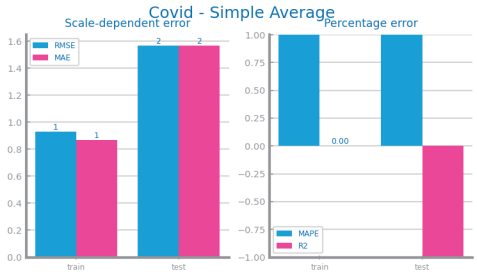


Figure 87: Forecasting results obtained with Simple Average model over time series 1

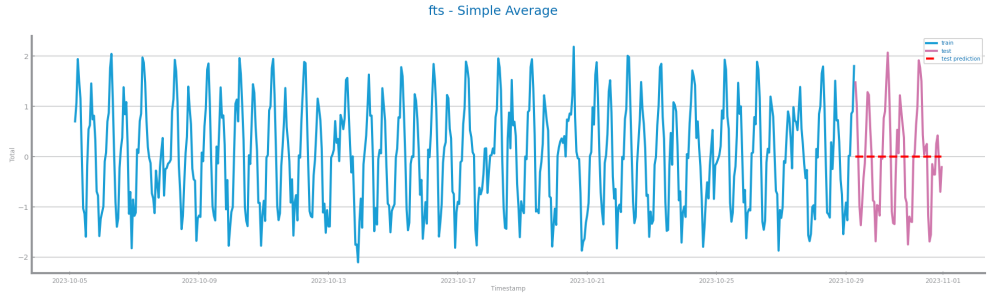


Figure 88: Forecasting plots obtained with Simple Average model over time series 2

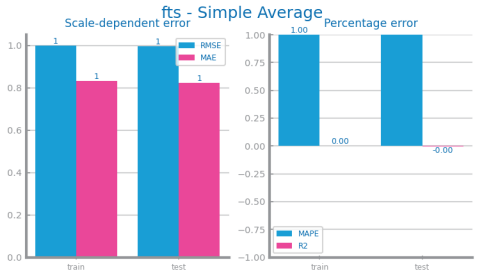


Figure 89: Forecasting results obtained with Simple Average model over time series 2

Persistence Model

The persistence model analysis displays bad results for the realist model and very good results for the optimistic. However, the optimistic model isn't capable to make long-term predictions, it can only accurately predict on a short term space whereas the realist model approximates for long distance. For these reasons, both are bad models for the datasets.

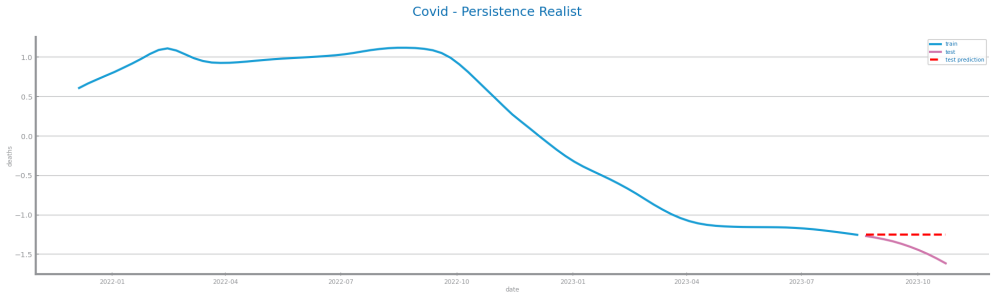


Figure 90: Forecasting plots obtained with Persistence model (long term) over time series 1

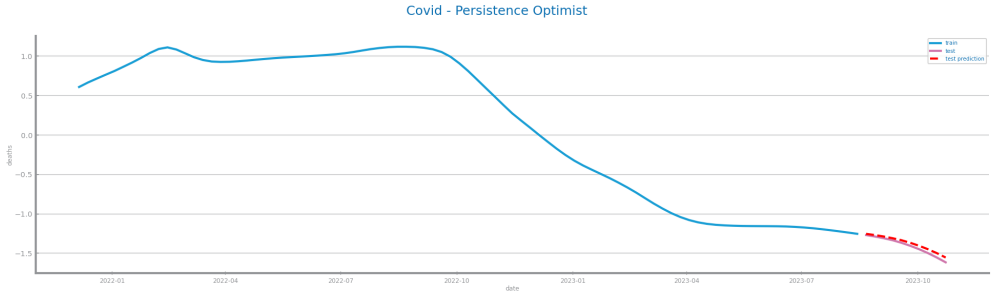


Figure 91: Forecasting plots obtained with Persistence model (one-set-behind) over time series 1

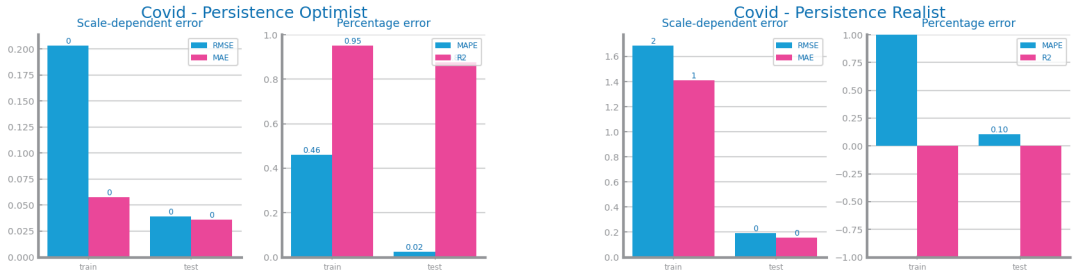


Figure 92: Forecasting results obtained with Persistence model in both situations over time series 1

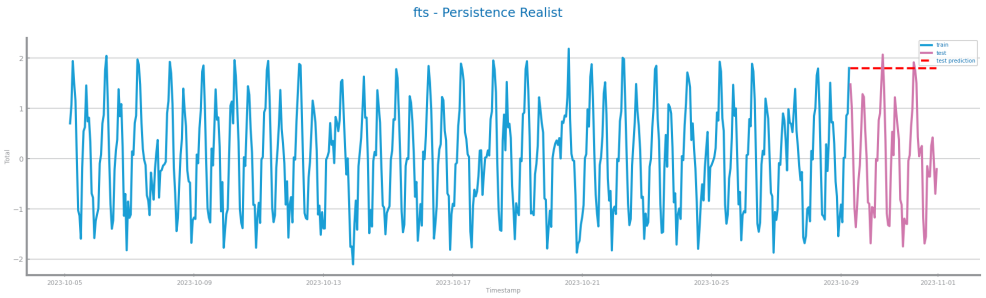


Figure 93: Forecasting plots obtained with Persistence model (long term) over time series 2

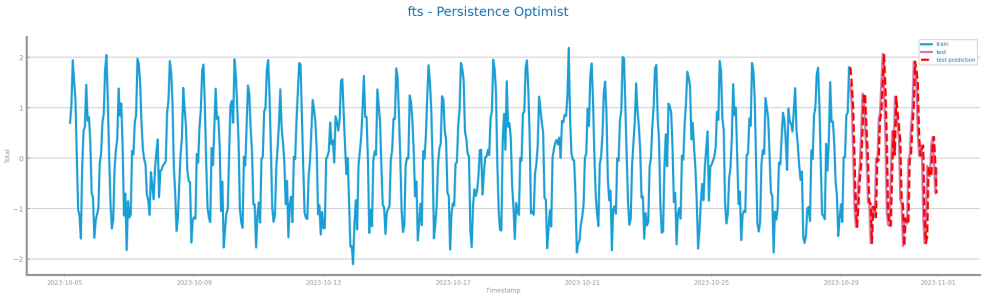


Figure 94: Forecasting plots obtained with Persistence model (one-set-behind) over time series 2

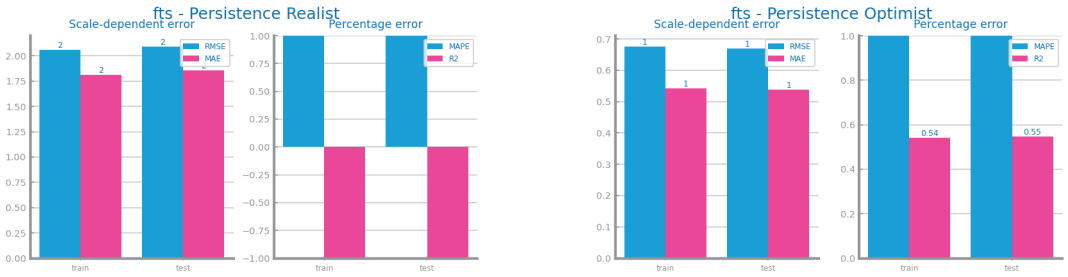


Figure 95: Forecasting results obtained with Persistence model in both situations over time series 2

Rolling Mean Model

Although this metric doesn't approximate any of the datasets correctly, the first dataset obtains better results for the MAE, RMSE and MAPE as it predicts values closer to the real ones but doesn't predict any correct value while the second dataset has a better R2 because it fluctuates between somewhat symmetric high and low values obtaining a horizontal line between them so there are some contact points but the line remains very distant from the minimum and maximum points.

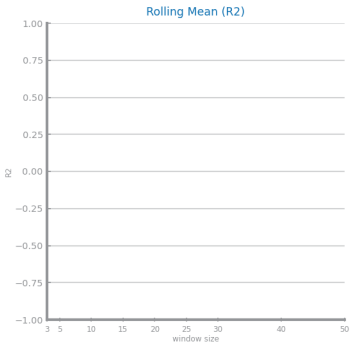


Figure 96: Forecasting study over different parameterisations of the rolling mean algorithm over time series 1

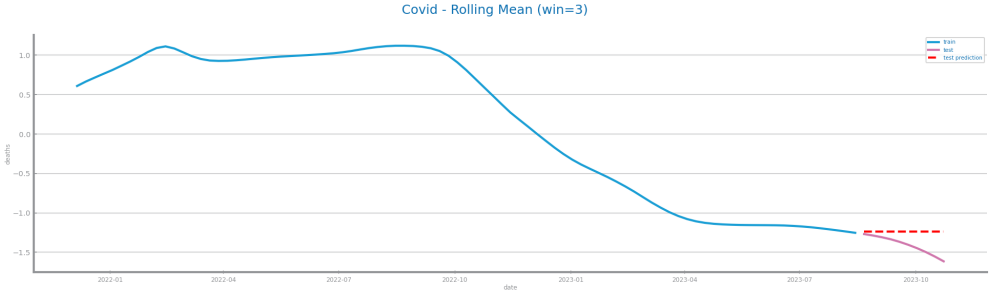


Figure 97: Forecasting plots obtained with the best parameterisation of rolling mean algorithm, over time series 1

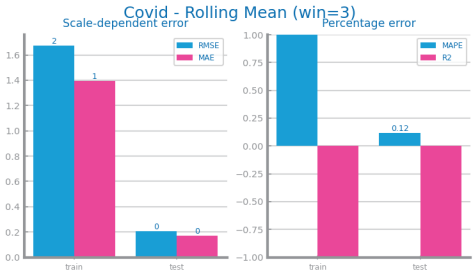


Figure 98: Forecasting results obtained with the best parameterisation of rolling mean algorithm, over time series 1

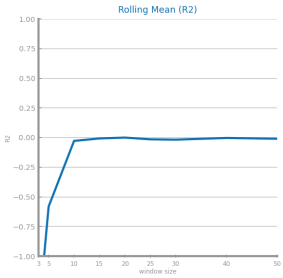


Figure 99: Forecasting study over different parameterisations of the rolling mean algorithm over time series 2

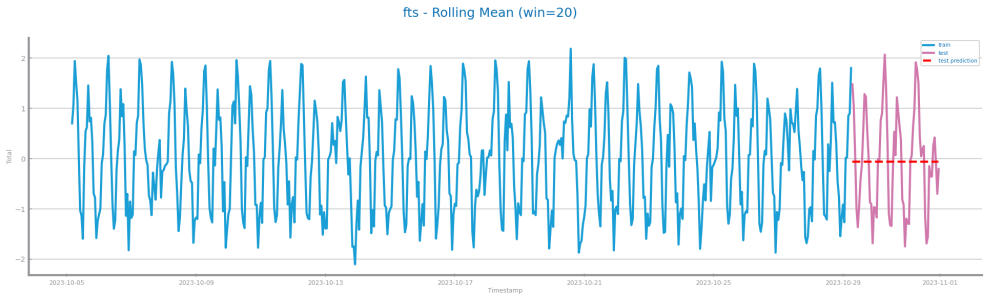


Figure 100: Forecasting plots obtained with the best parameterisation of rolling mean algorithm, over time series 2

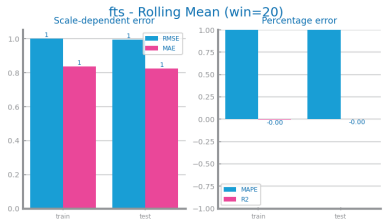


Figure 101: Forecasting results obtained with the best parameterisation of rolling mean algorithm, over time series 2

ARIMA Model

Dataset 1, with a trend-style pattern, benefits from parameters (p, d, q) set to (7, 2, 5). This configuration allows the model to capture and accommodate the complexities associated with trend-based data.

On the other hand, Dataset 2, exhibiting a cosine-like shape, attains superior performance with parameters at (3, 0, 5). This parameter choice enables the model to capture the cyclical and periodic components in the dataset, showcasing the adaptability of ARIMA to diverse time series patterns.

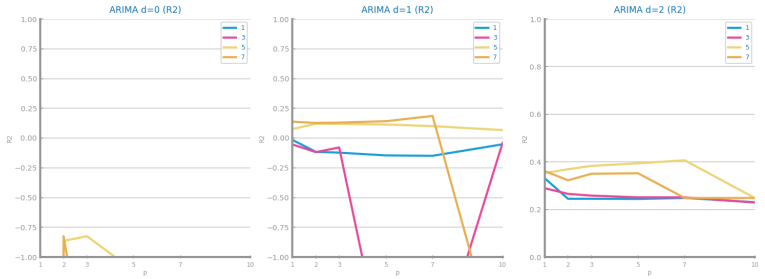


Figure 102: Forecasting study over different parameterisations of the ARIMA algorithm over time series 1

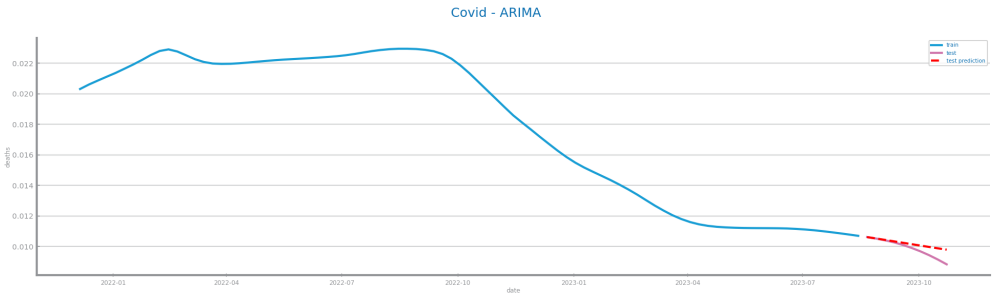


Figure 103: Forecasting plots obtained with the best parameterisation of ARIMA algorithm, over time series 1

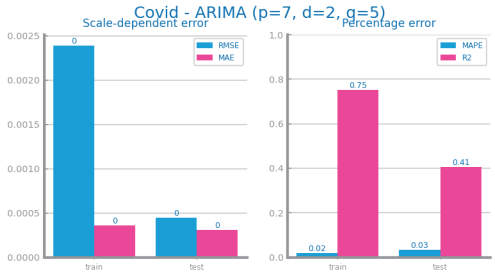


Figure 104: Forecasting results obtained with the best parameterisation of ARIMA algorithm, over time series 1

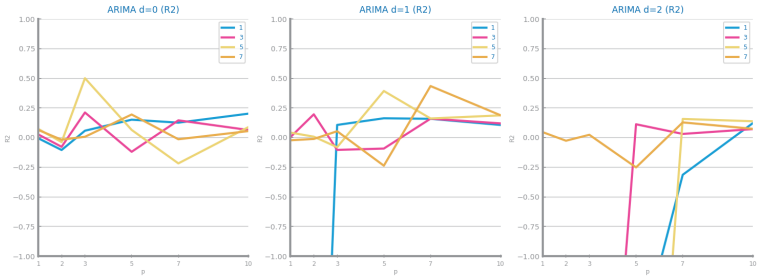


Figure 105: Forecasting study over different parameterisations of the ARIMA algorithm over time series 2

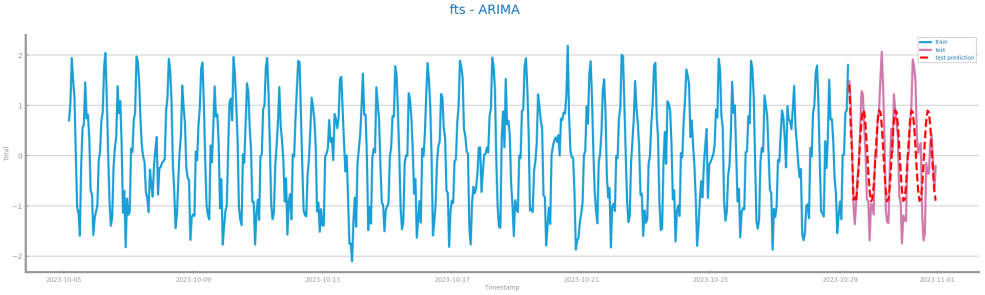


Figure 106: Forecasting plots obtained with the best parameterisation of ARIMA algorithm, over time series 2

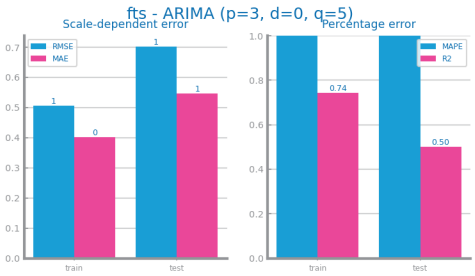


Figure 107: Forecasting results obtained with the best parameterisation of ARIMA algorithm, over time series 2

LSTMs Model

For Dataset 1, having a trend, the LSTM excels with parameters length=4, hidden=100 and nr_episodes=900, effectively capturing trend-oriented patterns. This adaptability extends to Dataset 2, featuring a cosine-like trend, where the same parameter configuration yields optimal results. As expected the LSTM's model achieves the best forecasting results for both datasets compared to the previous models.

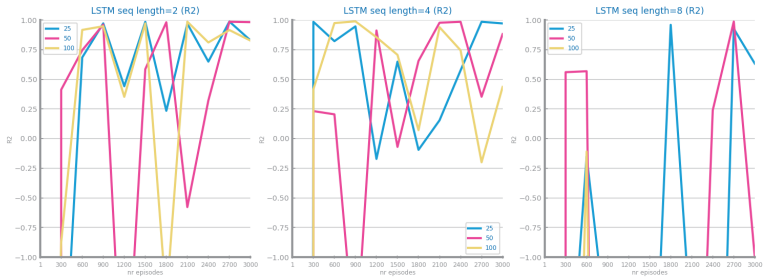


Figure 108: Forecasting study over different parameterisations of LSTMs over time series 1

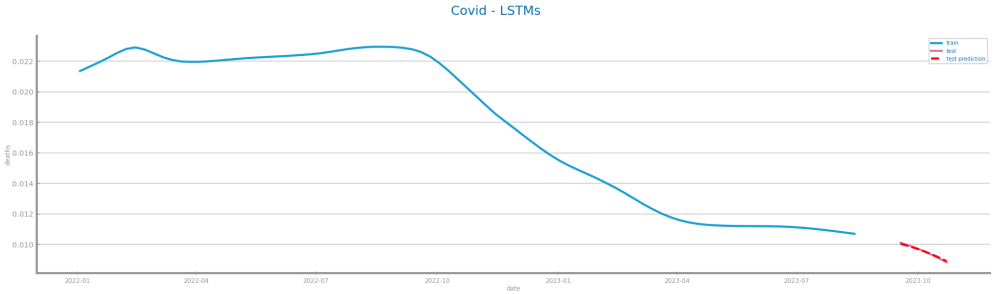


Figure 109: Forecasting plots obtained with the best parameterisation of LSTMs, over time series 1

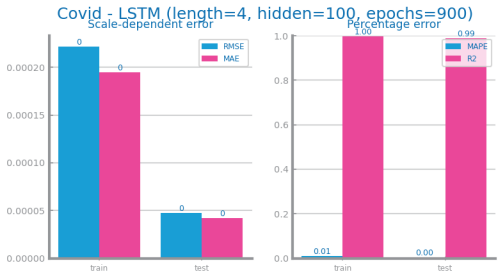


Figure 110: Forecasting results obtained with the best parameterisation of LSTMs, over time series 1

Figure 111: Forecasting study over different parameterisations of LSTMs with multiple variables over time series 1

Figure 112: Forecasting plots obtained with the best parameterisation of LSTMs with multiple variables over time series 1

Figure 113: Forecasting results obtained with the best parameterisation of LSTMs with multiple variables over time series 1

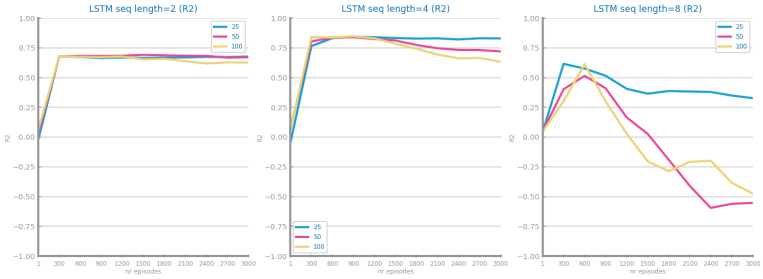


Figure 114: Forecasting study over different parameterisations of the LSTMs over time series 2

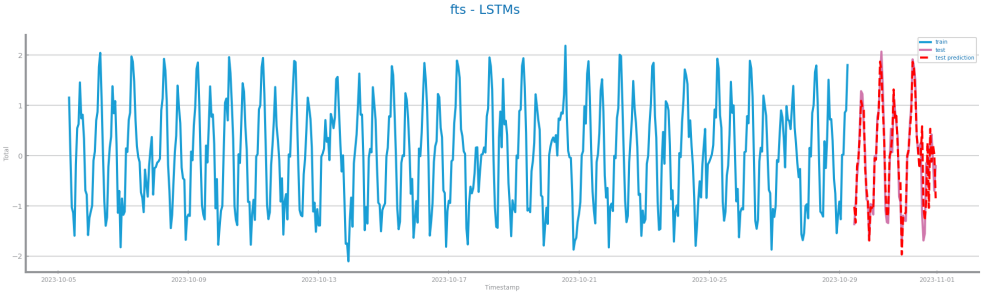


Figure 115: Forecasting plots obtained with the best parameterisation of LSTMs, over time series 2

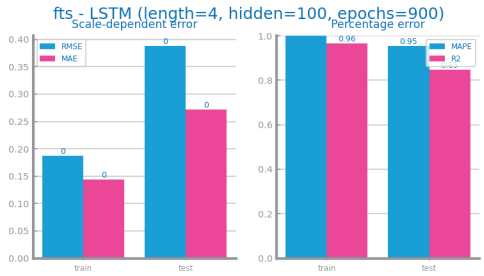


Figure 116: Forecasting results obtained with the best parameterisation of LSTMs, over time series 2

Figure 117: Forecasting study over different parameterisations of LSTMs with multiple variables over time series 2

Figure 118: Forecasting plots obtained with the best parameterisation of LSTMs with multiple variables over time series 2

Figure 119: Forecasting results obtained with the best parameterisation of LSTMs with multiple variables over time series 2

8 CRITICAL ANALYSIS

In a critical assessment, both ARIMA and LSTMs outperformed simpler models like simple average or rolling mean, emphasizing their adaptability and effectiveness. LSTMs, in particular, demonstrated superior performance across diverse datasets, positioning them as a favorable choice for both. The decision between these strands hinges on dataset specifics and the desired trade-off between interpretability and complexity. Dataset 1 reveals a non-stationary nature, characterized by an initial upward trend with it slowing down. Although on the smoothing phase this shape is flattened, this decline in Covid related death rates can be seen on the differentiation analysis, by employing the second derivative and removing the quadratic trend, recalling the beginning upwards activity and settling down later. In order to have best suited values for the LSTM model, we applied scaling. Notably, although we predicted it to be a good fit for this model, ARIMA is not a suitable fit for this series as it's prediction is too long-term based, not dealing with the abrupt decline. For dataset 2, also non-stationary, having applied both smoothing, differentiation and scaling to regularise the value spikes and revealing cyclical and seasonal behaviors of this series, but with no evident trend. As expected but differing from the first dataset, the ARIMA model provides a good fit, encapsulating the cyclicality of this series on a good level. Despite not being as accurate as LSTM, if the model complexity from the latest proves a bottleneck for the problem, ARIMA is a good substitute. The univariate nature of the time series hinders accurate prediction, leading to suboptimal model performance. For this reason and as the simple average model, persistence realist and optimist and rolling mean can only deal with linear series or short term predictions which are not the case for either datasets, none of these models seem good enough to solve these problems. **Shall not exceed 2000 characters.**