

# Data Science Project

<b>Team nr:</b> 20	<b>Student 1 :</b> André Alves	<b>IST nr:</b> 110857
	<b>Student 2 :</b> Francisco Abreu	<b>IST nr:</b> 110946
	<b>Student 3 :</b> João Ferreira	<b>IST nr:</b> 110954
	<b>Student 4 :</b> Rúben Nobre	<b>IST nr:</b> 99321

## CLASSIFICATION

### 1 DATA PROFILING

#### DS1

Imbalanced target: No 70% vs Yes 30%

BMI = weight (kg) / [height (m)]<sup>2</sup>

#### DS2

Remove & check input errors like

Narrow age from 0 to 120 in *Transformation*

Unusual values *Interest Rate* above 150%

#### *Data Dimensionality*

No dimensionality curse in both datasets.

#### DS1

Most variables are binary

No date variables

Max % of missing values is 9.1, not that relevant in the context of this Dataset

## DS2

Most variables are **numeric**

10 out of 27 are **symbolic**, can cause difficulties when trying to convert them to ordinal later

*CreditMix* has 20% of MV which must me dealt with later on in the Preparation Step

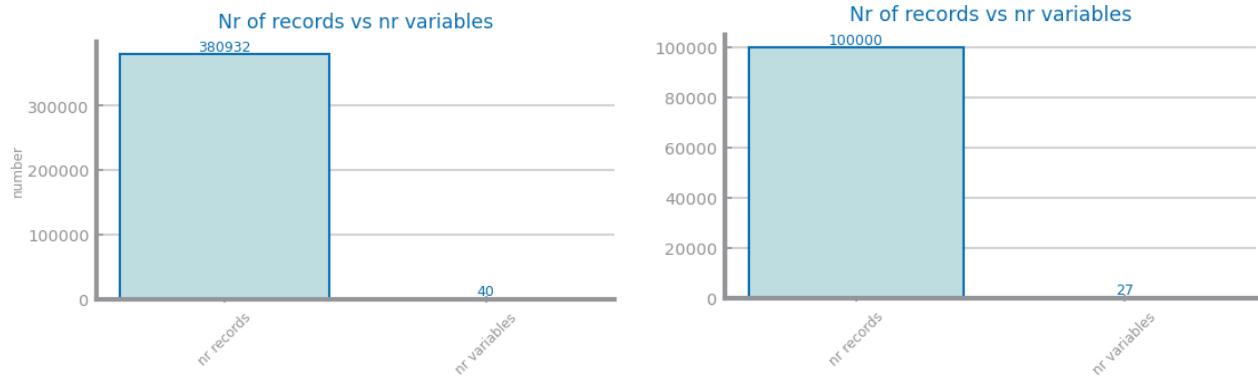


Figure 1 Nr Records x Nr variables for dataset 1 (left) and dataset 2 (right)

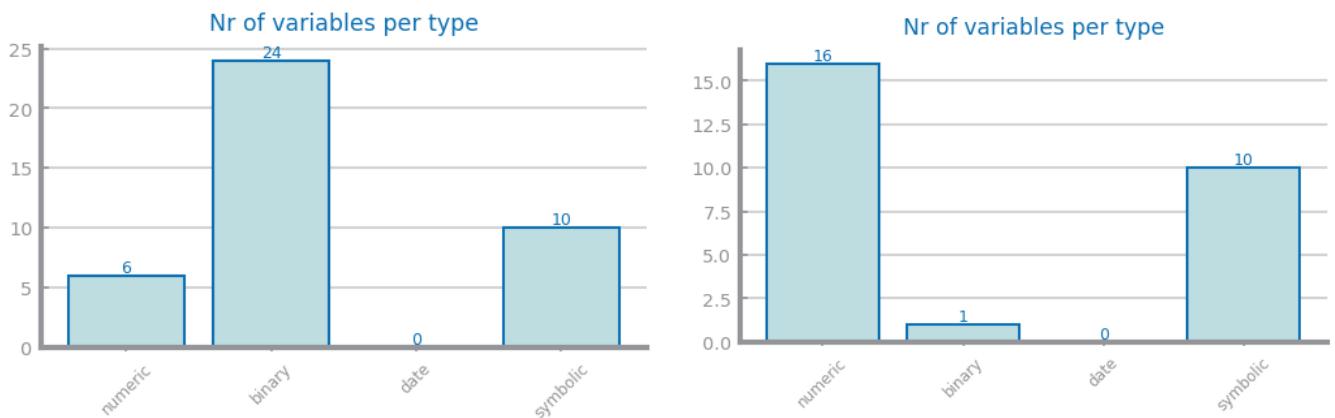


Figure 2 Nr variables per type for dataset 1 (left) and dataset 2 (right)

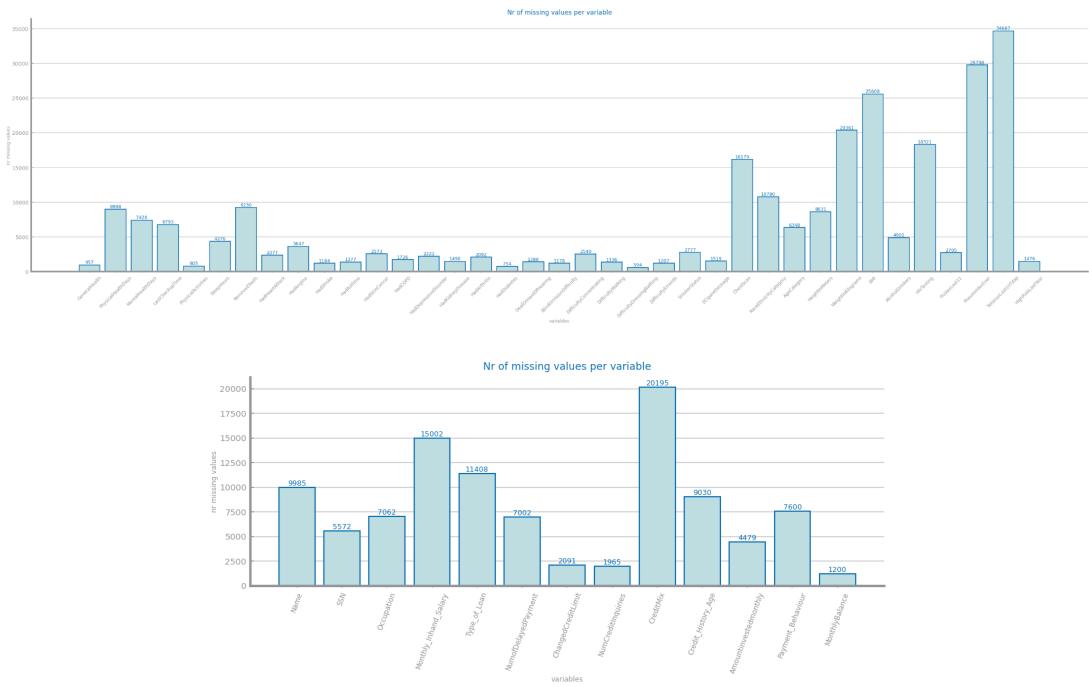


Figure 3 Nr missing values for dataset 1 (top) and dataset 2 (bottom)

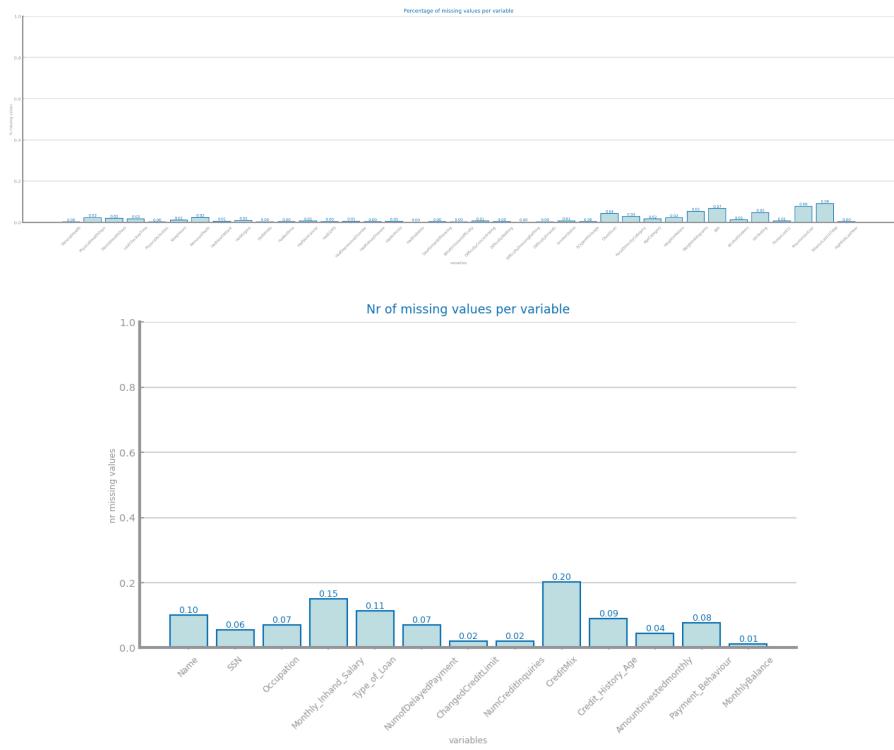


Figure 3.1 Nr missing values for dataset 1 (top) and dataset 2 (bottom) percentages (%)

## Data Distribution

### DS1

No clearly visible outliers on boxplots

Many 0s on physical and mental health days, possible MV. Other numeric vars show near-normal distribution

Unbalanced target, majority=No (70/30)

### DS2

In Fig4&6, outliers skew scales

Remove not meaningful ID's - ID , CustomerID, SSN & Name

Remove Extreme Outlier in *MonthlyBalance* 9Recs = -3.3333E+26 *not in the context of the problem*

#### Neg Input Errors

*Credit\_History\_Age* treat as **numeric**

**Unbalanced Target** 0.41:1, majority is Good Credit Score

Most **Log-n** distributions

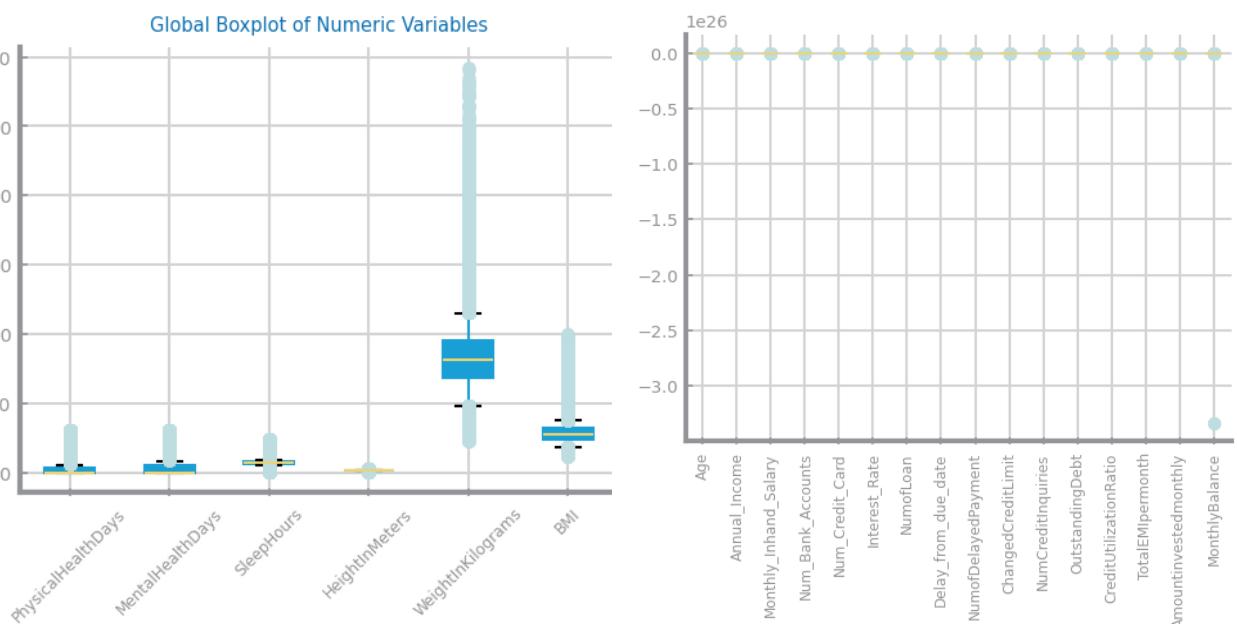


Figure 4 Global boxplots dataset 1 (left) and dataset 2 (right)

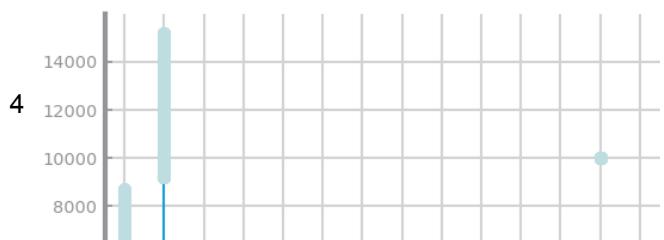


Figure 4.1 Global boxplots dataset 2 (without specific vars)

Single Numeric Variables Boxplots

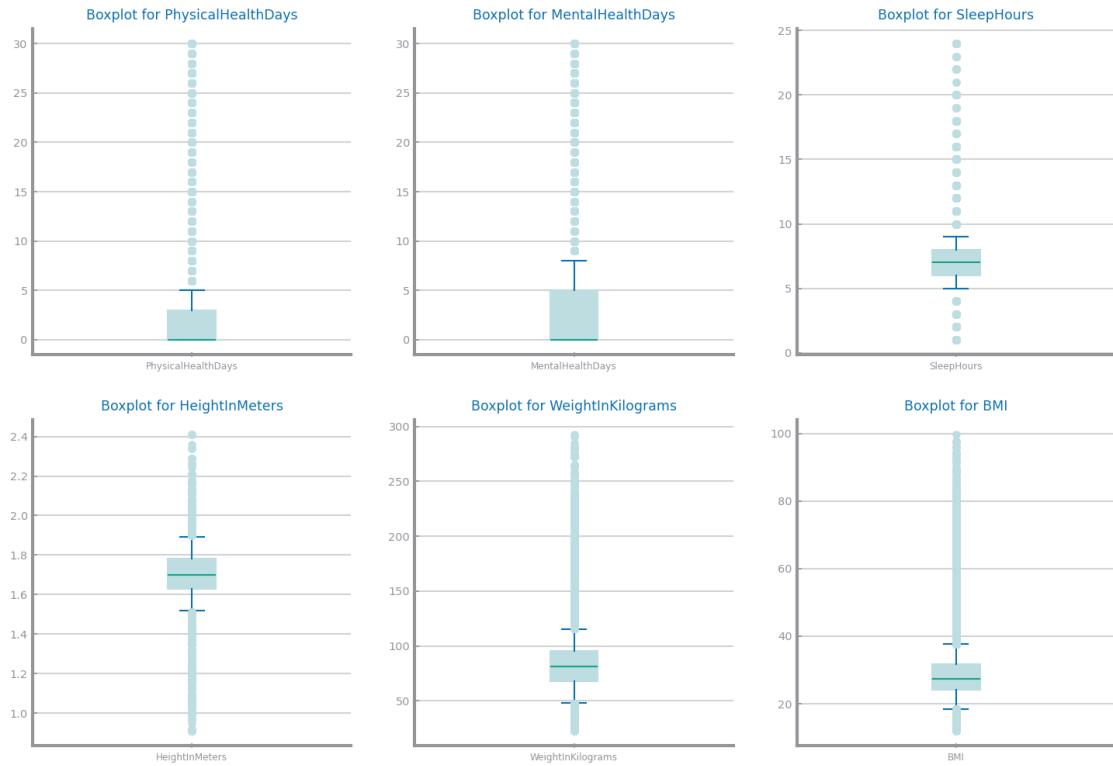


Figure 5 Single variable boxplots for dataset 1

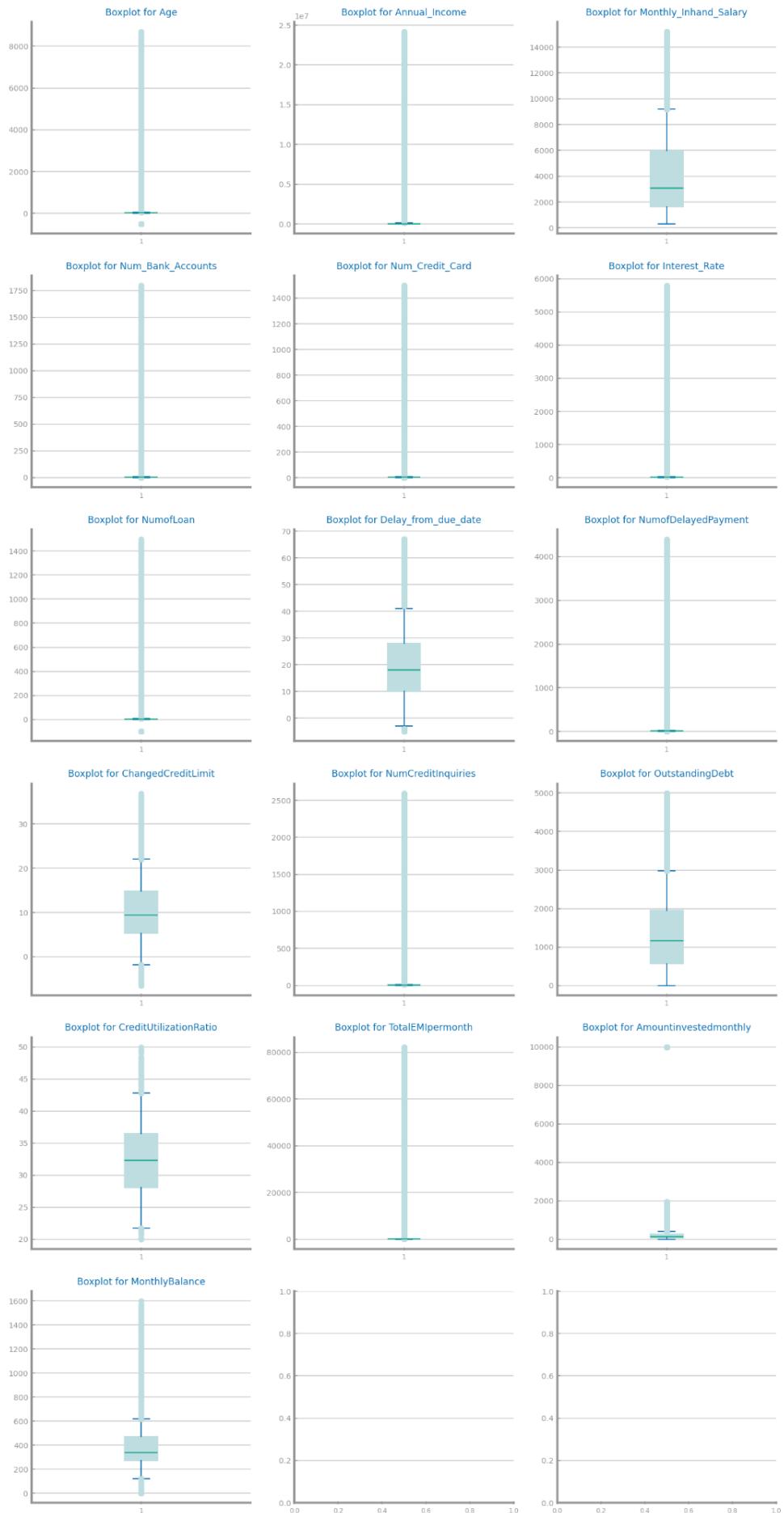


Figure 6 Single variable boxplots for dataset 2

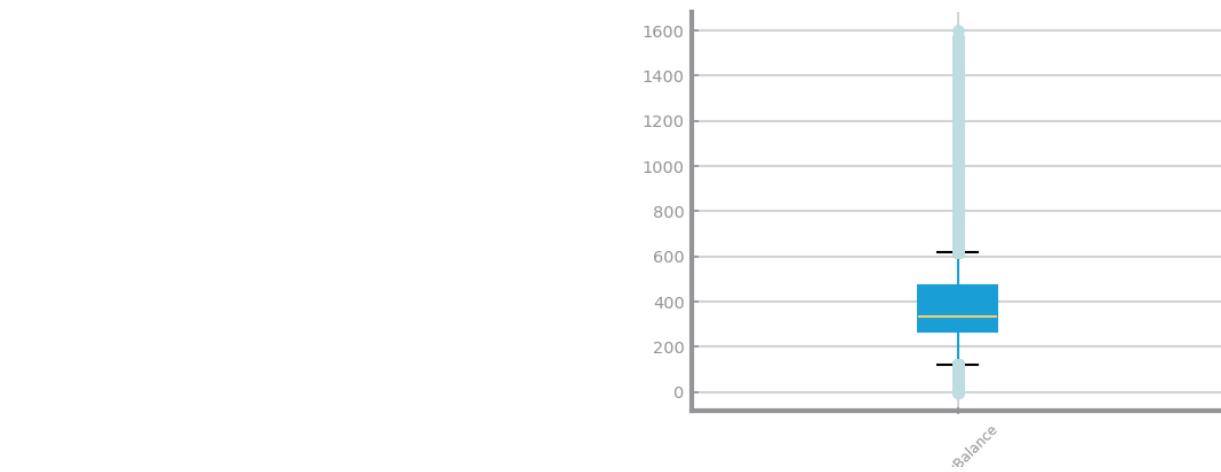


Figure 6.1 Single variable boxplot for Monthly\_Balance dataset 2

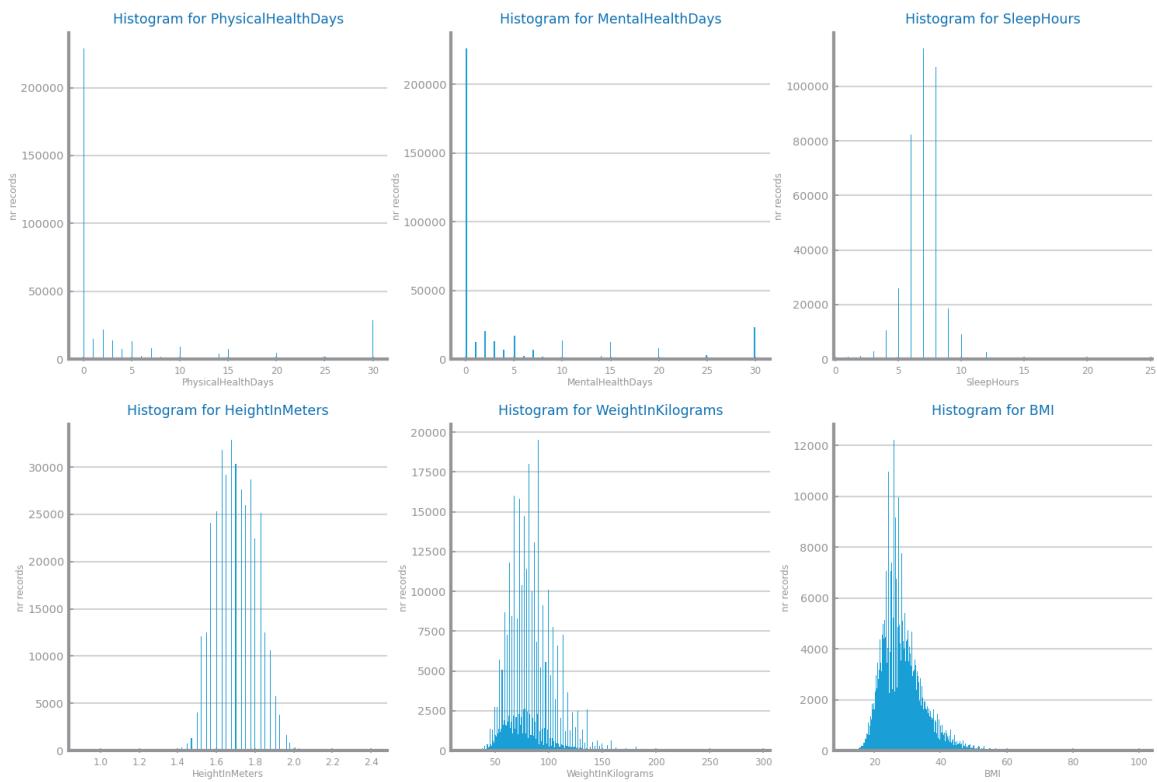


Figure 7 Numeric Histograms for dataset 1

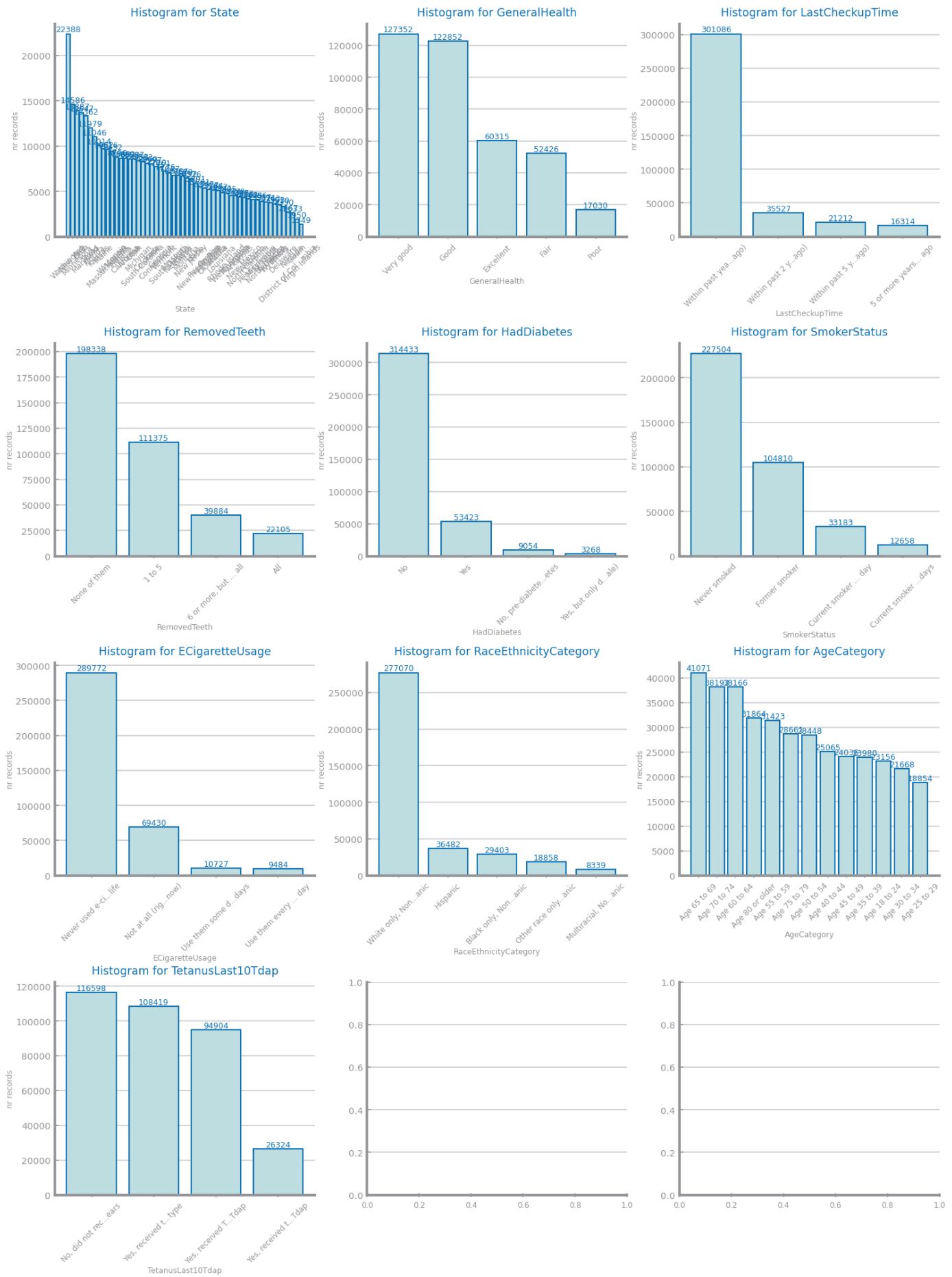


Figure 7.1 Symbolic Histograms for dataset 1

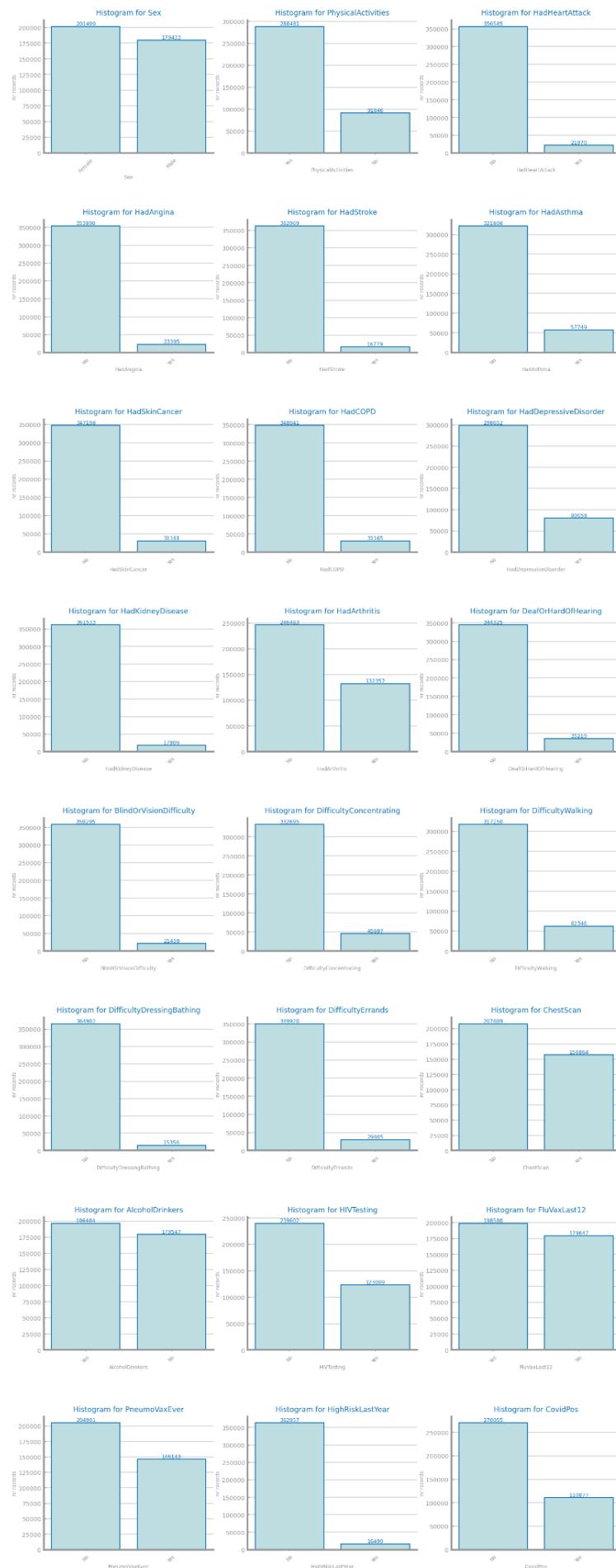
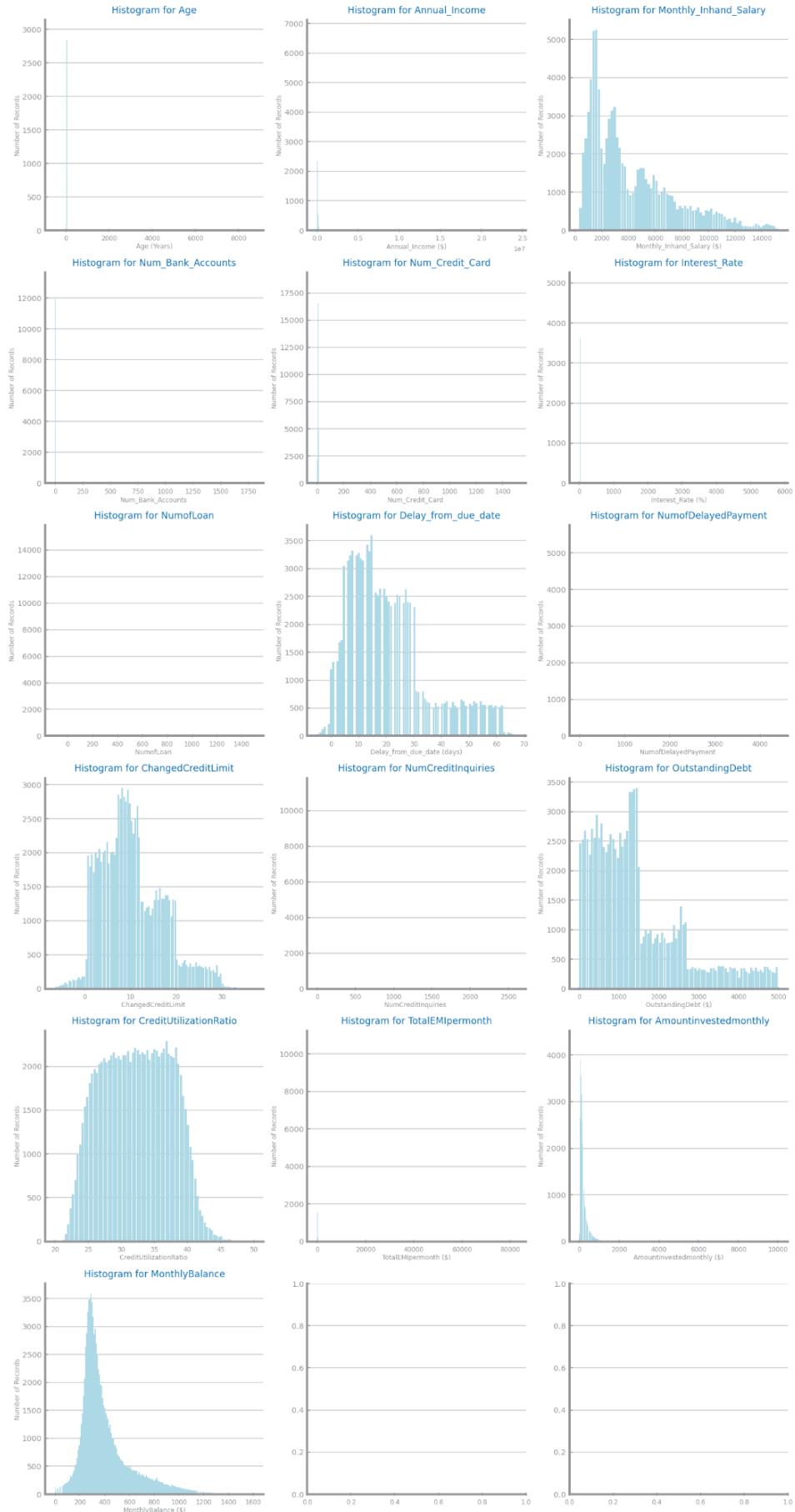


Figure 7.2 Binary Histograms for dataset 1



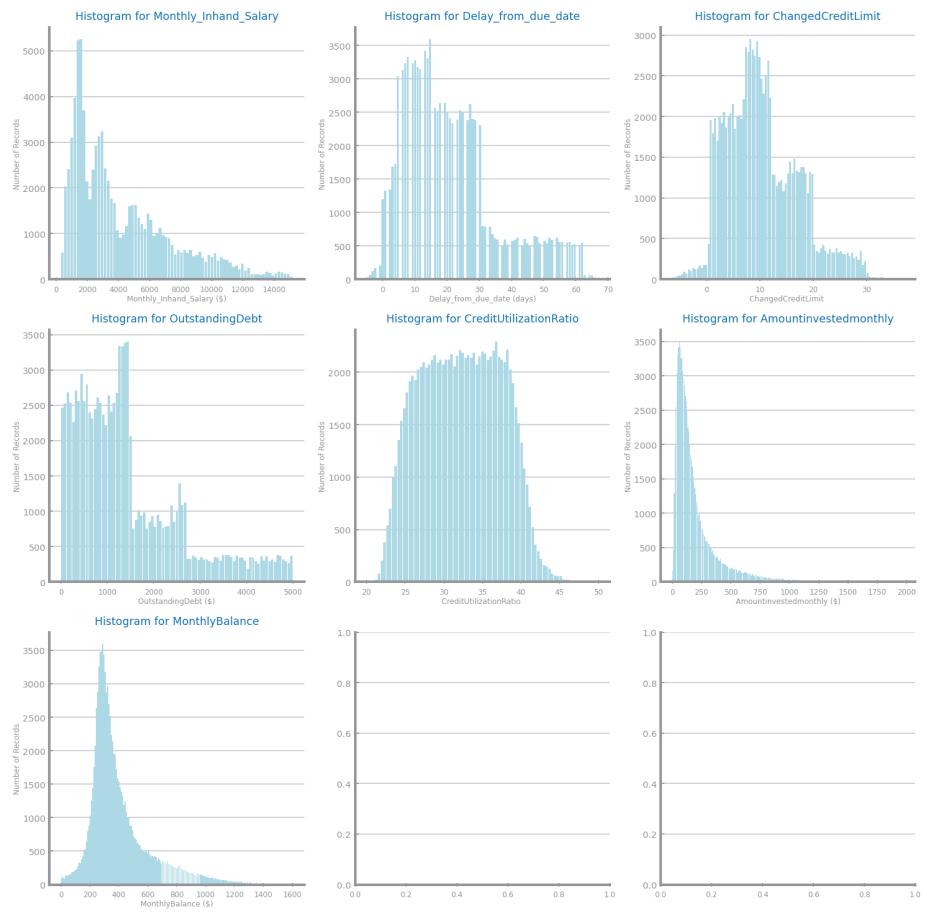


Figure 8 Numeric Histograms for dataset 2



Figure 8.1 Symbolic Histograms for dataset 2 (with distributions is enough)

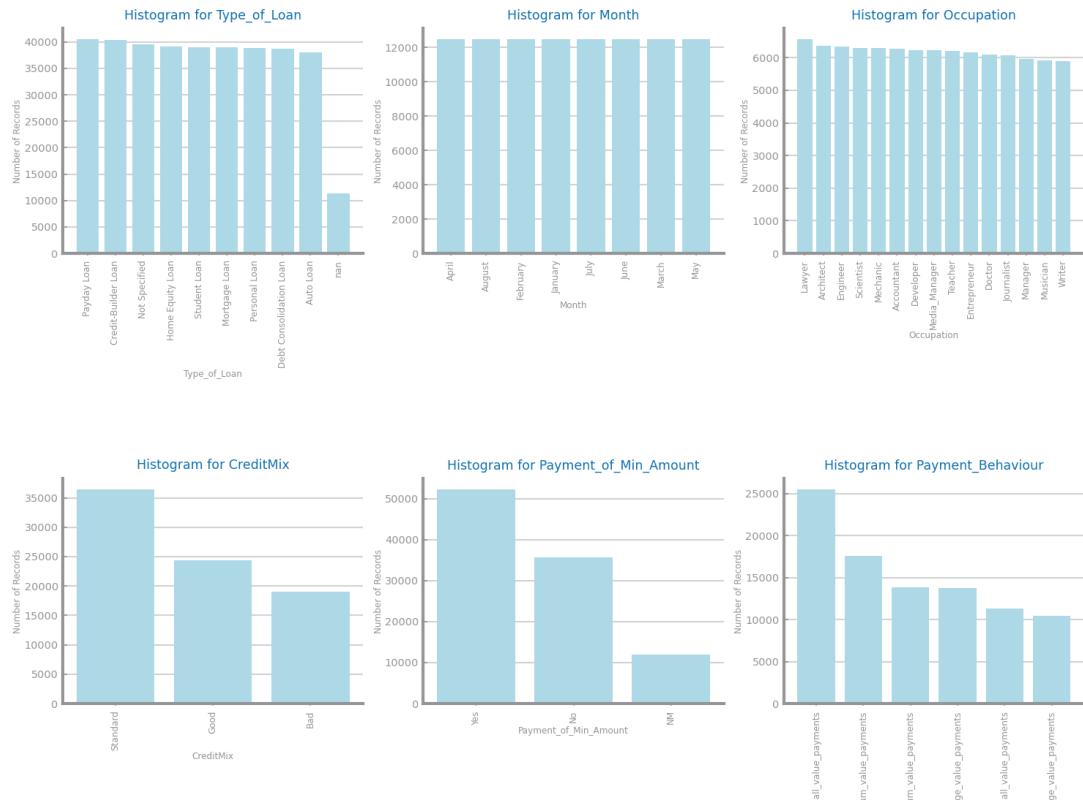


Figure 8.2 Relevant Symbolic Histograms for dataset 2

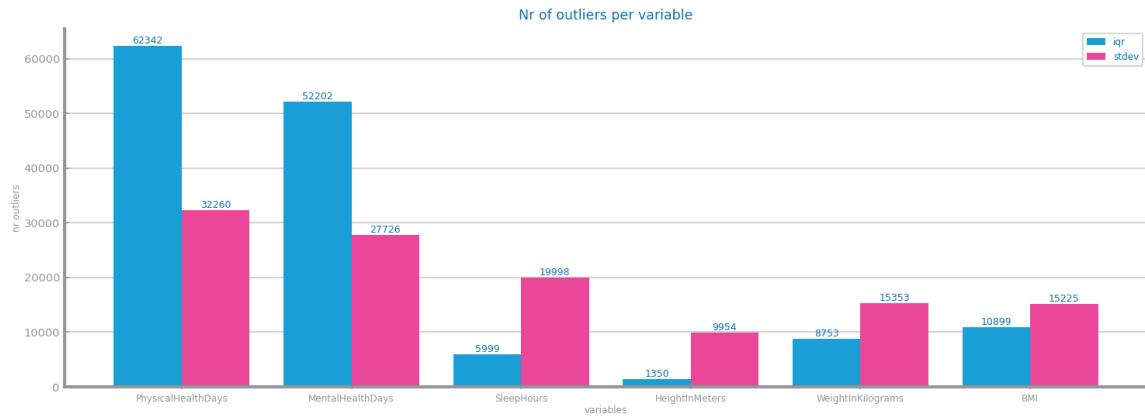


Figure 9 Outliers study dataset 1

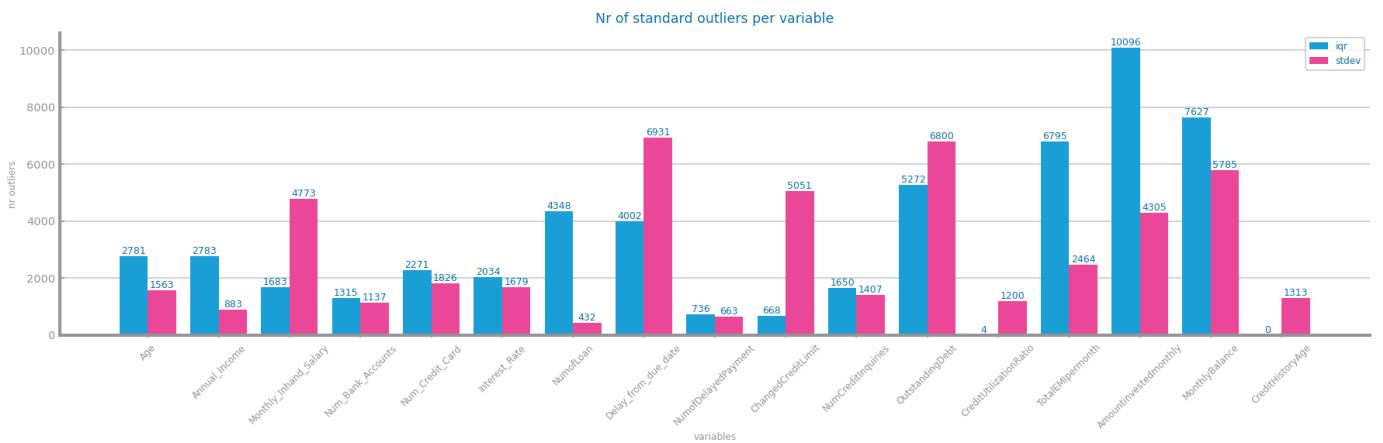


Figure 10 Outliers study for dataset 2

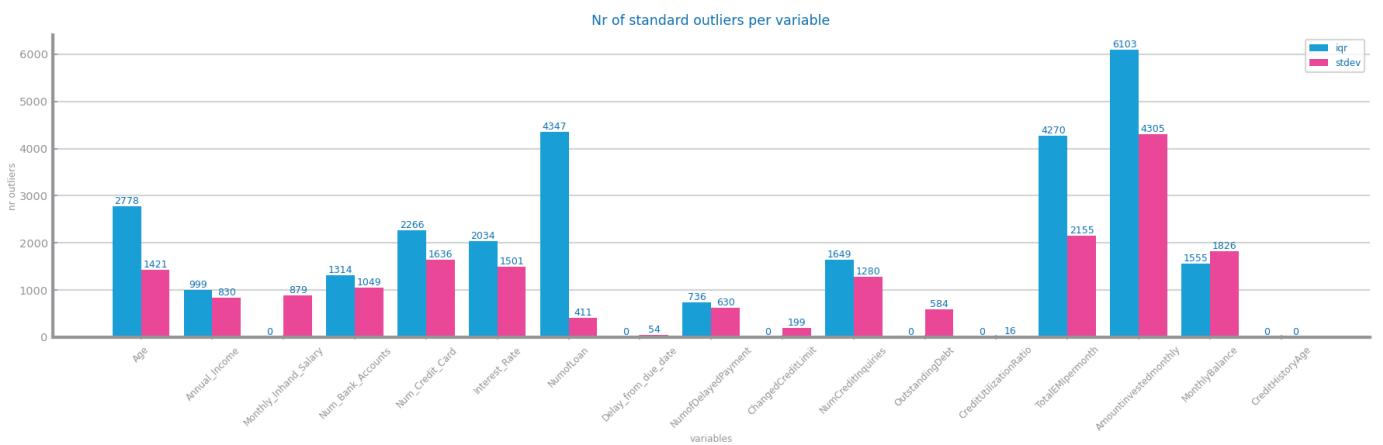


Figure 10.1 Outliers study for dataset 2

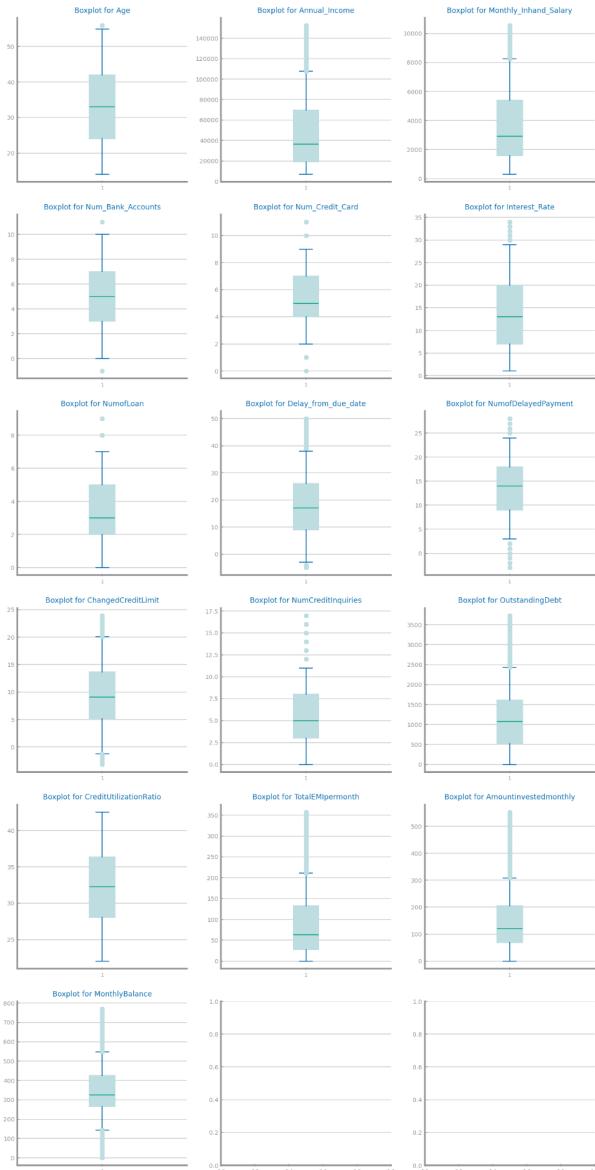


Figure 10.2 Boxplots without Outliers

### Target distribution (target=CovidPos)

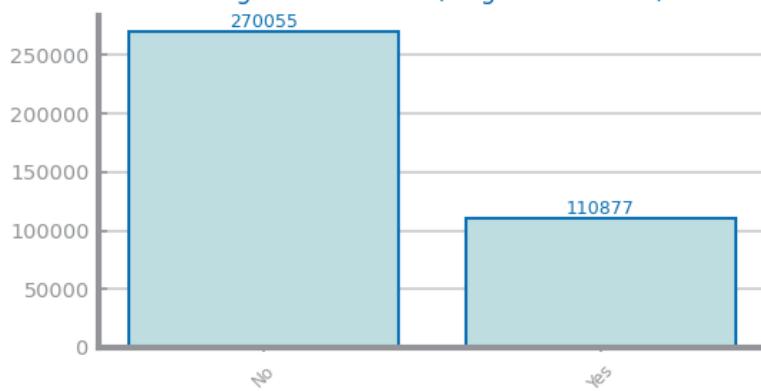


Figure 11 Class distribution for dataset 1

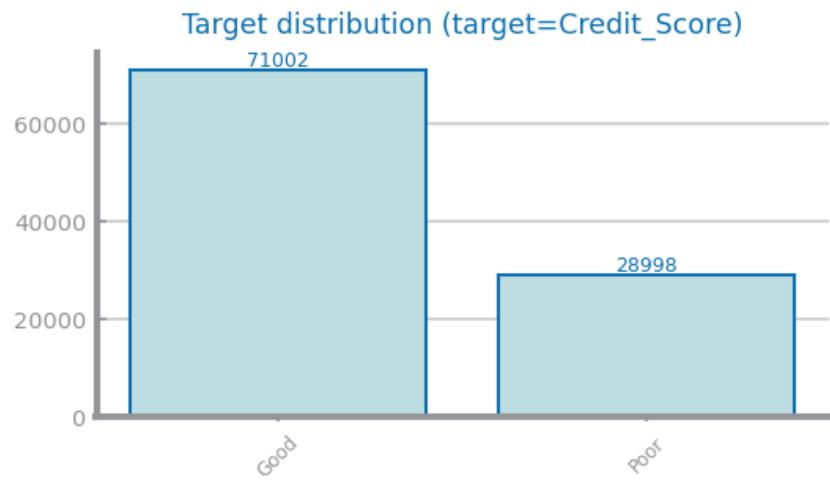


Figure 12 Class distribution for dataset 2

### Data Granularity

#### DS1

The age ranges have been categorized and segmented into groups.

Figure 13.3 displays the aggregated data of the 'state' variable, categorized by region.

#### DS2

Taxonomies -> 1. Occupation - divided in different sectors, Loan Risk - from Low to High , Loan Type - from real world experience

Confirm majority has *Log-n* distribution

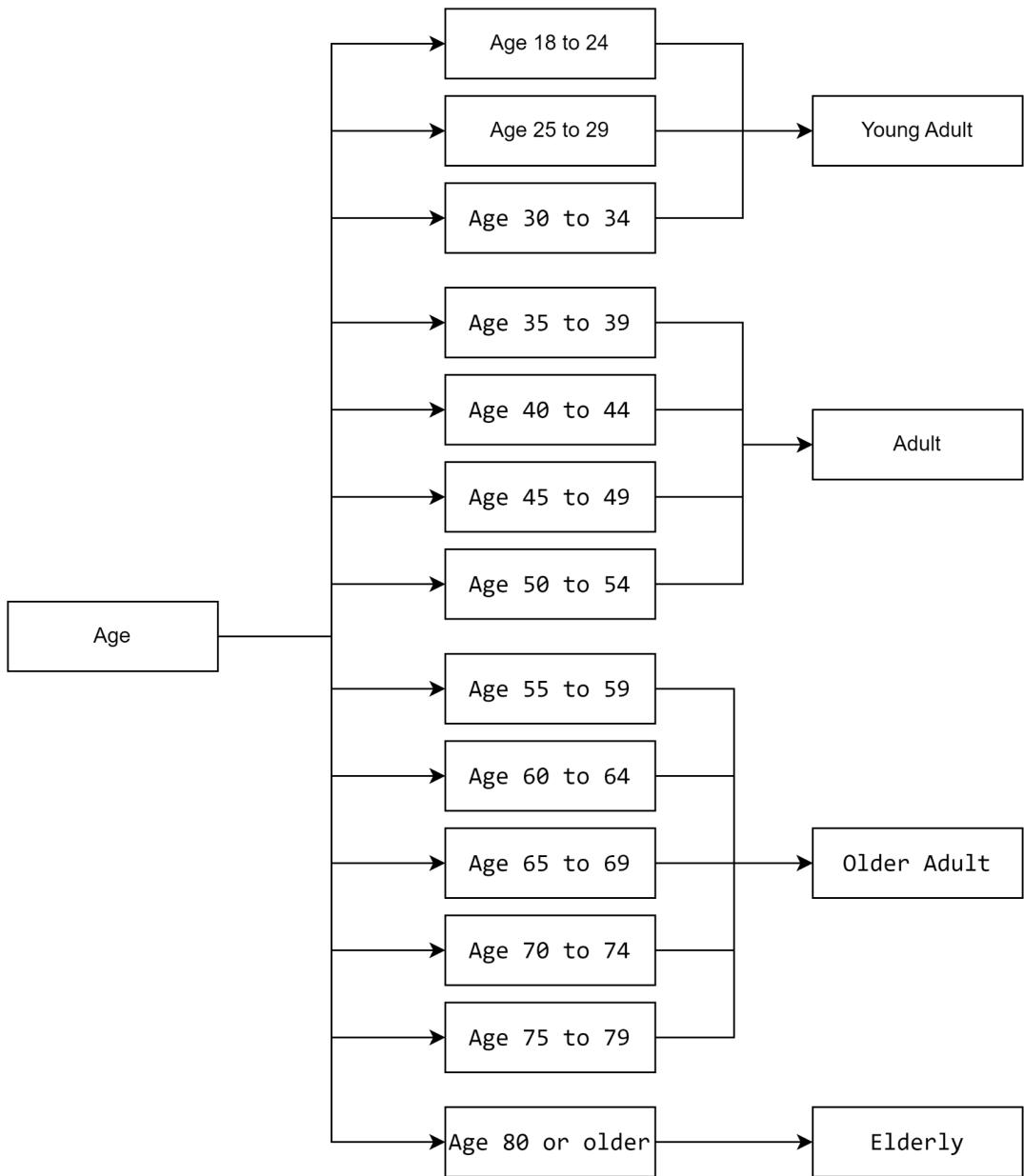


Figure 13.1 Granularity analysis for dataset 1 (Age)

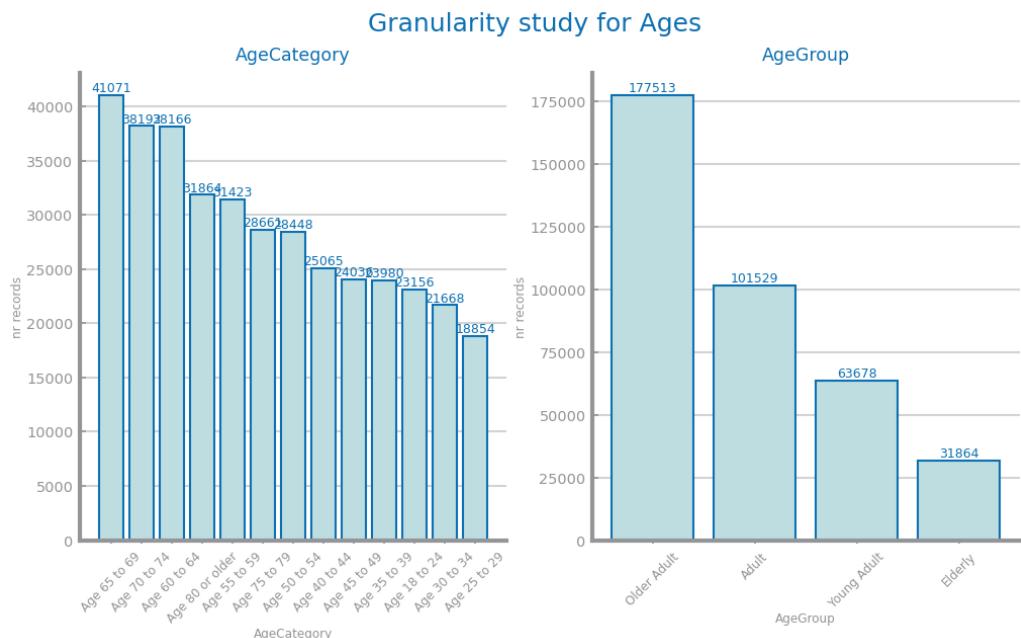


Figure 13.2 Granularity study for dataset 1 (Age)

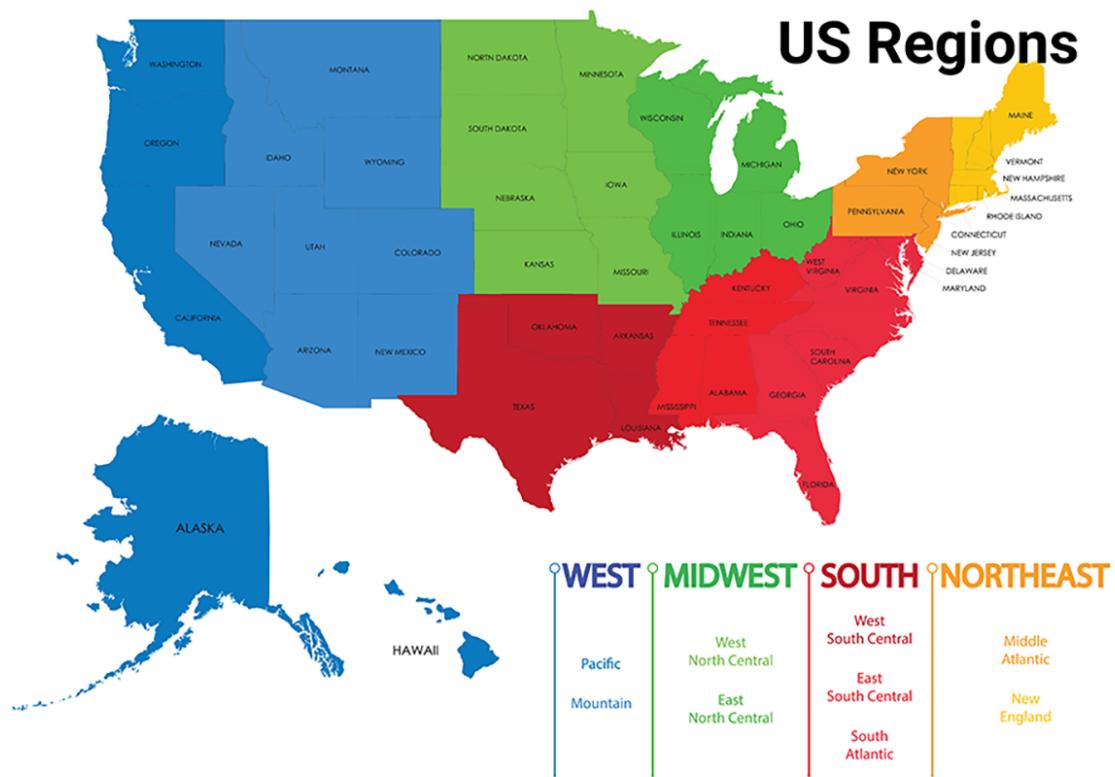


Figure 13.3 Granularity analysis for dataset 1 (State)

<sup>1</sup> <https://www.50states.com/city/regions.htm>

## Granularity study for Location

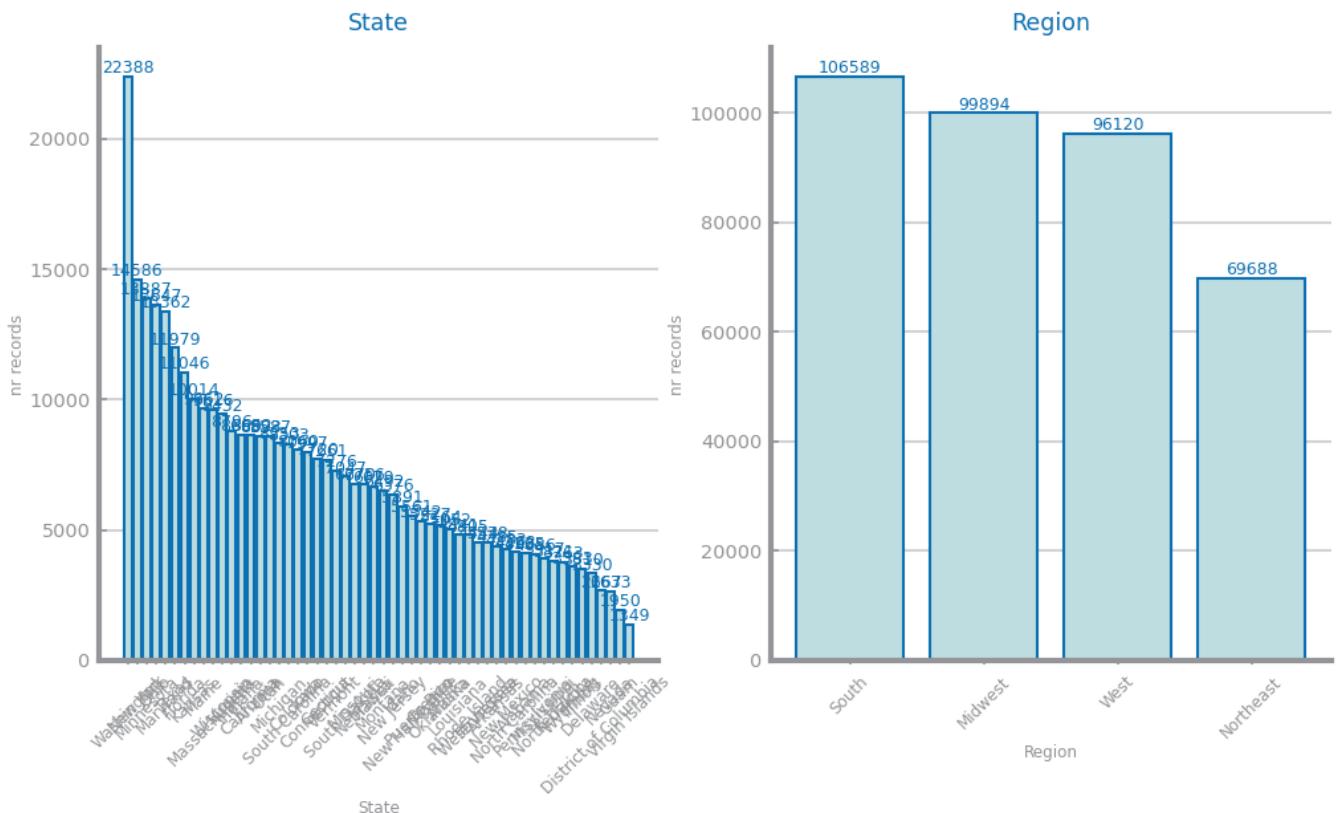
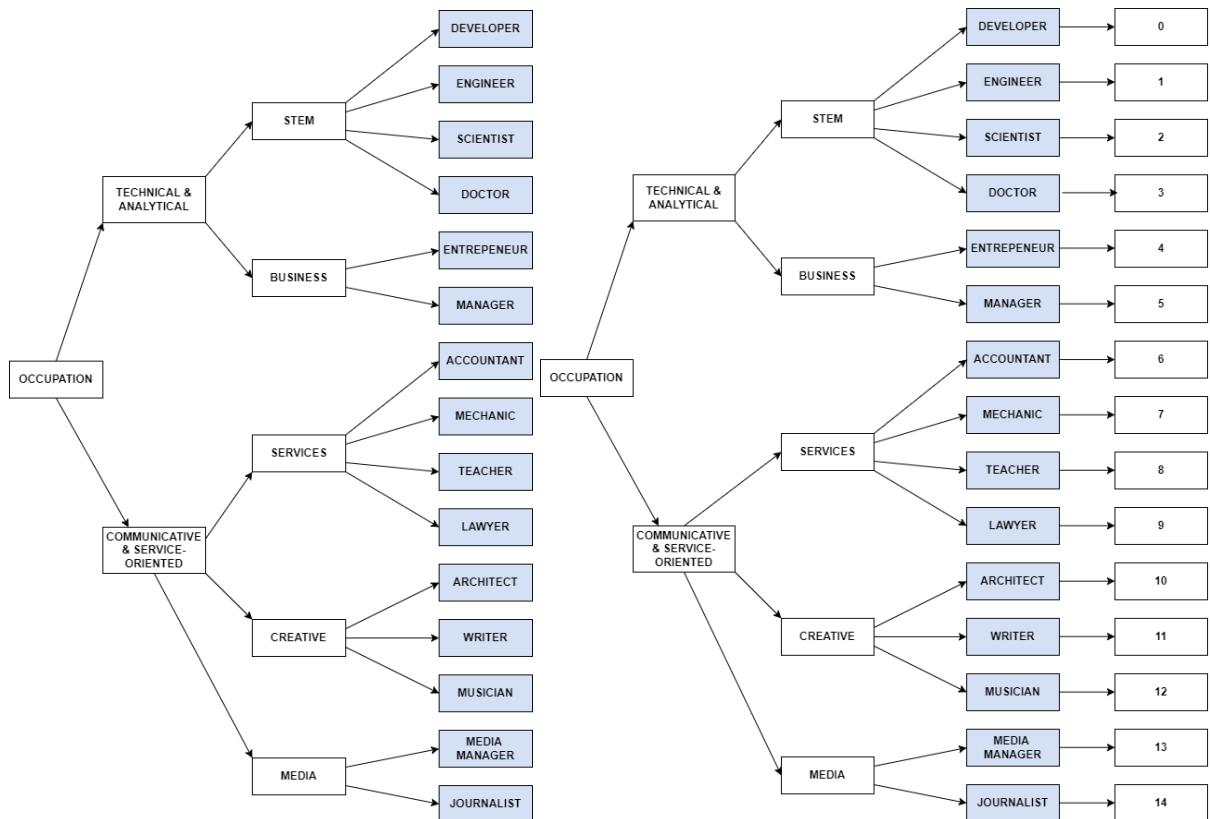
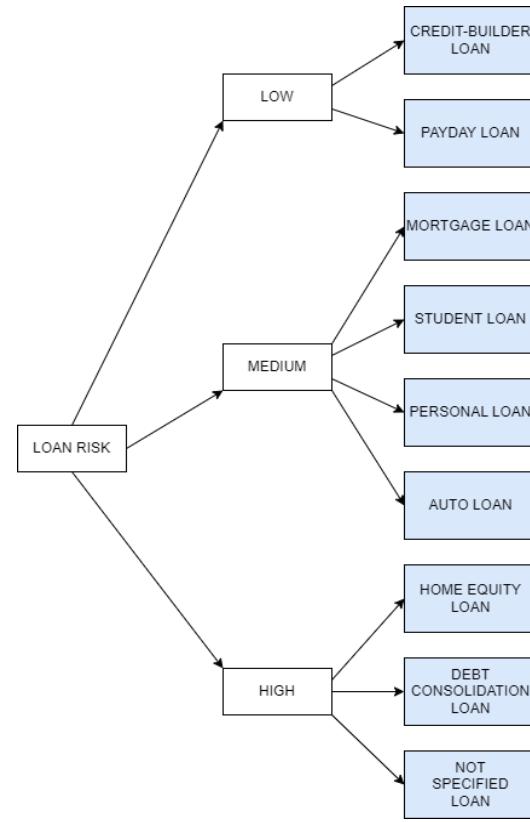


Figure 13.3 Granularity study for dataset 1 (State)

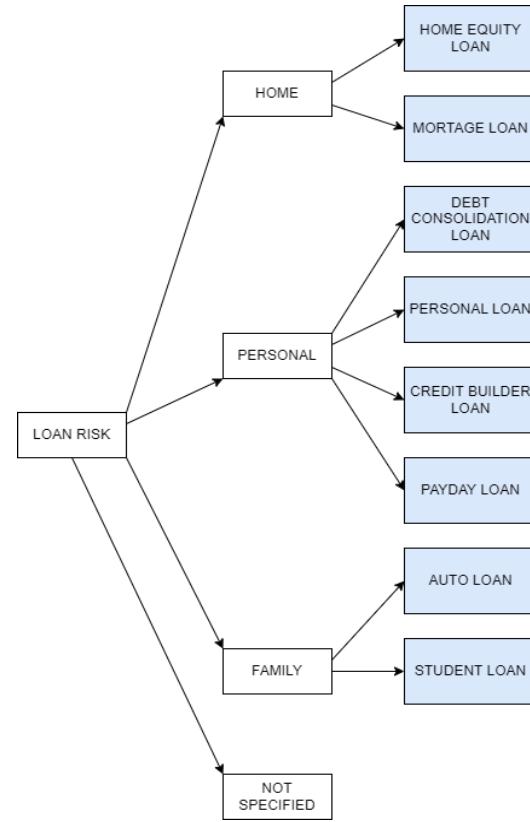
Below are our chosen taxonomies for dataset 2: Occupation, Loan Risk and Loan Type.



Explicar Taxonomia feita e valores para ordinal taxonomy encoding

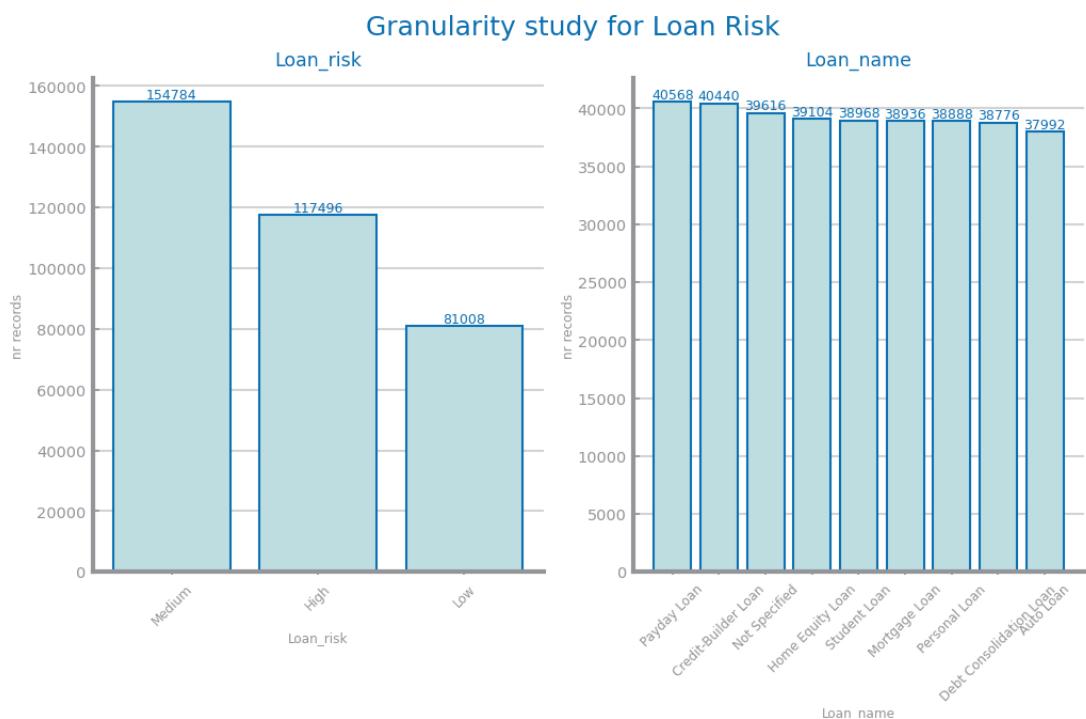
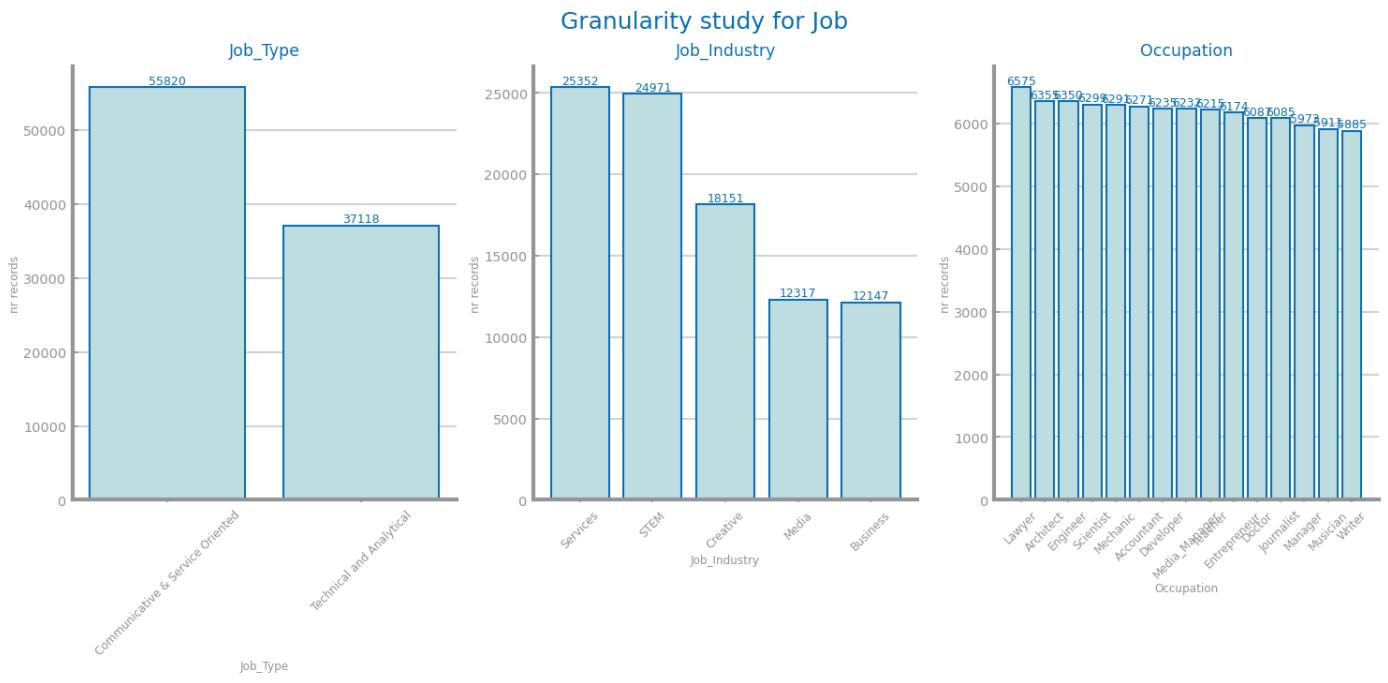


*Explicar Taxonomia feita e valores para ordinal taxonomy encoding*



*Explicar Taxonomia feita e valores para ordinal taxonomy encoding*

And the corresponding granularity analysis:



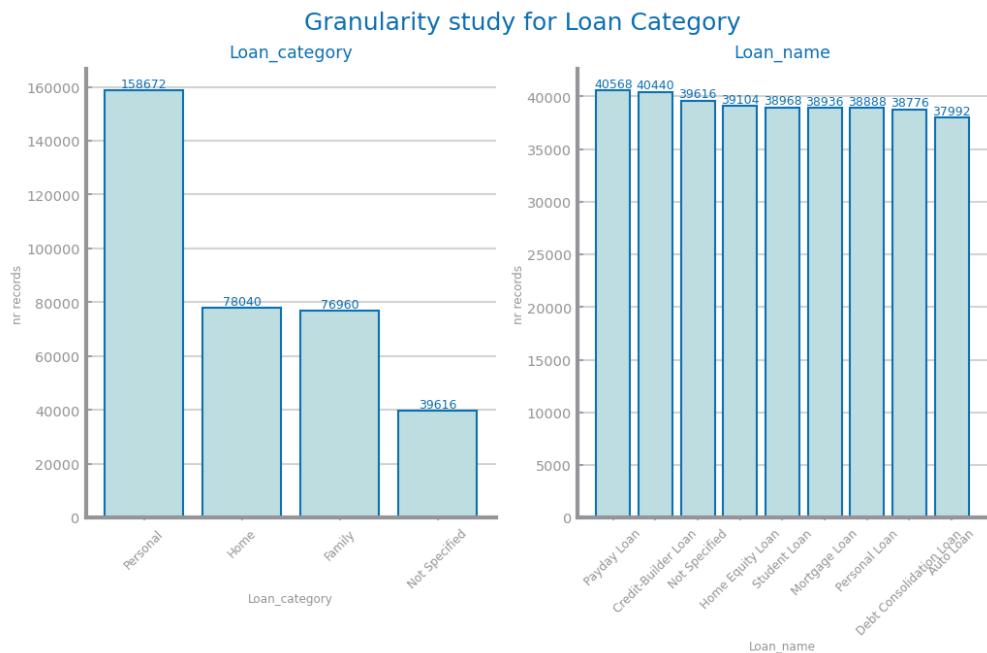


Figure 14 Granularity analysis for dataset 2

## Data Sparsity

### DS1

Challenges w/ scatter plots w/ many qualitative vars

High *BMI* correlation with *weight* & *height* (fig15 row5 plot1)

### DS2

Data well covered (**Good Sparsity**); issues due w/ outliers & incorrect entries

Many qualitative features

HeatMap shows good correlations *OutstandingDebt* & *Delay\_From\_due\_date*, *Credit\_History\_Age* & *OutStandingDebt*, *Payment\_of\_Min\_Amount* & *Credit\_History\_Age*, *Credit\_History\_Age* & *Delay\_From\_due\_date*

Considering Naive Bayes assumption of independence, **reduction of features is advised**



Figure 15 Sparsity analysis for dataset 1

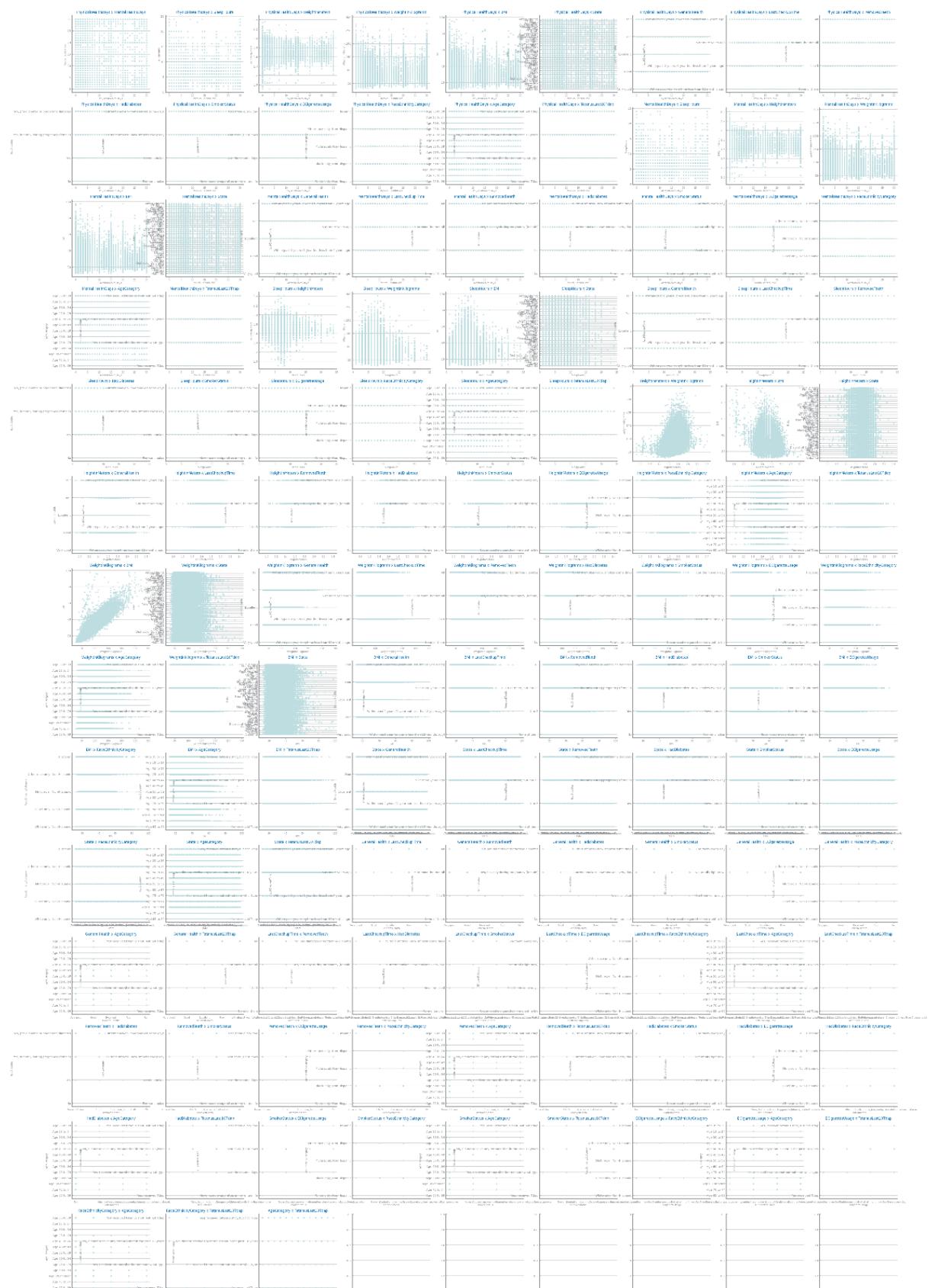


Figure 15.1 Sparsity analysis for dataset 1 without empty plots

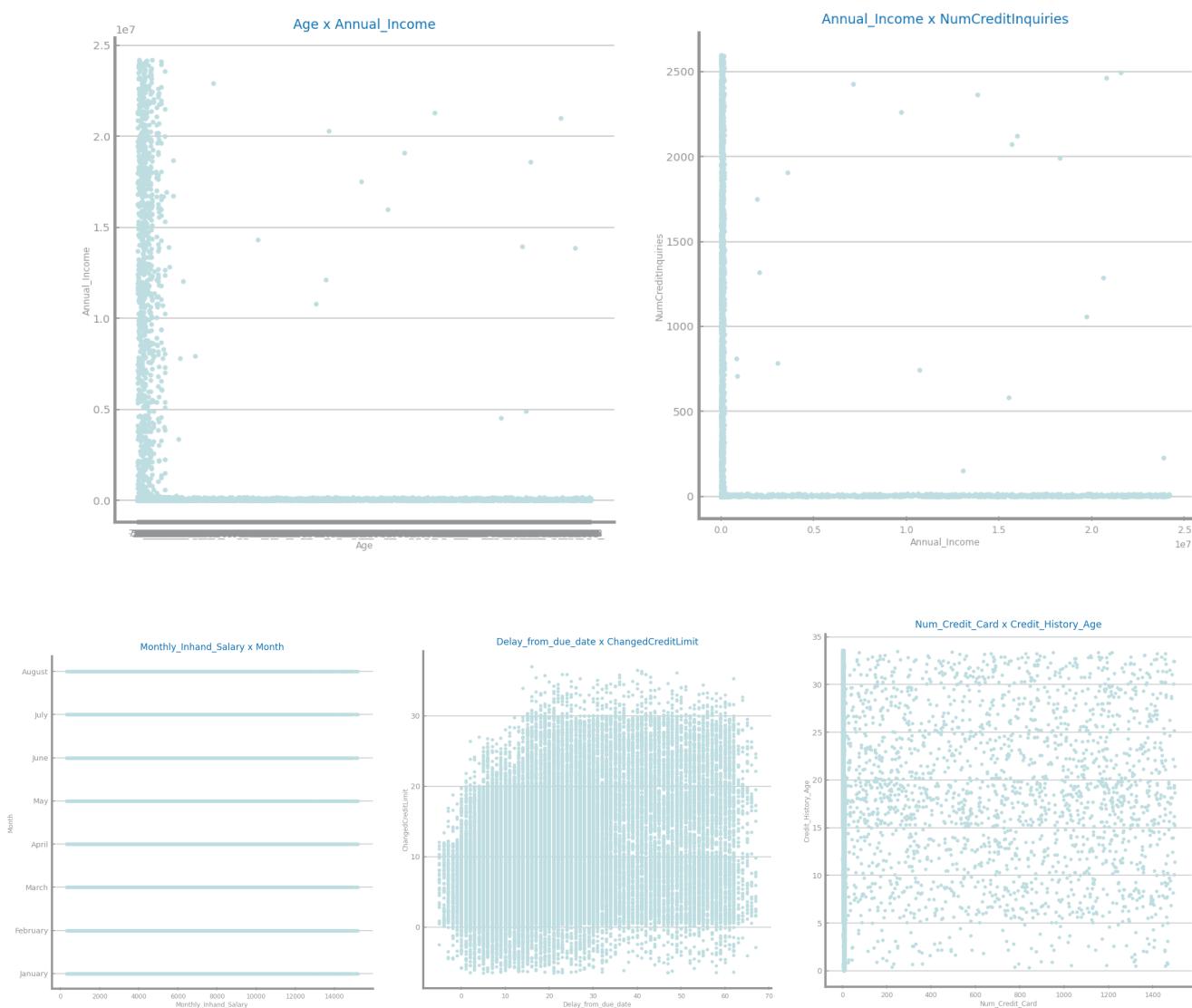


Figure 16 Sparsity analysis for dataset 2

### Correlation Analysis

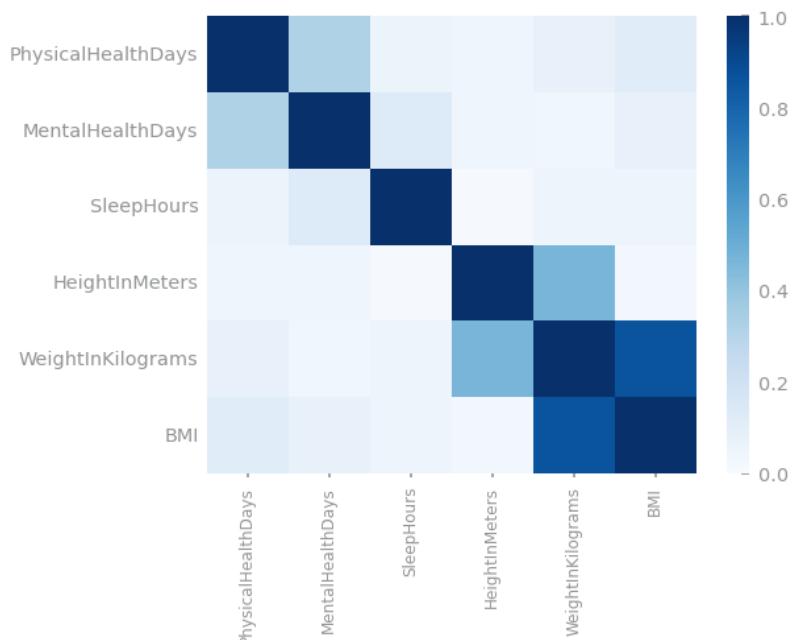


Figure 17 Correlation analysis for dataset 1

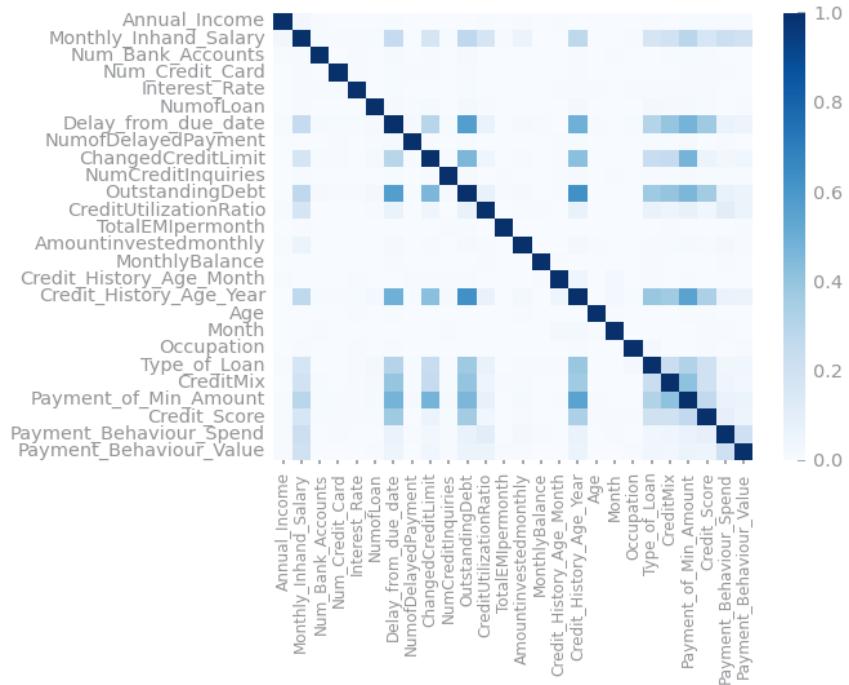


Figure 18 Correlation analysis for dataset 2

## 2 DATA PREPARATION

### Variables Encoding

#### DS1

**State:** replaced states with latitude and longitude values.

**Ordinal Linear:** GeneralHealth, LastCheckupTime, RemovedTeeth, HadDiabetes, SmokerStatus, ECigaretteUsage, RaceEthnicityCategory, AgeCategory, TetanusLast10Tdap, AgeGroup

**Ordinal Linear (from taxonomy):** AgeCategory, Regions

#### DS2

ID used as **PK**

**Drop identifying columns (check & clean)** - Customer\_ID, SSN, Name

**Ordinal Linear** Credit\_Score (*binary*), CreditMix, Payment\_of\_Min\_Amount

**Ordinal Taxonomy** Payment\_Behaviour, Occupation (*Granularity Section*)

**Transform to numeric** Credit\_History\_Age

**Cyclic** Month, Credit\_History\_Age

**Other** Type\_of\_Loan - Mix One-hot & Ordinal Taxonomy, Each type gets a column. Counter increment according to *Taxonomy* rules specified in *Granularity Section*, aiding in separate frequency and trait analysis.

### Missing Value Imputation

#### DS1- Continue w/ knn mean

Very small changes but we used knn mean due to better results

#### DS2 - Continue w/ Approach 1

Invalid 'Age', 'Num\_Bak\_Accounts', 'NumofLoan' values set as MV

Fig3.1 shows most MV under 15%, except 'CreditMix'

Alternatives:

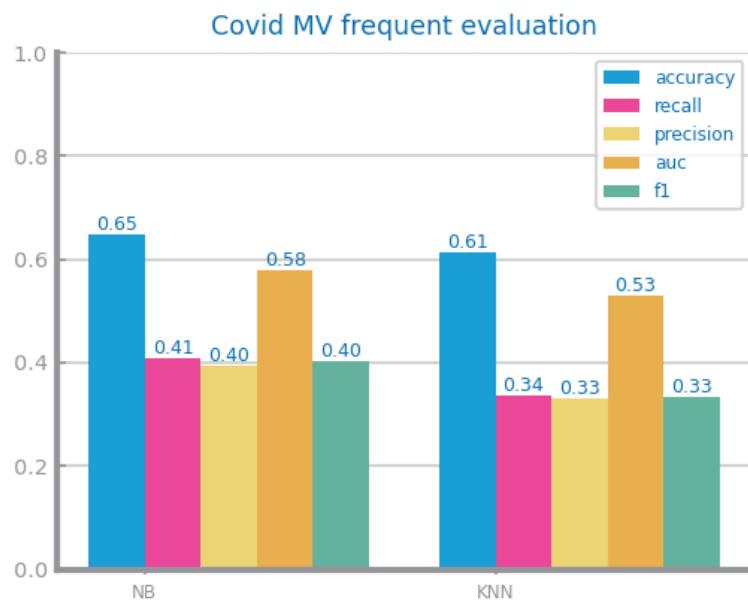
1° Eliminate columns above certain Threshold

- Columns > 15%, Rows > 16%

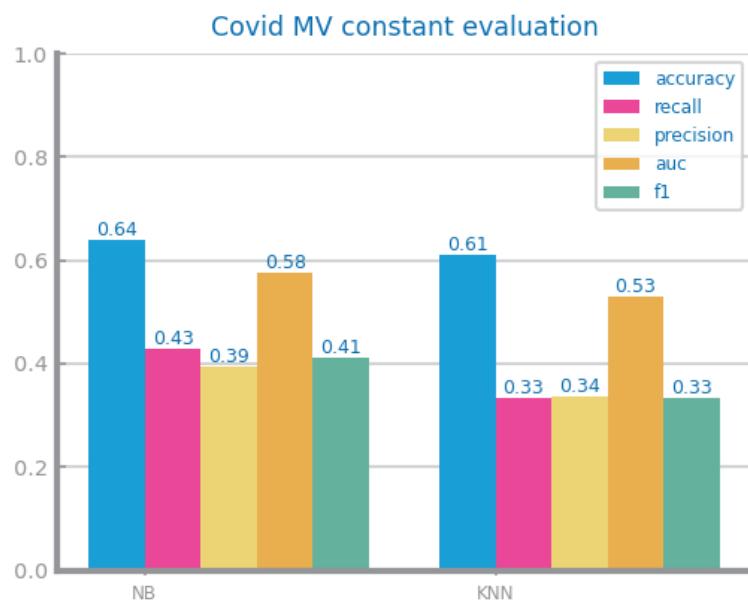
- Fill MV Mean Numerics, Most Freq Symbolic

2° Fill MV Median & Most Freq

3° Don't eliminate col/row KNN Imputer



*Figure 19.1 Missing values imputation results with the most frequent approach for dataset 1*



*Figure 19.2 Missing values imputation results with the constant approach for dataset 1*

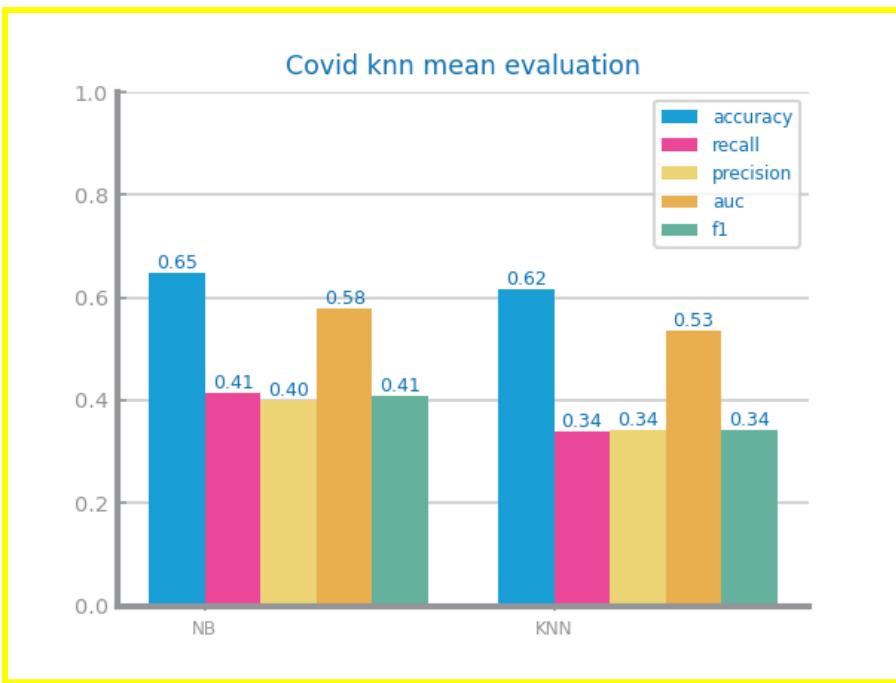


Figure 19.3 Missing values imputation results with KNN mean approach for dataset 1

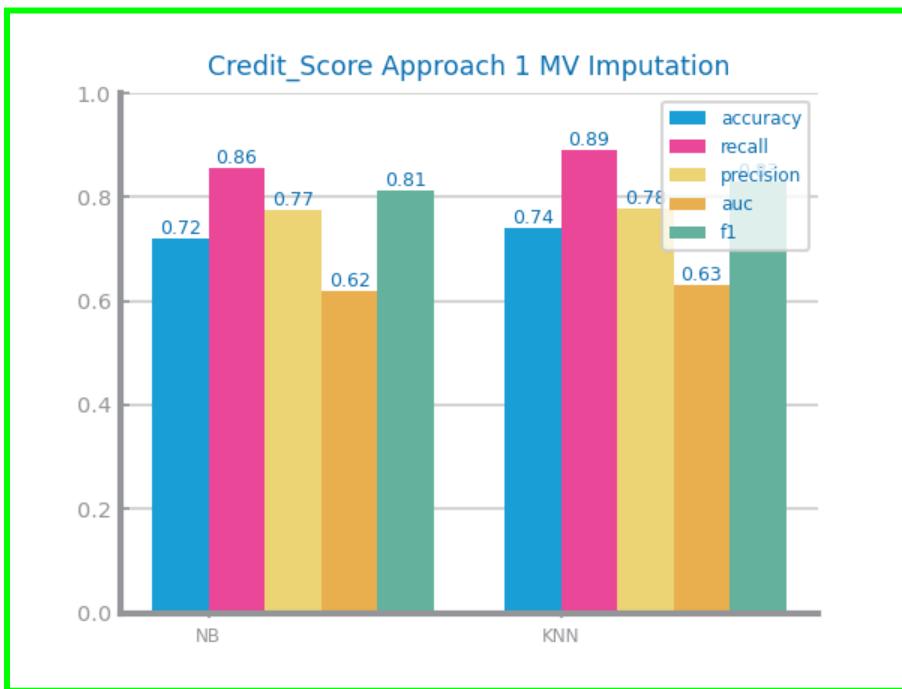
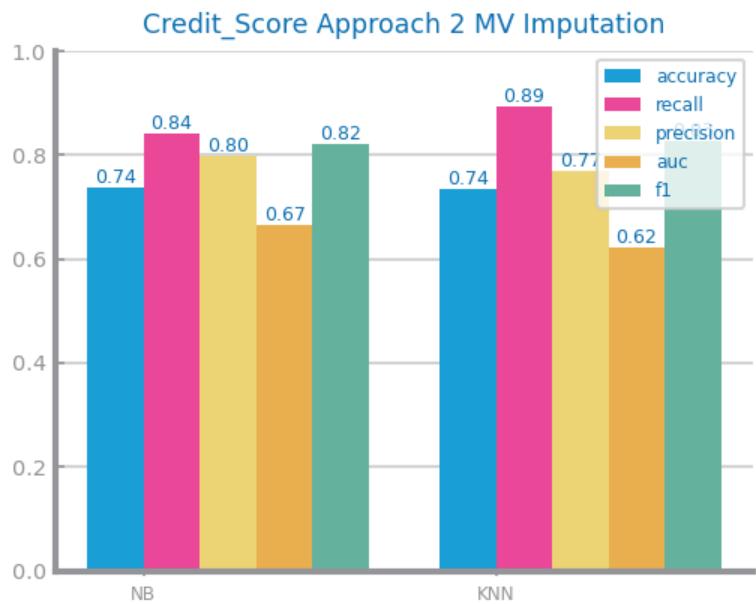
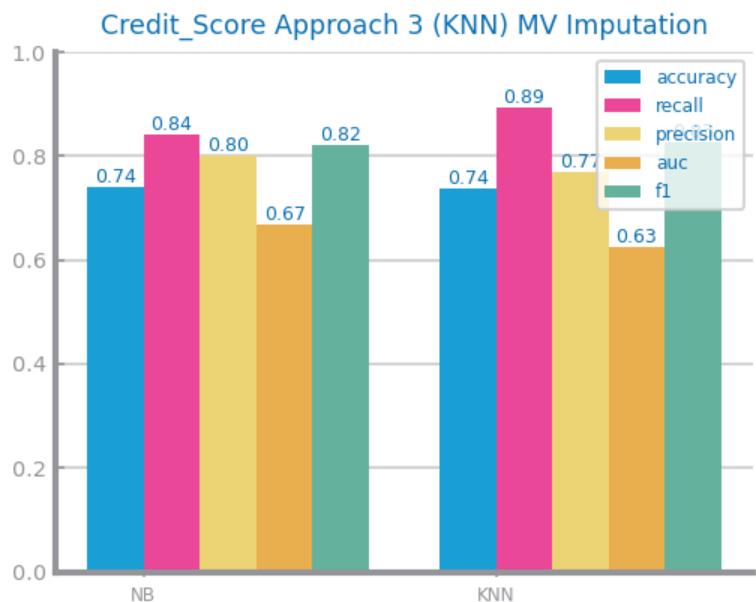


Figure 20 Missing values imputation results with approach 1 for dataset 2



*Figure 20.1 Missing values imputation results with approach 2 for dataset 2*



*Figure 20.2 Missing values imputation results with approach 3 for dataset 2*

## ***Outliers Treatment***

### **DS1**

drop outliers (dropped 168324 records) was the chosen one because of the better results. The best of the replaces was the mean(fig 21.2)

### **DS2 - Continue w/ Approach 2 (mode)**

Alternatives (*values and other statistics are adjusted depending on the strategy*):

1º Drop - dropped out **13543** records. *too many*

2º Replaced w/ fixed value

3º Truncate

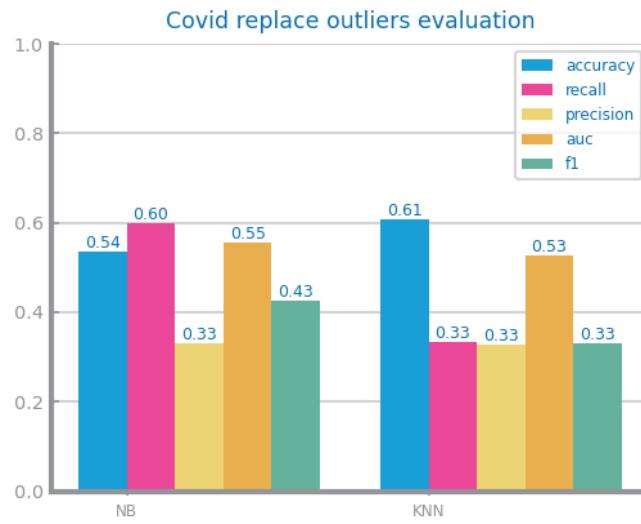


Figure 21.1 Outliers imputation results with replace with median approach for dataset 1

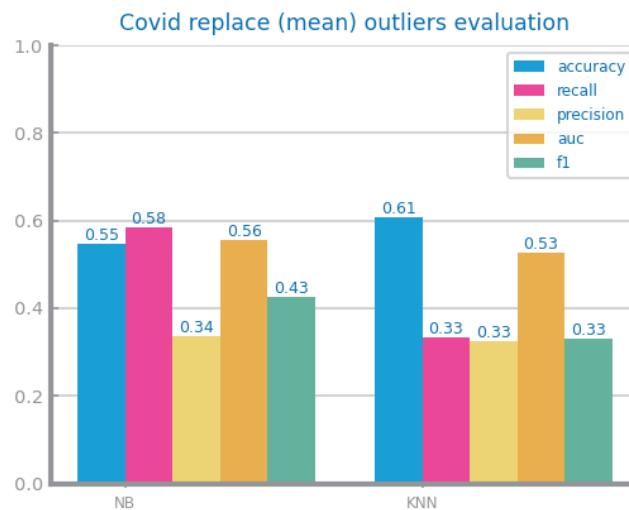


Figure 21.2 Outliers imputation results with replace with mean approach for dataset 1

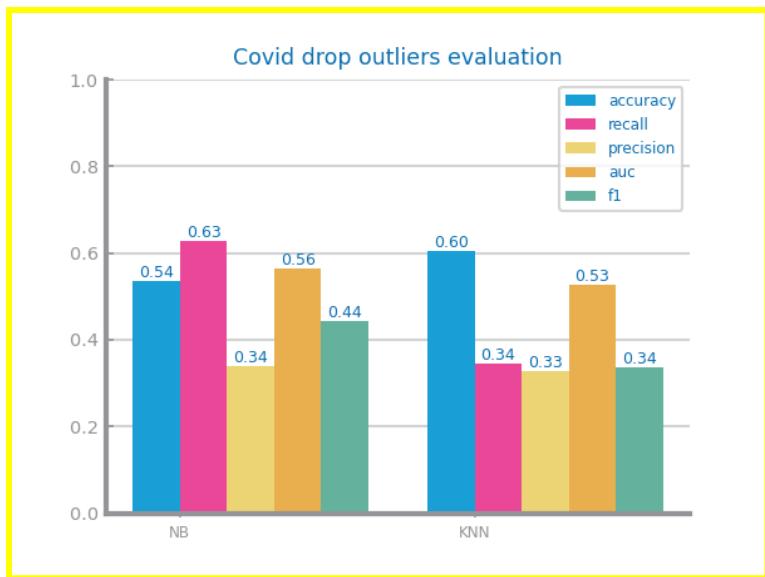


Figure 21.2 Outliers imputation results with drop approach for dataset 1

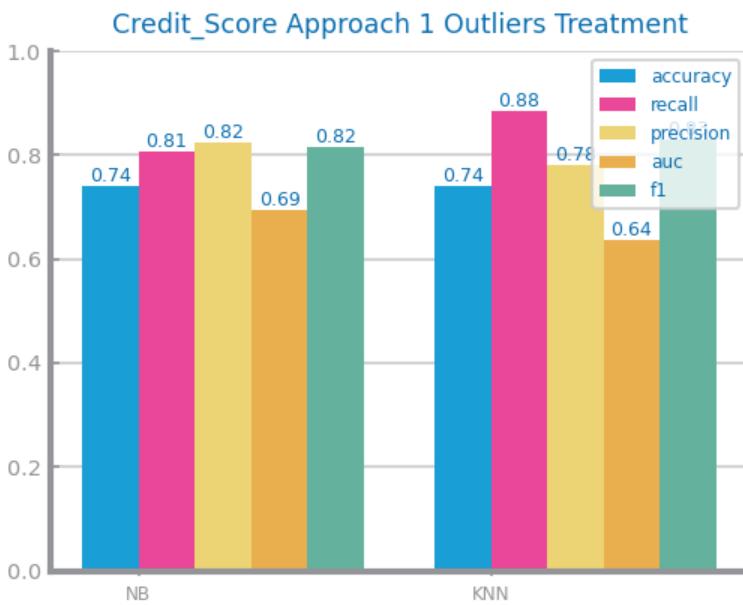


Figure 22 Outliers imputation results with approach 1 (DROP) for dataset 2

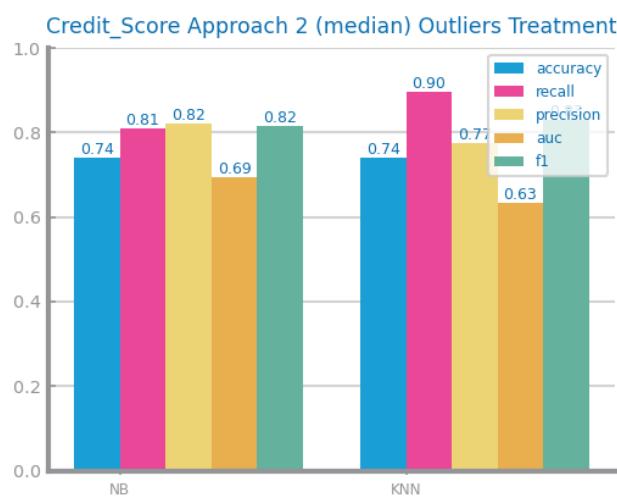
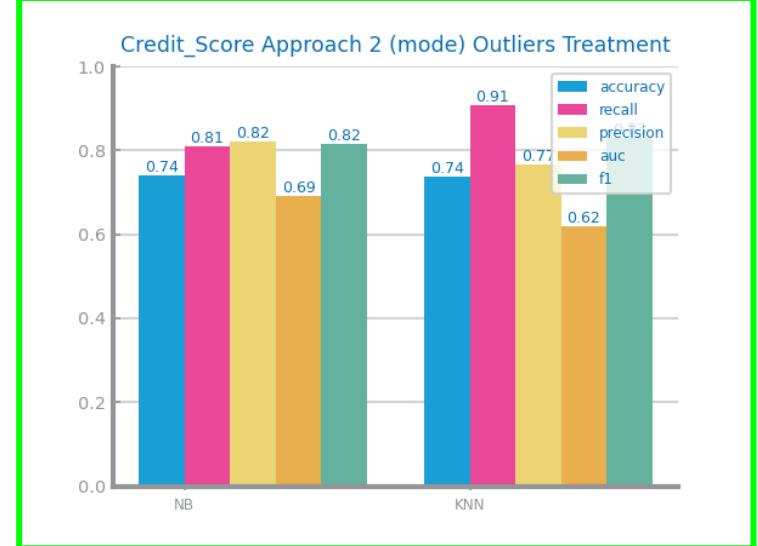
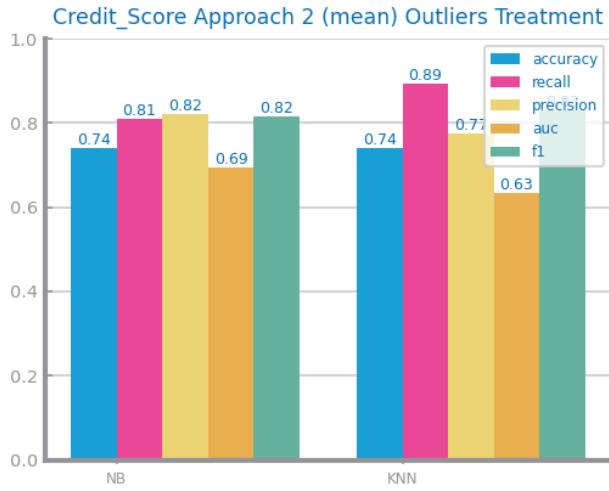


Figure 22.1 Outliers imputation results with approach 2 (FILL) for dataset 2

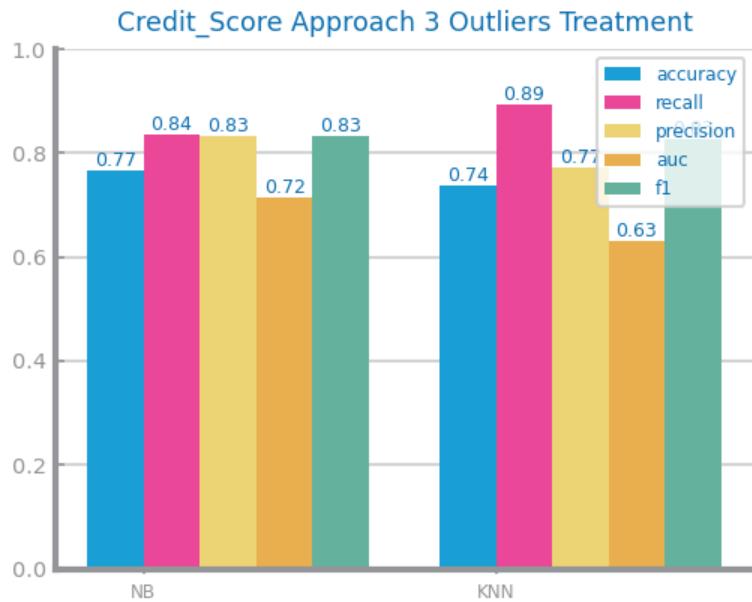


Figure 22.2 Outliers imputation results with approach 3 (TRUNCATE) for dataset 2

## Scaling

### DS1

Results on NB should be identical, discrepancy prob due to numbers being too small & data structures not prepared

Analyzing knn we opted by not using scaling

### DS2 - Continue w/ Approach 1

Outliers far from the main clusters

Alternatives:

1º Standard Scaler - for Gaussian

2º MinMax - used for Neural Networks; reduce the impact of outliers

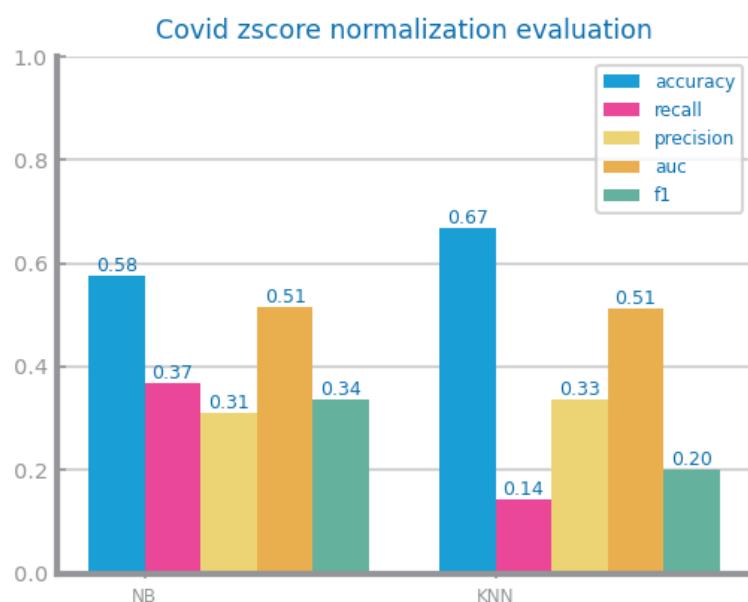


Figure 23.1 Scaling results with (Z-SCORE) normalization approach for dataset 1

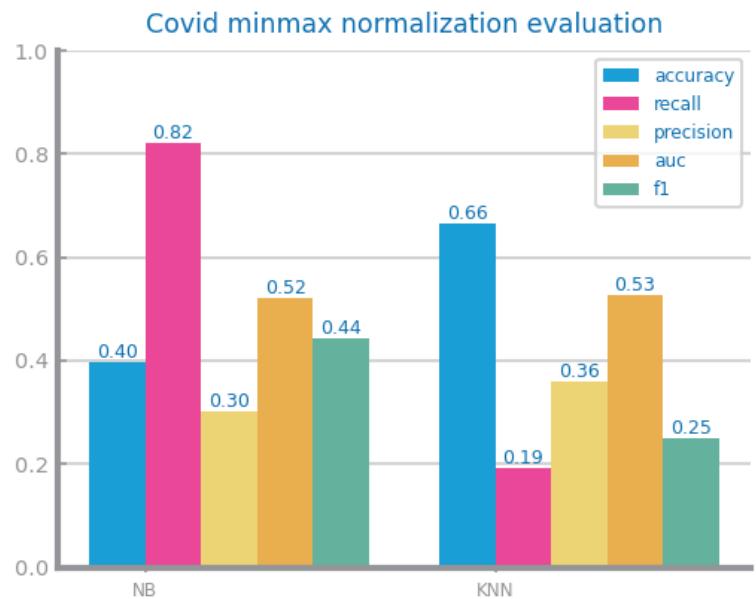


Figure 23.2 Scaling results with (MINMAX) normalization approach for dataset 1

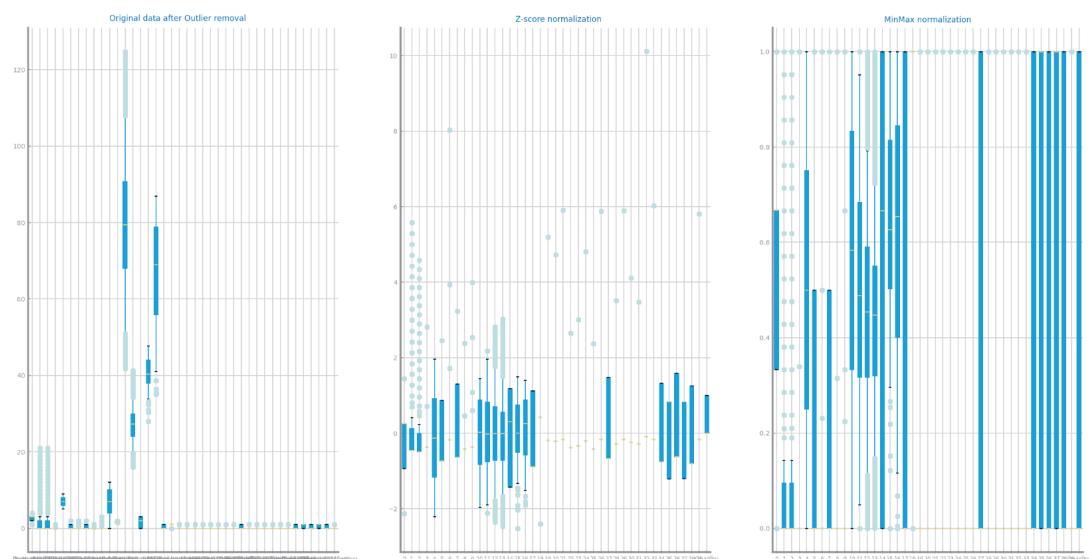


Figure 23.3 Scaling Boxplots for dataset 1

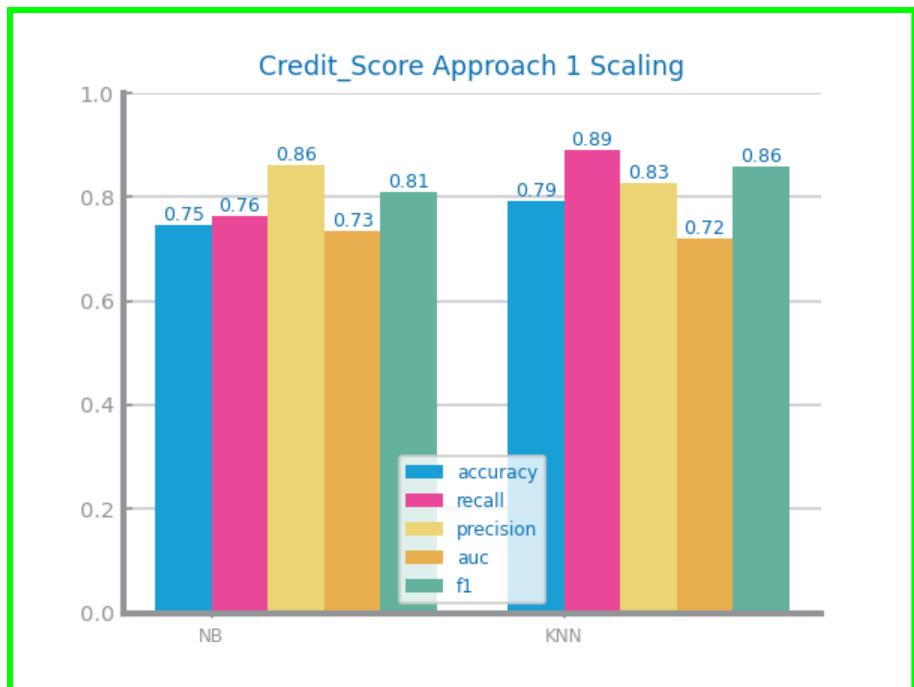


Figure 24 Scaling results with approach 1 (Z-SCORE) for dataset 2

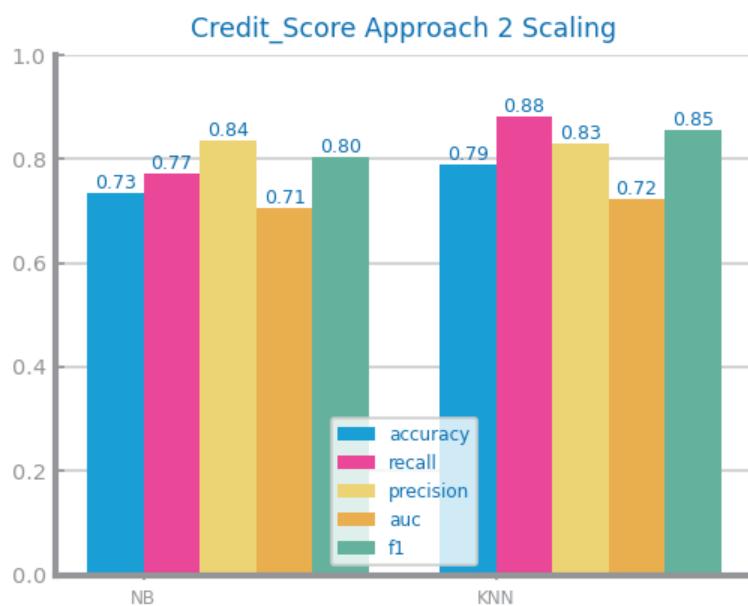


Figure 24.1 Scaling results with approach 2 (MINMAX) for dataset 2

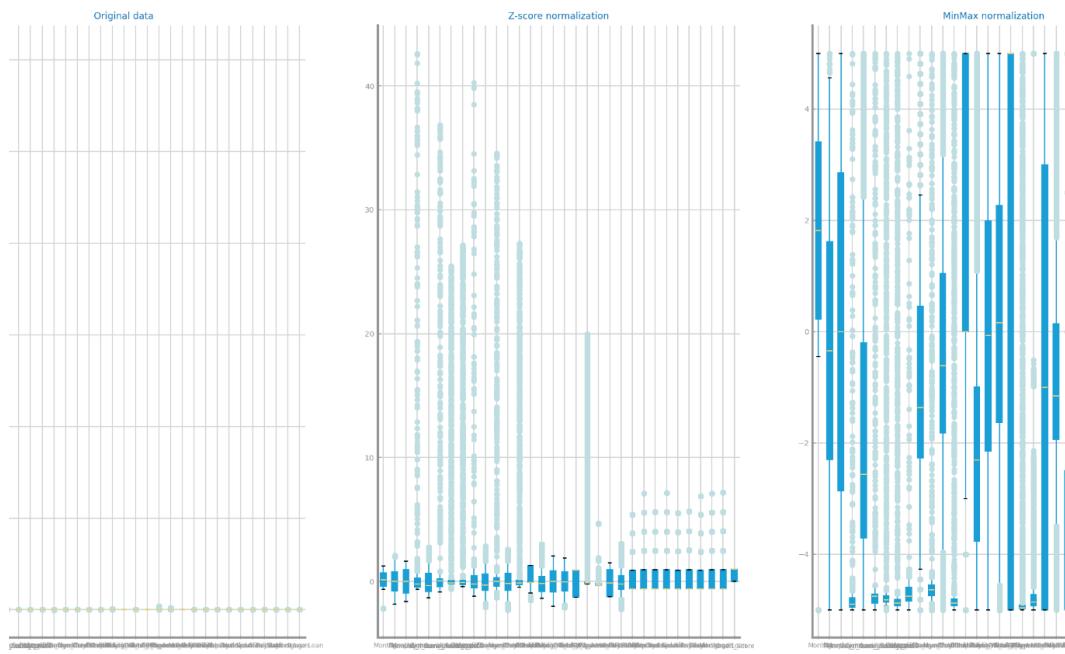


Figure 24.2 Scaling Boxplots for dataset 2

## Balancing

### BALANCE TRAINING SET ONLY

#### DS1- Continue w/ SMOTE

Big imbalance (fig25)

over&under- recall is better because minority class has more importance

SMOTE-best accuracy due to more diverse samples in minority class

#### DS2 - Continue w/ Approach 2

Fig12 reveals significant class imbalance, causing models to neglect minority class.

Alternatives:

1° Under - Loses major class info

2° Over - Viable, but larger & costly dataset

3° SMOTE - creates synthetic minority instance for better generalization

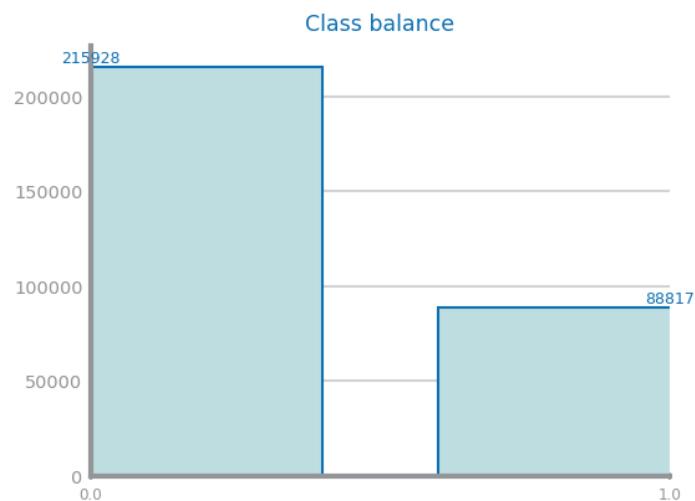


Figure 25 Class distribution before balancing for dataset 1

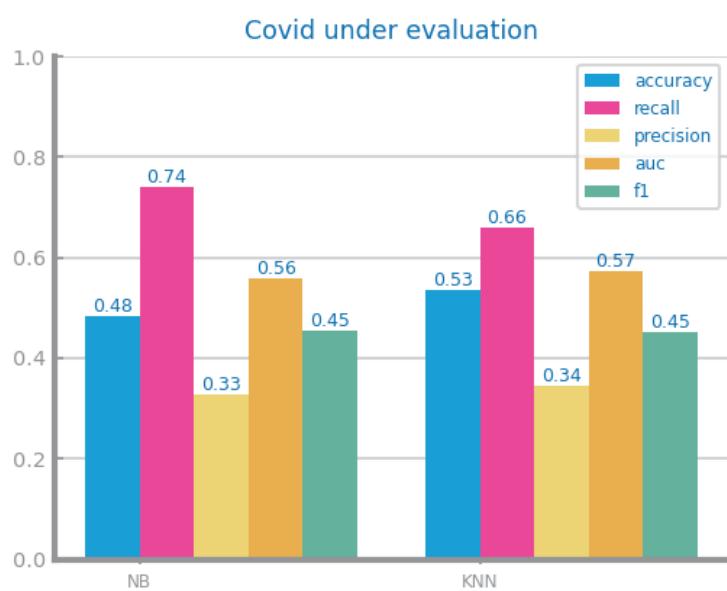
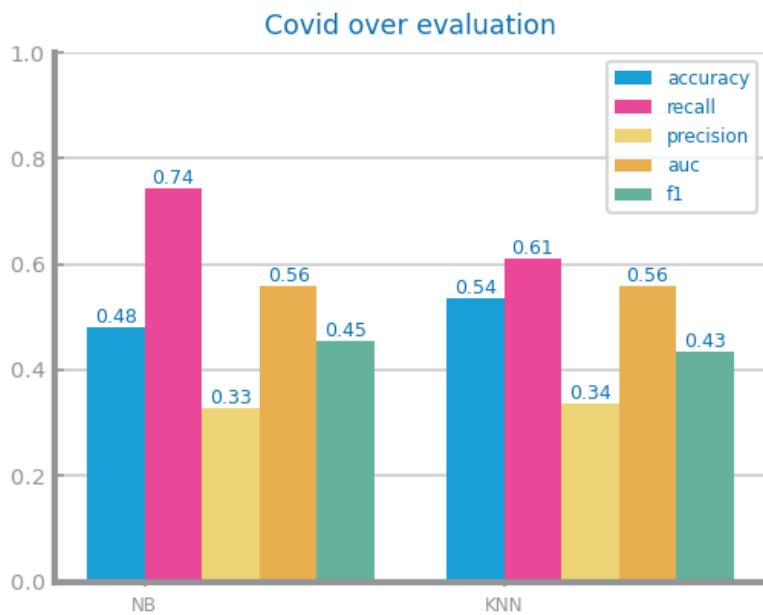
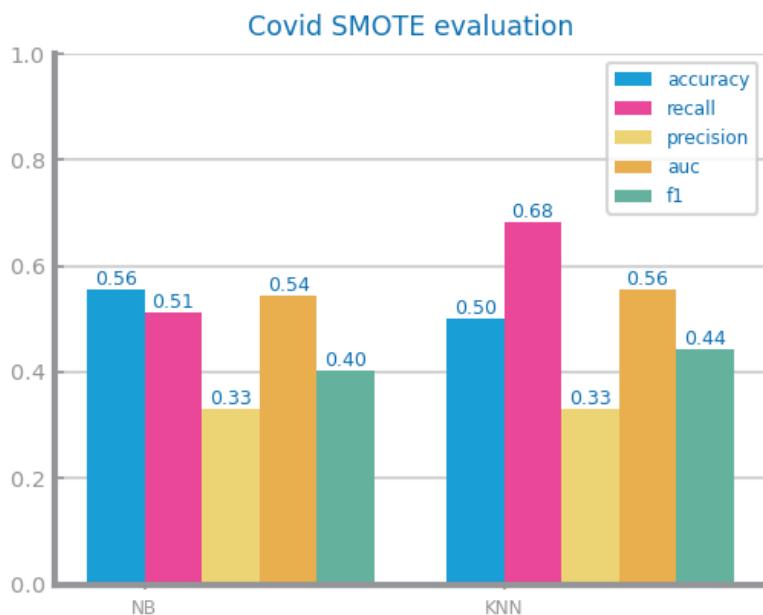


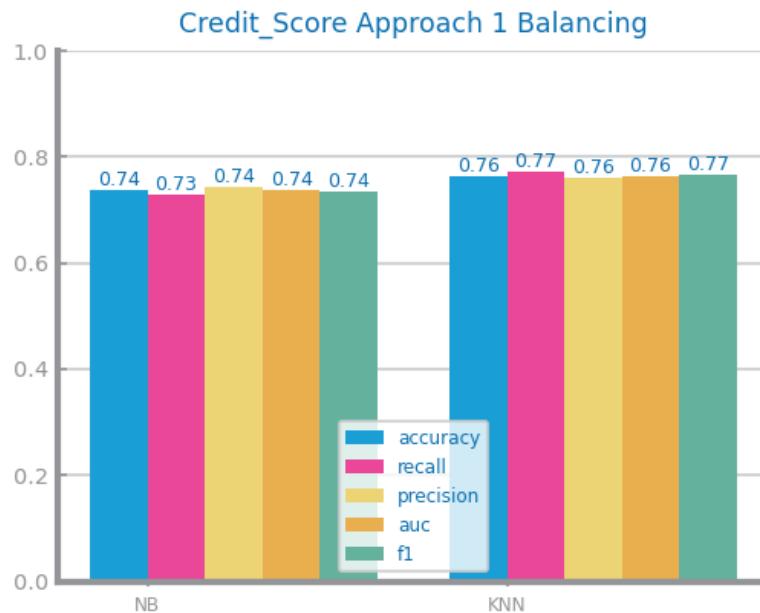
Figure 25.1 Balancing results with under approach for dataset 1



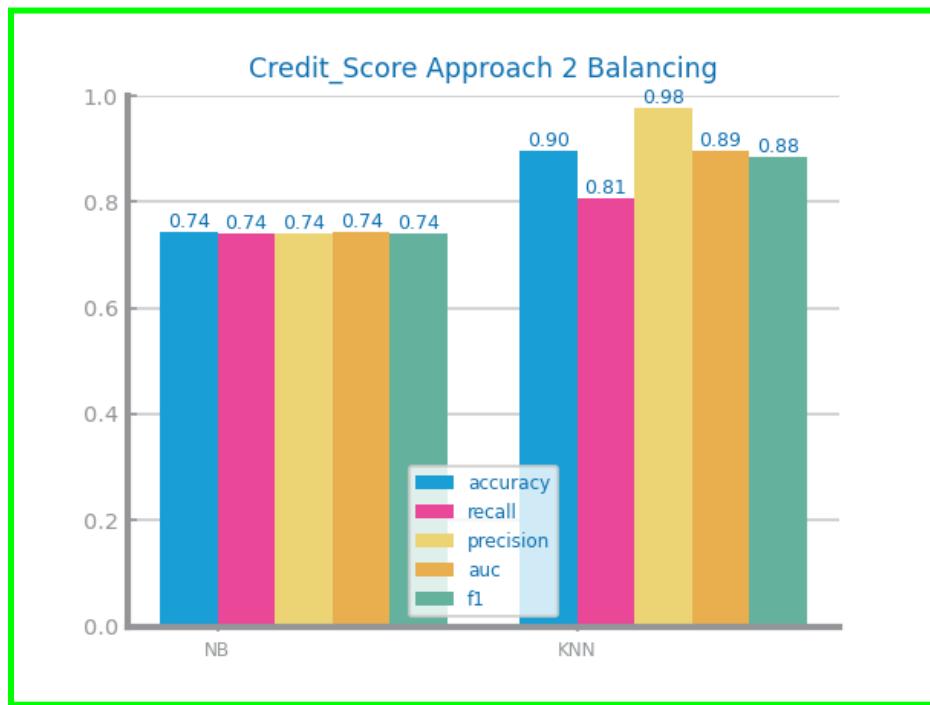
*Figure 25.2 Balancing results with over approach for dataset 1*



*Figure 25.2 Balancing results with SMOTE approach for dataset 1*



*Figure 26 Balancing results with approach 1 (UNDER) for dataset 2*



*Figure 26.1 Balancing results with approach 2 (OVER) for dataset 2*

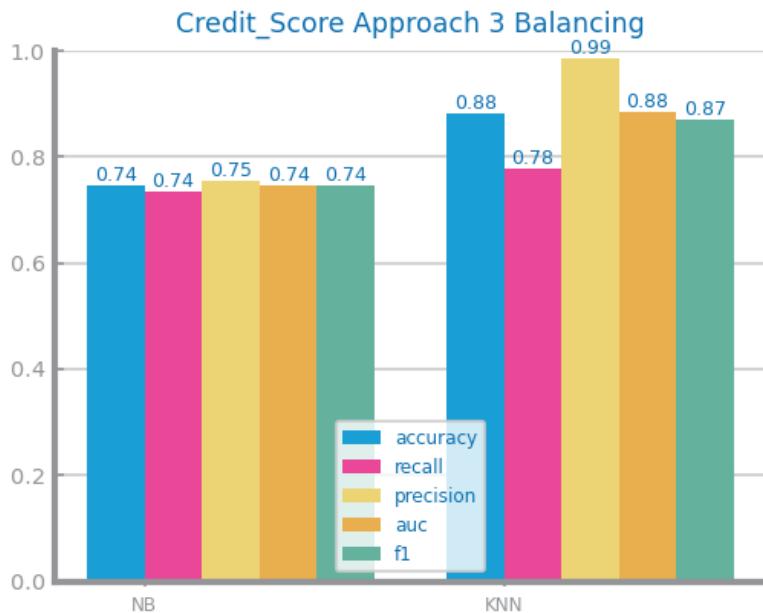


Figure 26.2 Balancing results with approach 3 (SMOTE) for dataset 2

## Feature Selection

### DS1

The redundant feature selection improved the KNN and didn't have a significant impact on the NB, while the relevant approach only dropped the accuracy score so we proceeded with the redundant selection approach on the dataset.

### DS2 - Continue w/ Approach 2

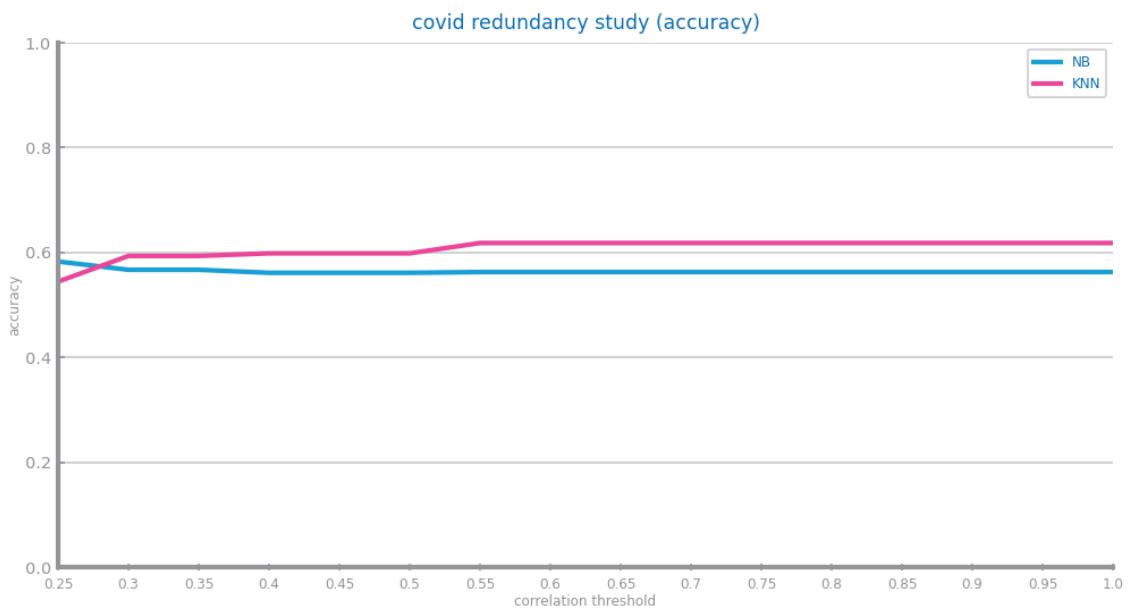
Applied after Balancing

Alternatives:

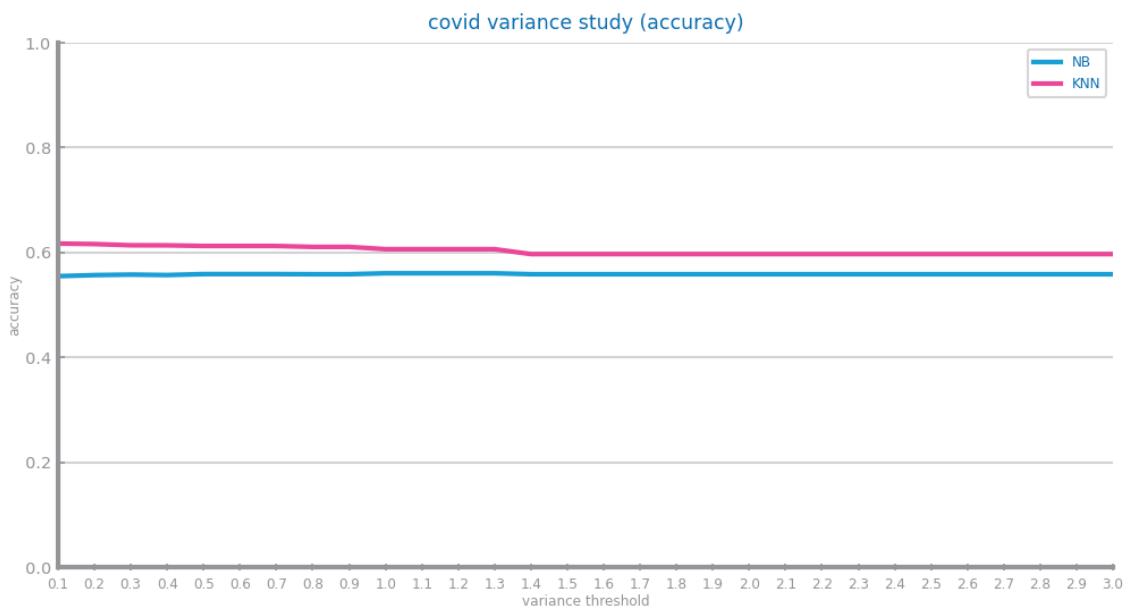
1st Select relevant variables, inconclusive

2nd Drop redundant variables, good

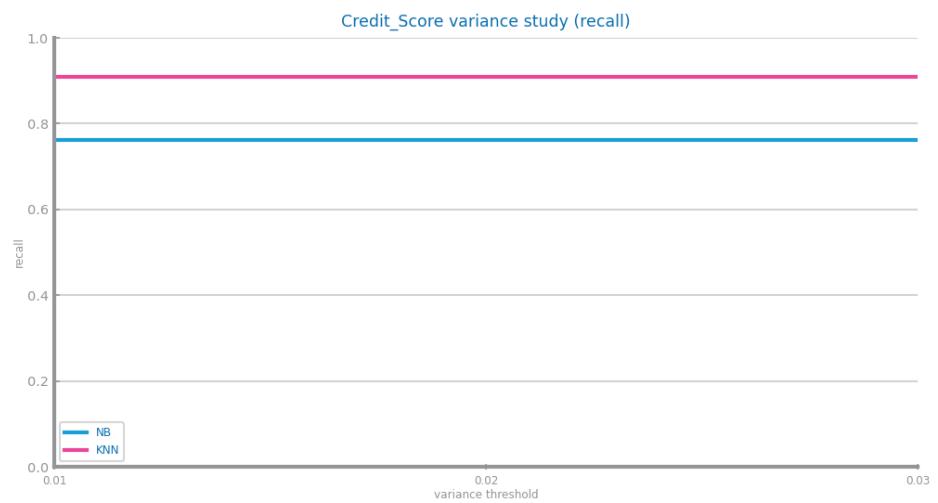
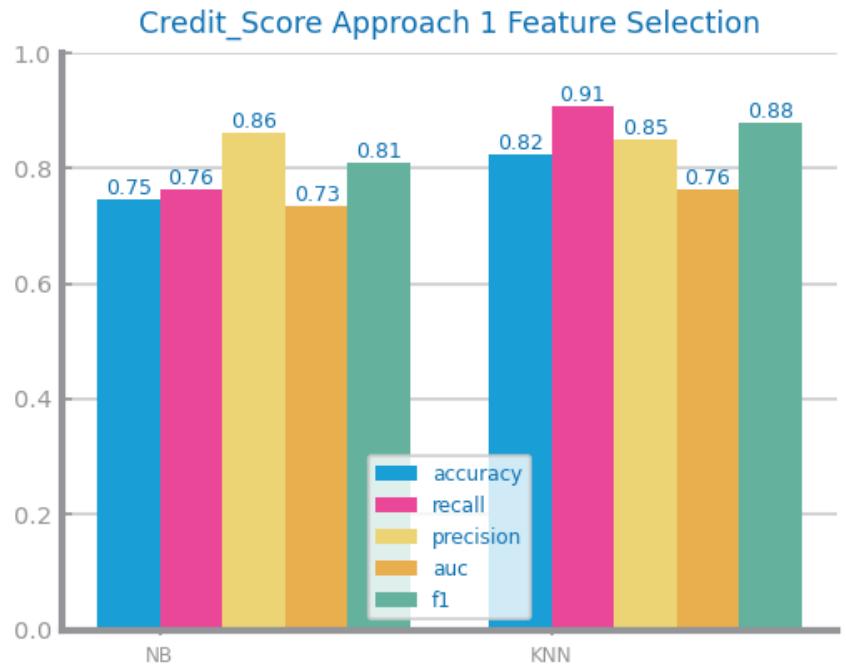
3rd Recursive Feature Elimination, best approach, dropped 6 columns



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

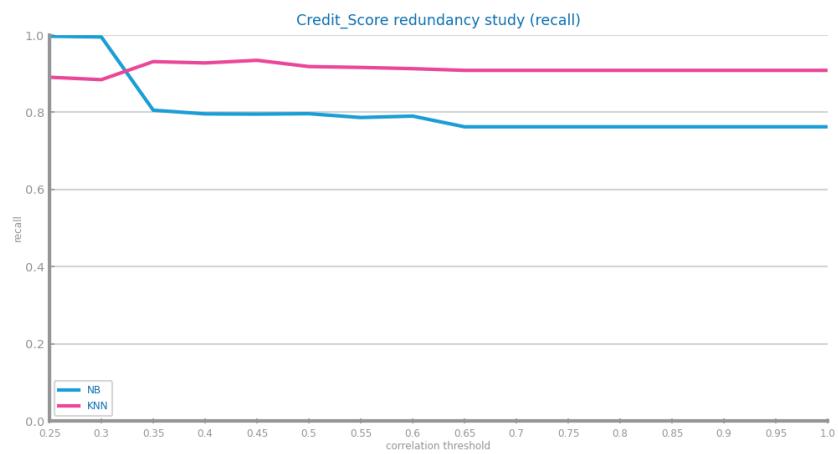
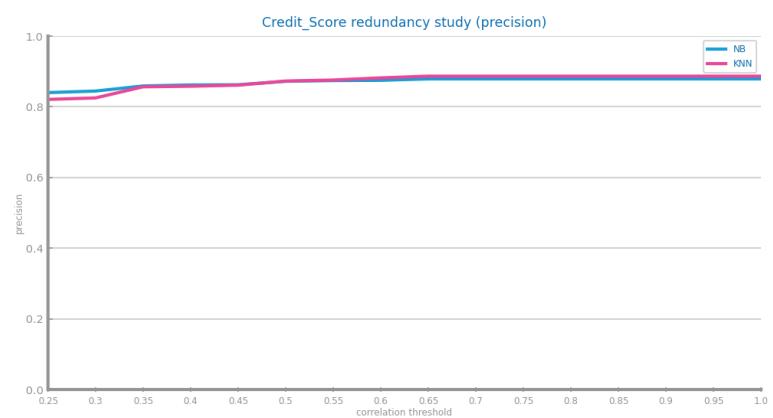
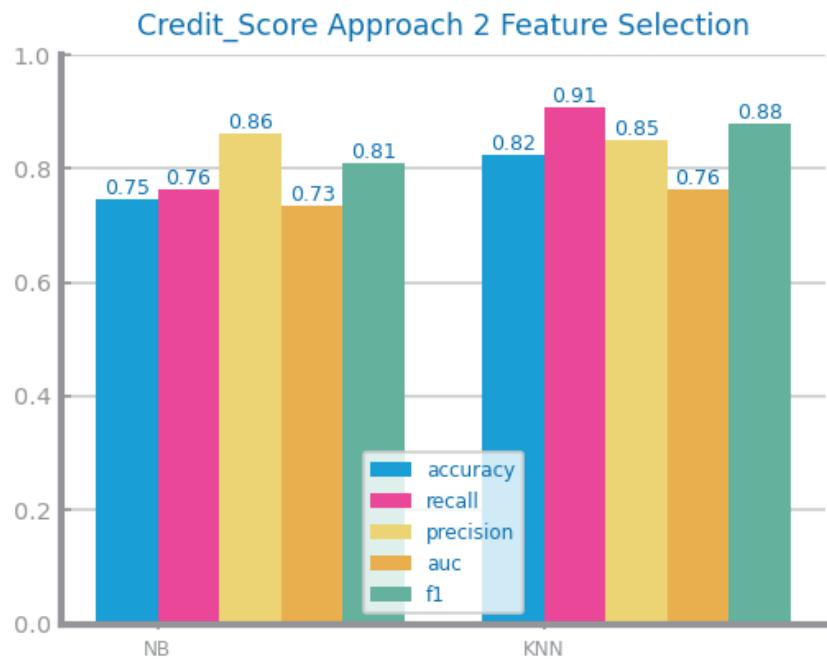


*Figure 27 Feature selection of relevant variables results with different parameters for dataset 1 (variance study)*

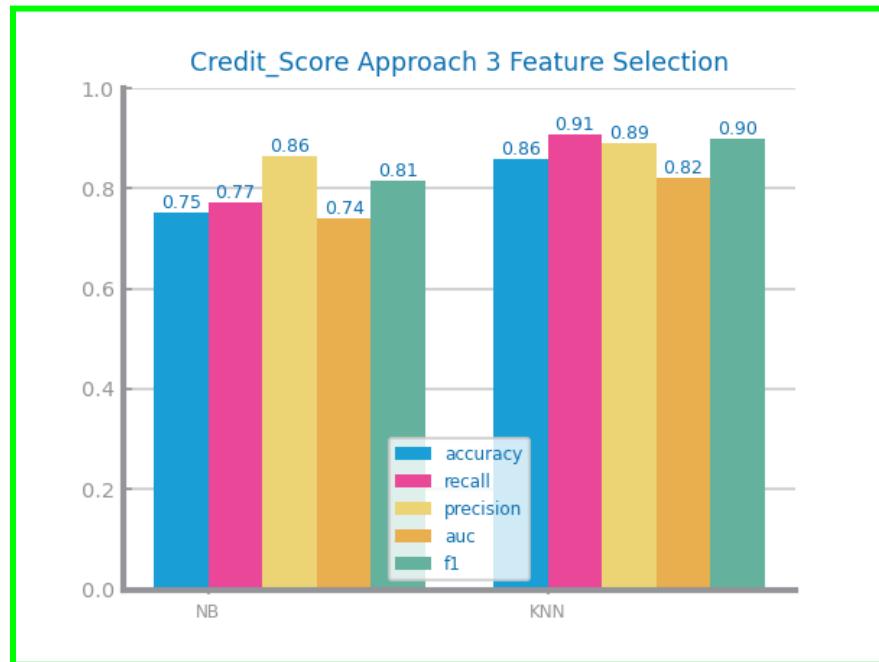
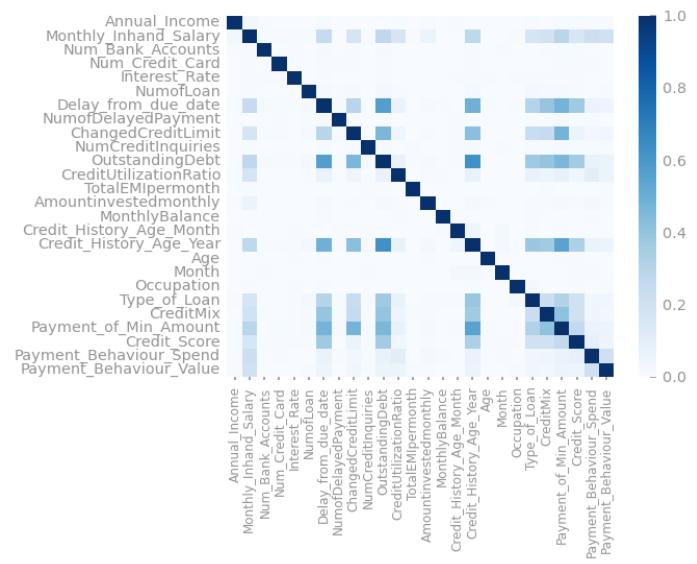


*Figures 27 Feature selection of relevant variables results with different parameters for dataset 2 (variance study)*

(the results are inconclusive because all variables have variance of 1 (z-scaling scaling method))



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



### Additional Feature Generation (optional)

DS	Feature(s)	Description	Formula	Modelling Impact
2	"Name"_Loan	Loan Type Count	$n * a \ (0 \leq n \leq \text{max\_n})$	Loan diversity insight

Figure 31 Feature generation results for dataset 1

Figure 32 Feature generation results for dataset 2

### 3 MODELS' EVALUATION

#### DS1

Hold-out strategy (same data used to compare parameters and models)

Prioritize accuracy (no class is more important)

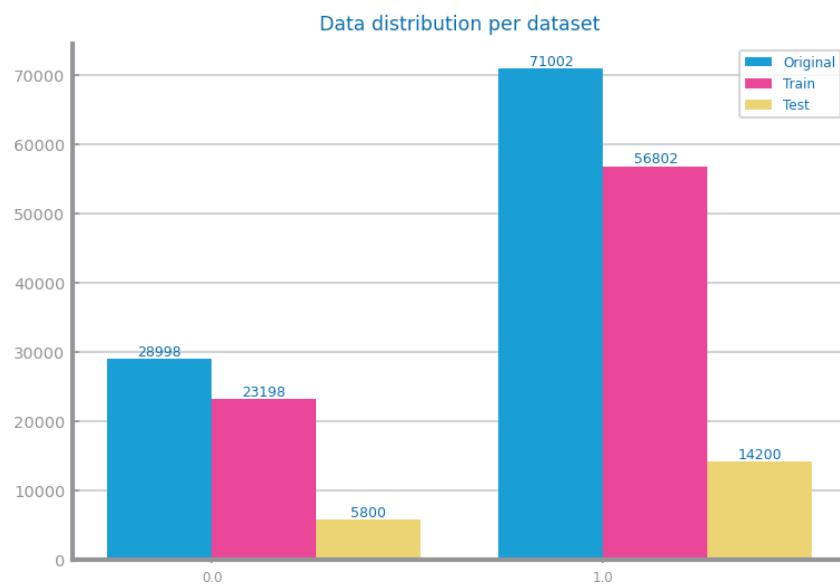
#### DS2

**Train strat** Hold-out for speed, lower cost, size

Balance *FP* & *FN* but prioritize **Precision** (*Willing to accept FN in order to reduce FP*)

Maximize **Precision & Recall**, but try to prioritize **Precision**.

(Cost of a **bad loan** *FP* is very high, so minimize risk of default, even if it means missing out a good customer)



### Naïve Bayes

#### DS1

Multinomial & bernoulli similar & better for accuracy (gaussian more for continuous data)

not good performance: auc=0.57 (similar to random)

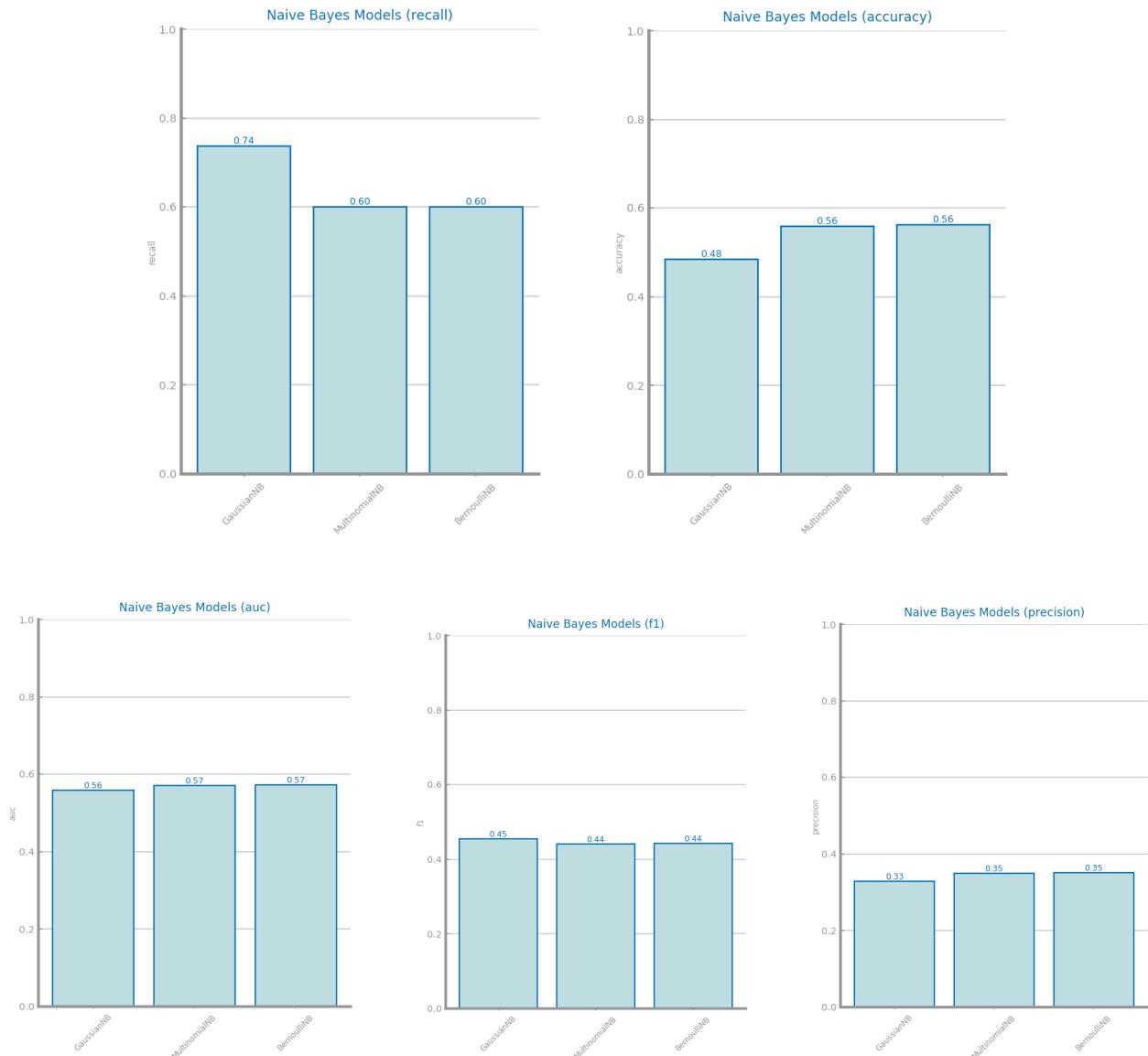
## DS2

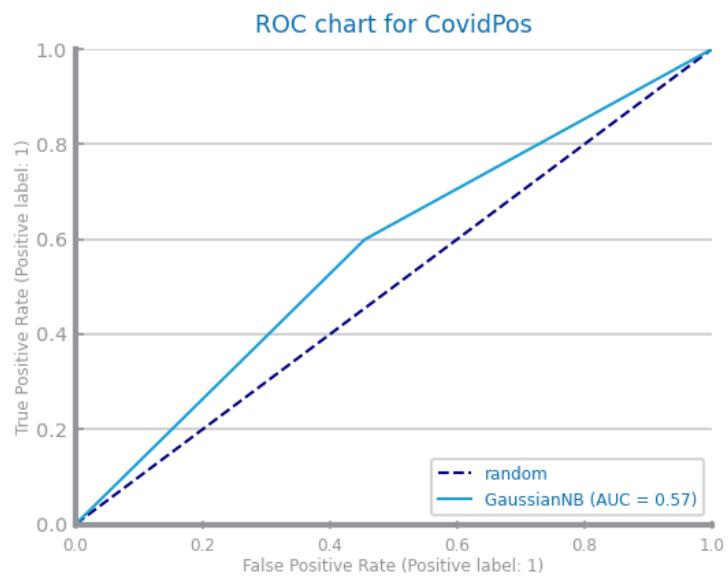
### Comparison baseline

Bernoulli best

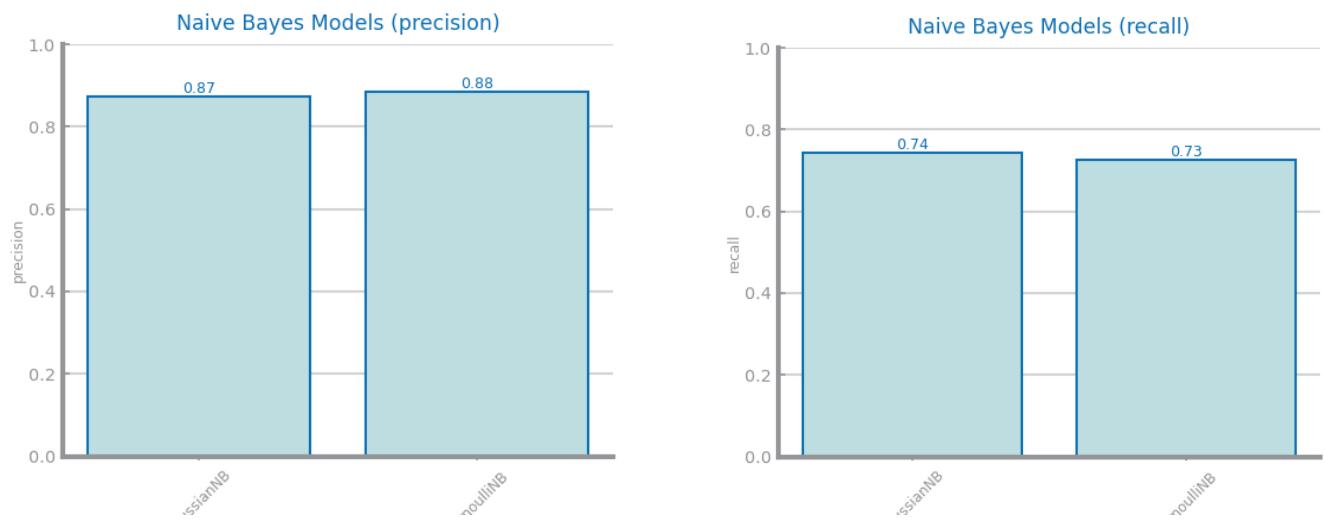
Naive Bayes ignores correlation between variables presented in the correlation matrix

Bad due to simplicity & complex nature of data





*Figure 33 Naïve Bayes alternatives comparison for dataset 1*



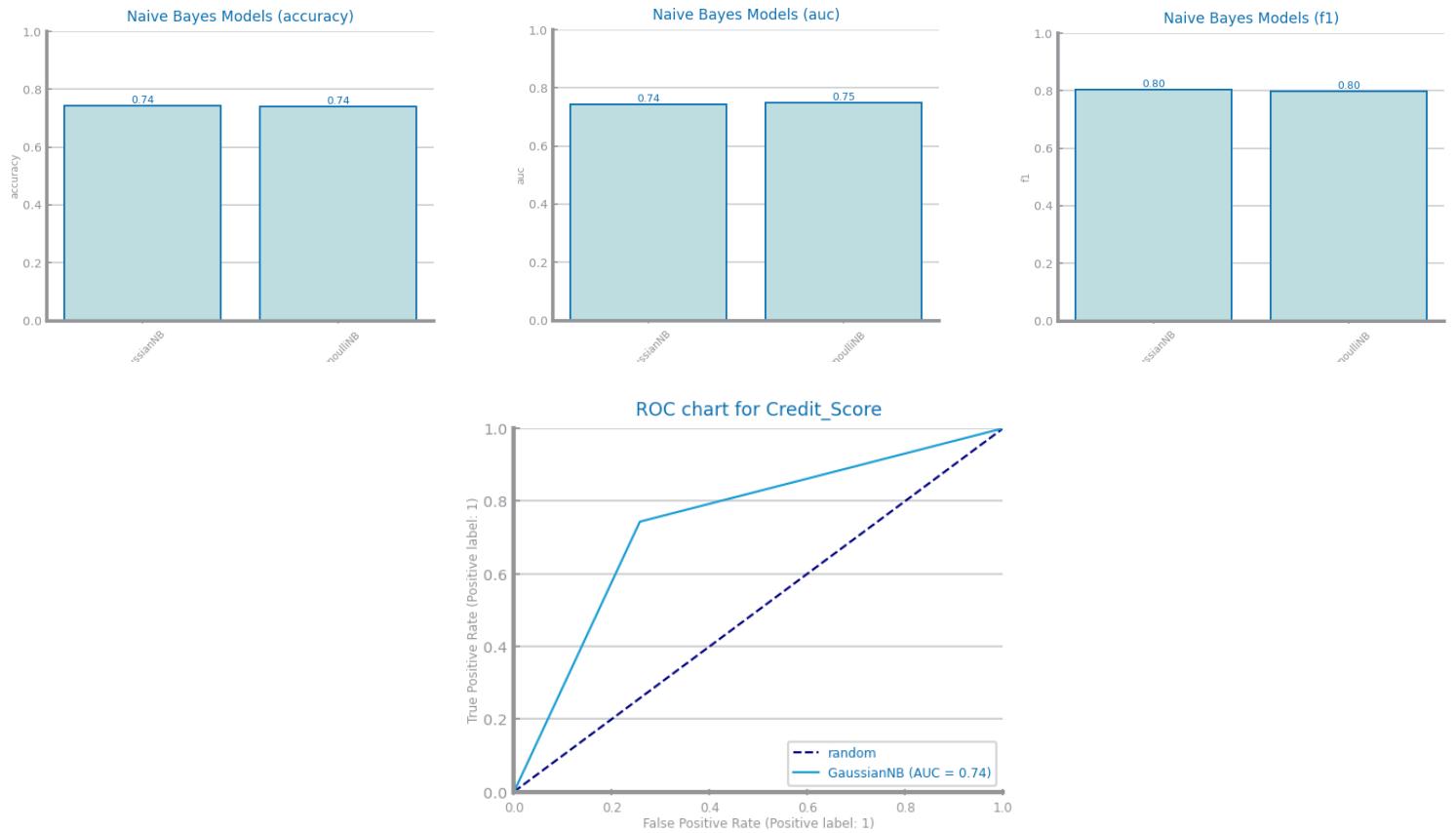


Figure 34 Naive Bayes alternative comparison for dataset 2

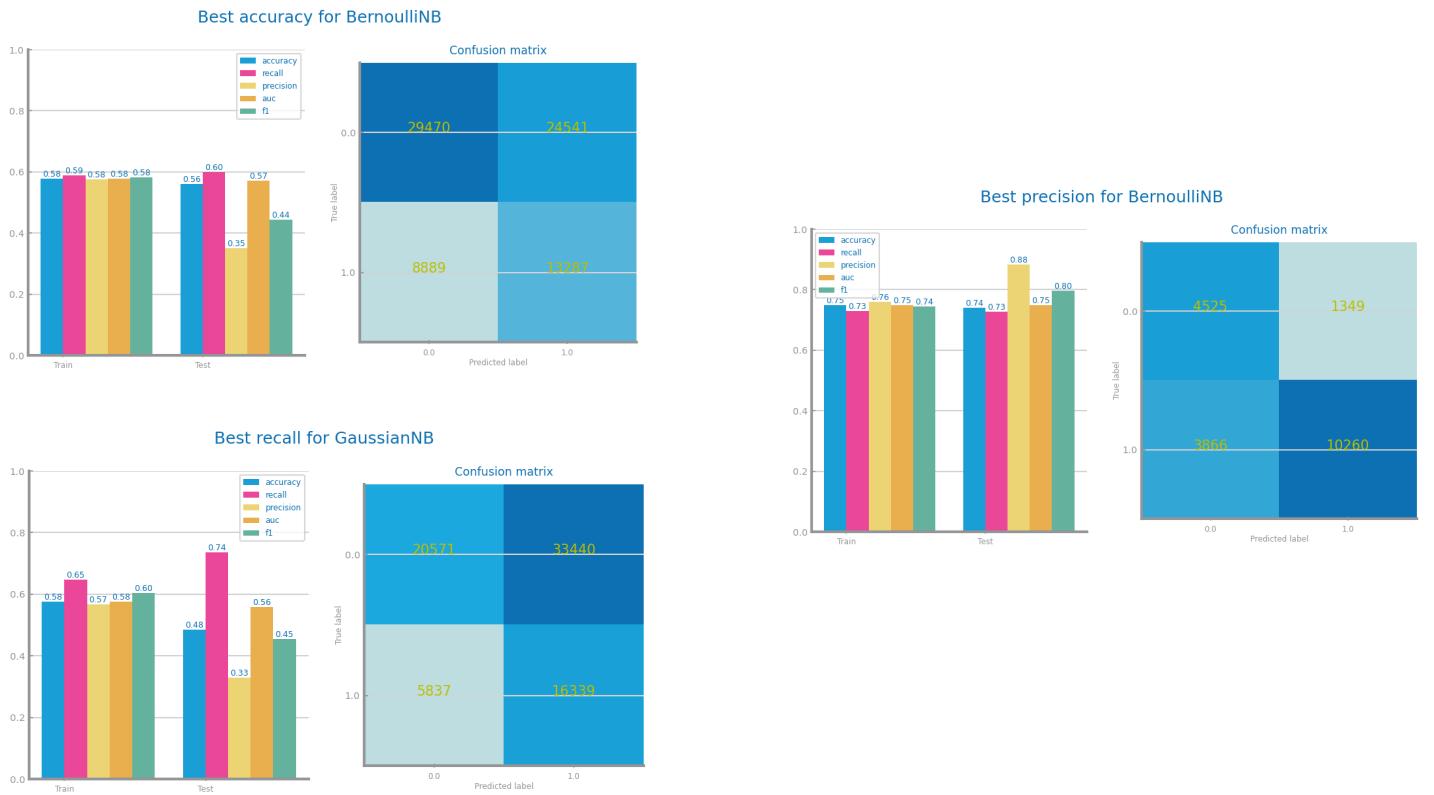


Figure 35 Naïve Bayes best model results for dataset 1 (left) and dataset 2 (right)

## KNN

### DS1

Manhattan & euclidean - best parameters

There is overfitting w/ k=1, however it was the best considered model

### DS2

Test K values from 1 to 1000 (larger range)

KNN best w/ k=5

Manhattan best measure for **all** metrics

For k=1, there is overfitting - train and test have different trends when maximizing *Precision*;

Model conservative about predicting the positive class; more balanced and generalizable, less likely to overfit

Small neighbors number, so model could be over-specialized

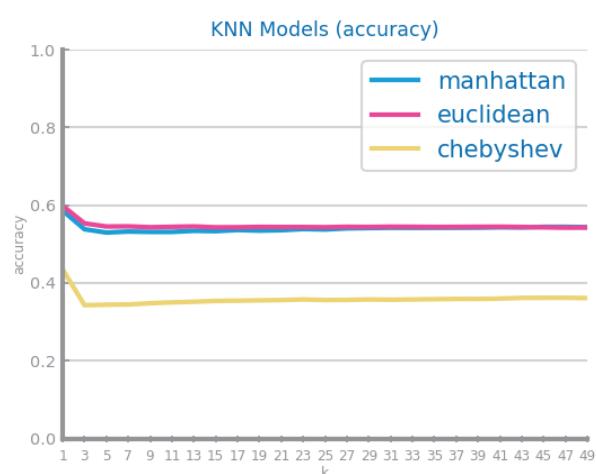


Figure 36 KNN different parameterisations comparison for dataset 1

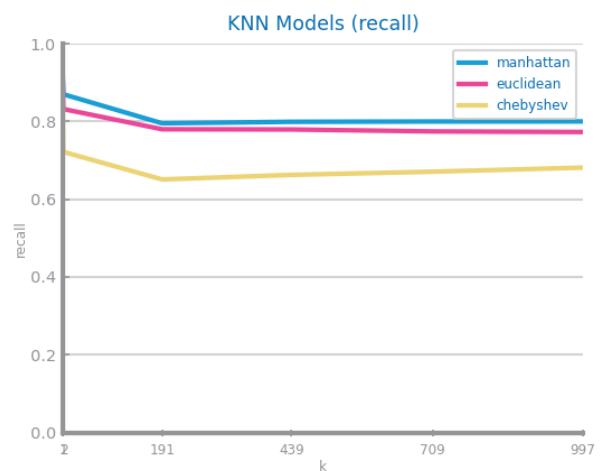
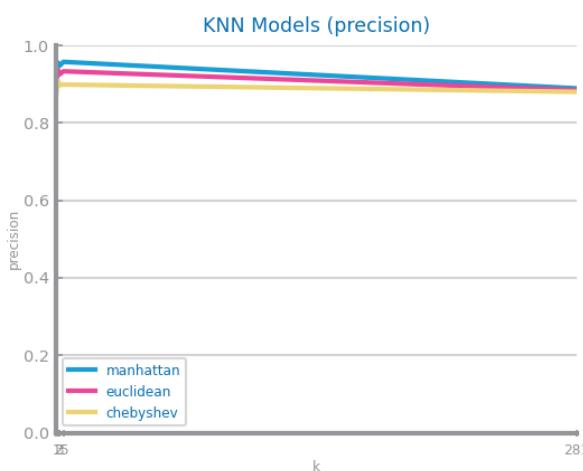


Figure 37 KNN different parameterisations comparison for dataset 2

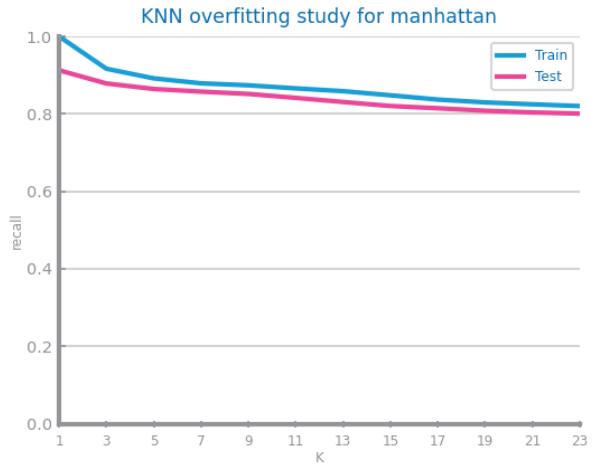
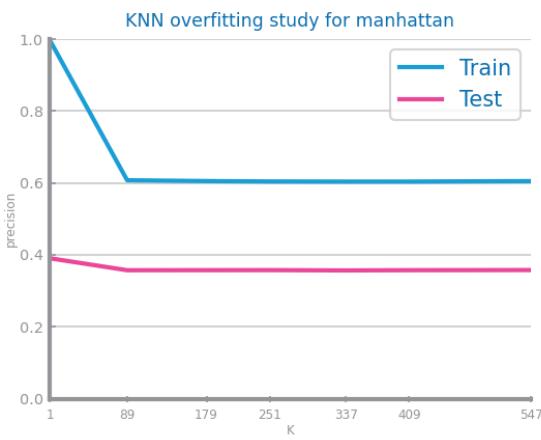
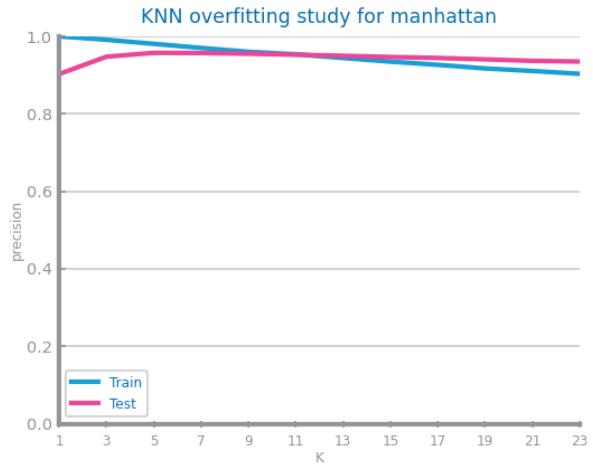
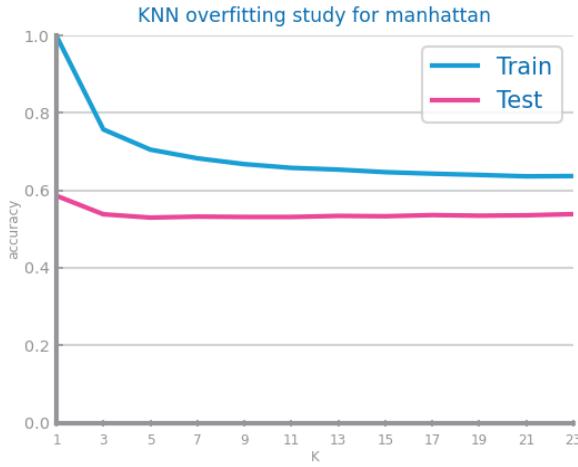


Figure 38 KN

N overfitting analysis for dataset 1 (left) and dataset 2 (right)

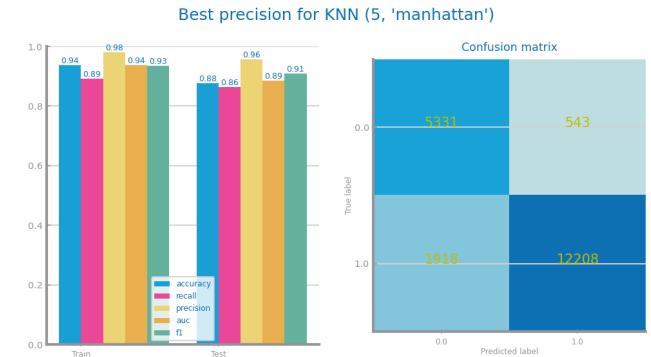
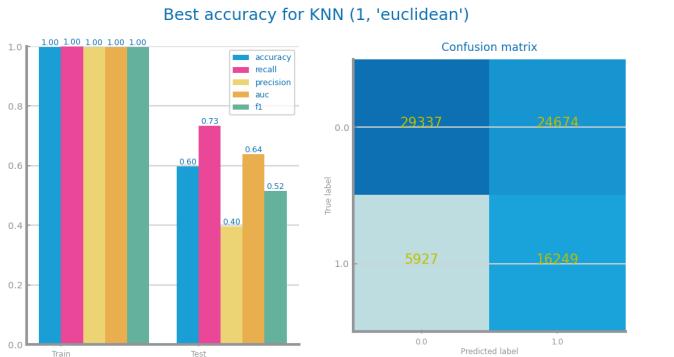


Figure 39 KNN best model results for dataset 1 (left) and dataset 2 (right)

## Decision Trees

### DS1

Model starts overfit in depth=46

Best model .entropy, **depth=40**

after depth 30 model doesn't improve

### DS2

DT best with **gini** and **depth=18**

after +-26 **depth** model doesn't improve

*Overfitting* after **max\_depth** of 6 or 7 where test precision stagnates; this is the value of highest **optimal depth**, gives a trade-off & helps choose the right complexity of the model

*OutstandingDebt* by far & *CreditMix* are most important; have the first as root so its most important discriminator between classes

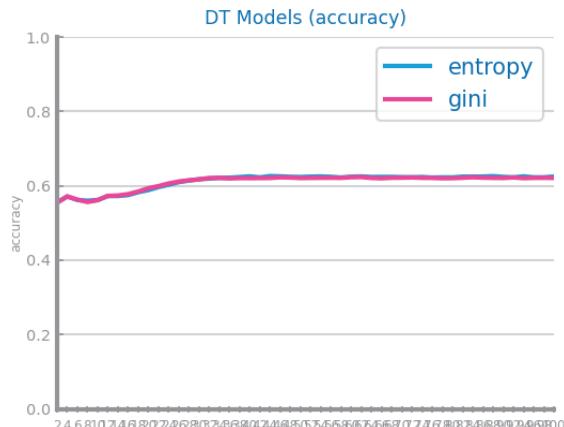
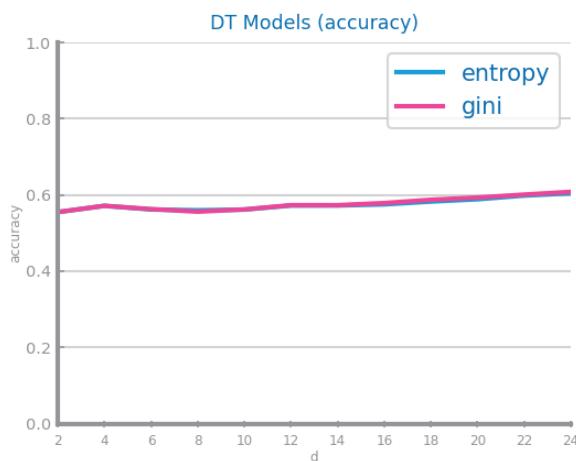


Figure 40 Decision Trees different parameterisations comparison for dataset 1

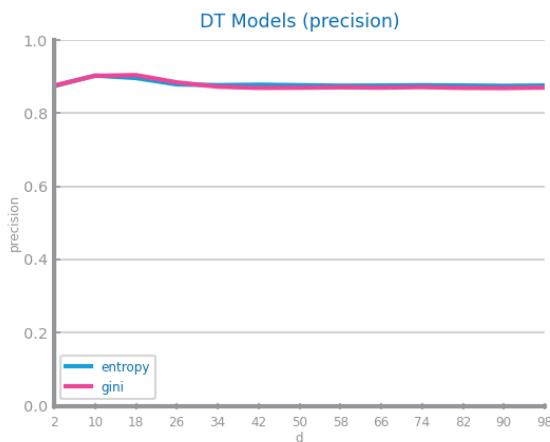
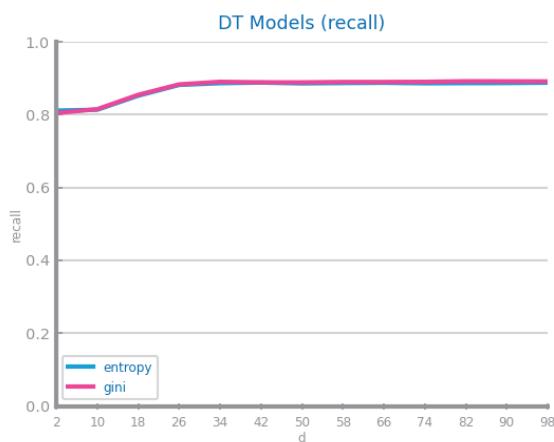


Figure 41 Decision Trees different parameterisations comparison for dataset 2

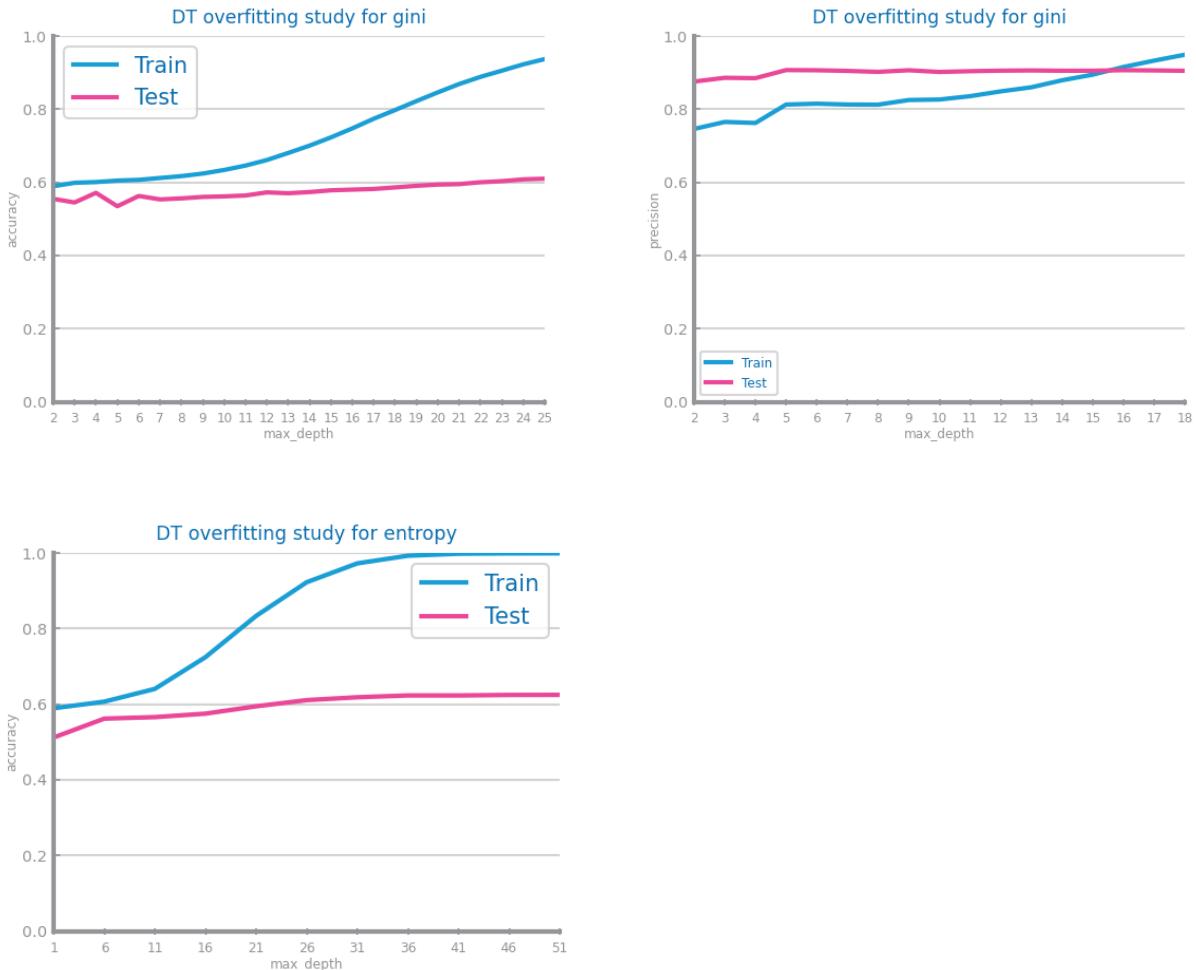


Figure 42 Decision Trees overfitting analysis for dataset 1 (left) and dataset 2 (right)

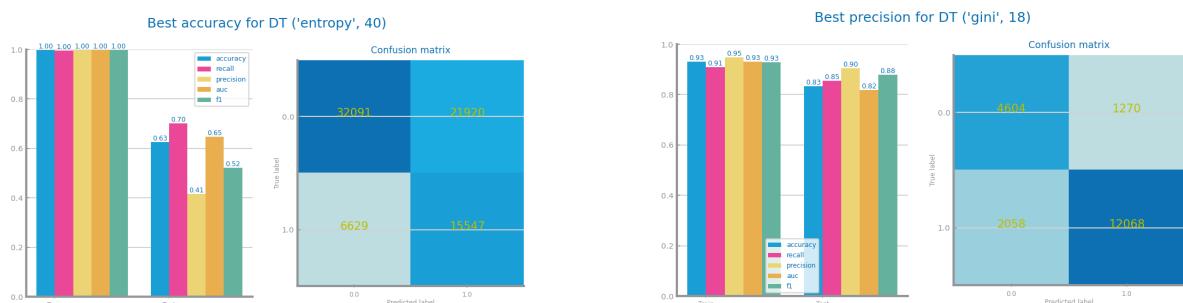
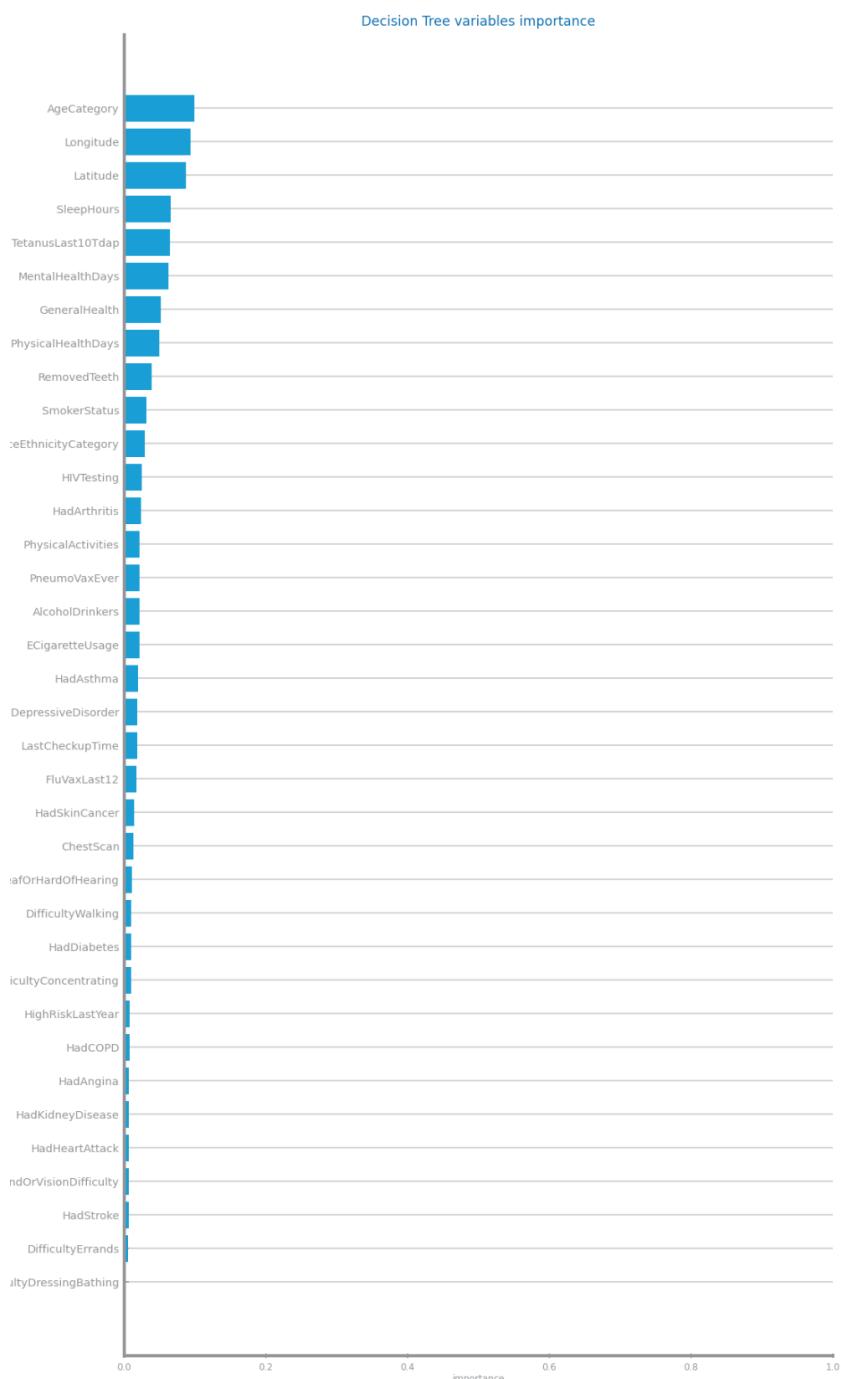


Figure 43 Decision trees best model results for dataset 1 (left) and dataset 2 (right)



*Figure 44.1 Variables importance for dataset 1*

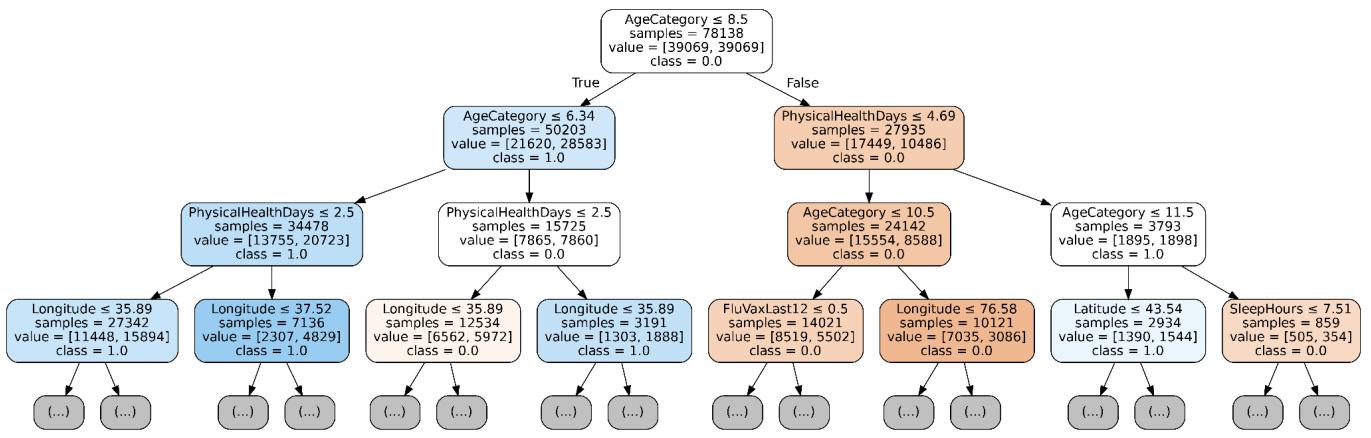


Figure 44.2 Best tree for dataset 1

### Decision Tree variables importance

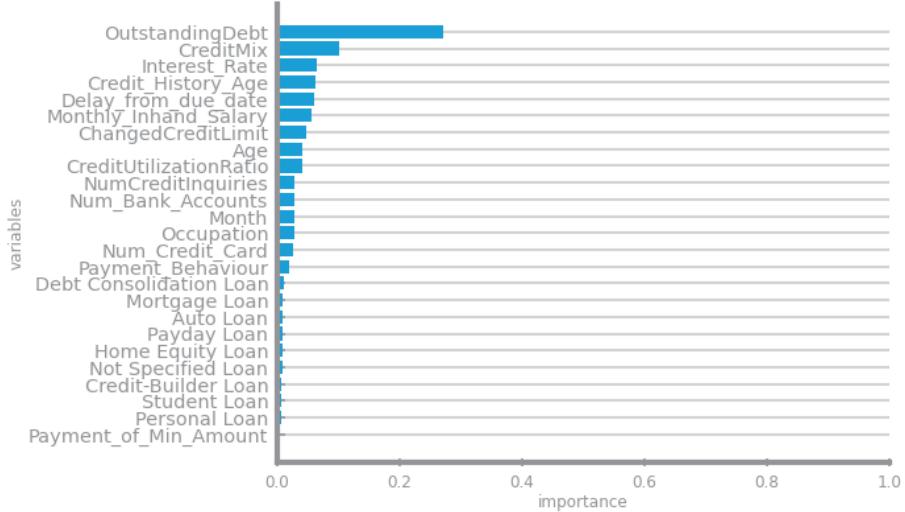


Figure 45.1 Variables importance for dataset 2

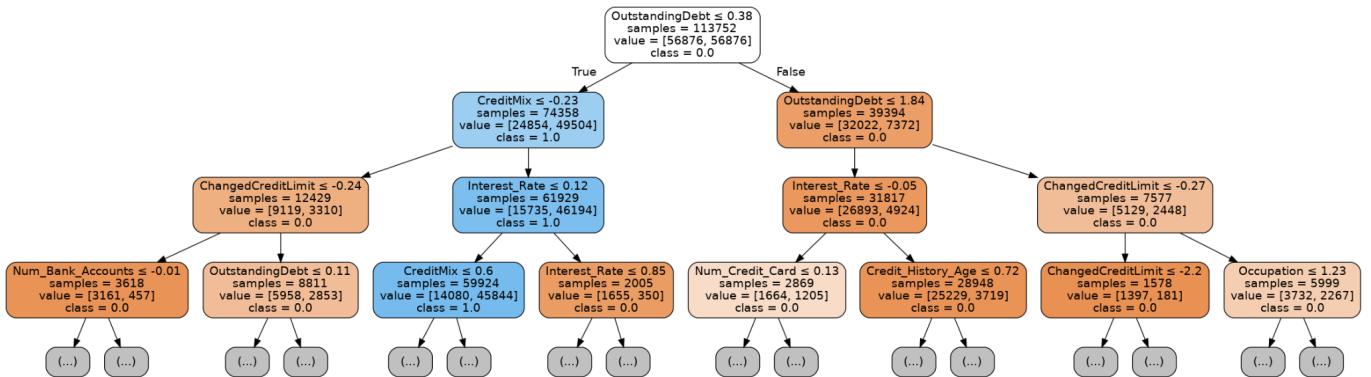


Figure 45.1 Best trees for dataset 2

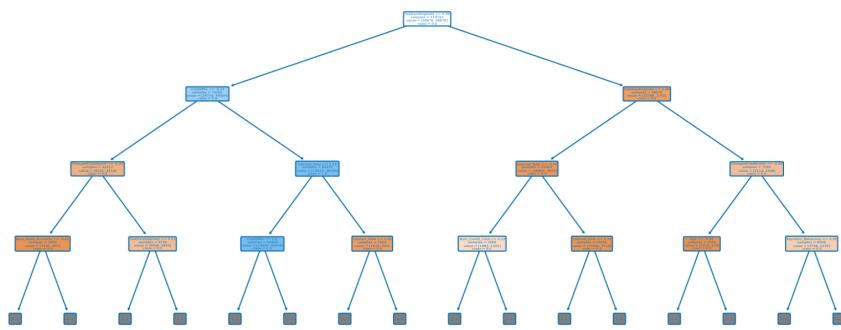


Figure 45.2 Best trees simplified for dataset 2

## Random Forests

### DS1

No big differences in the parameters (fig46) & No overfitting observed

Most important var AgeCategory

### DS2

RF best with 100 trees ( $d=7$ ,  $f=0.3$ )

Key variables in model *OutstandingDebt*, *Interest\_Rate* & *CreditMix*; small stdev means the importance of these features are consistent across different trees; some variables have more variability

Precision shows less variance in vars ranking compared to recall

There is no overfitting, since the *Precision* over both datasets almost doesn't change (expected)

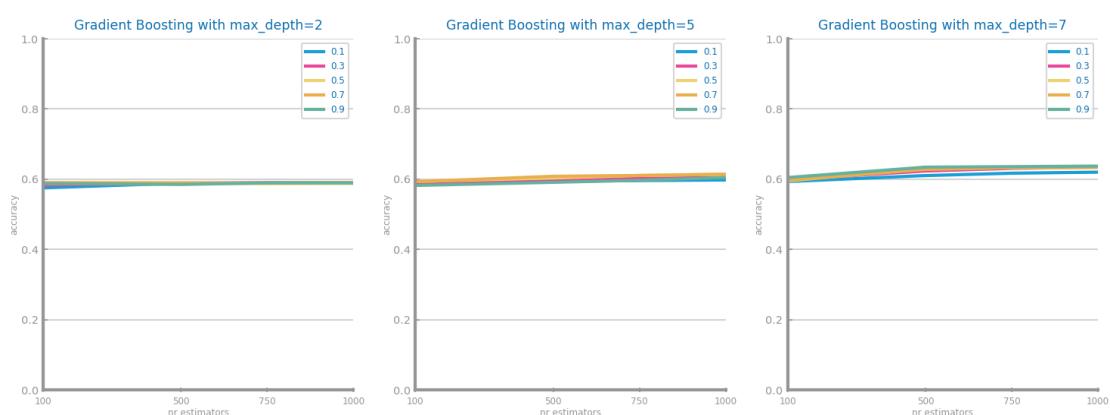


Figure 46 Random Forests different parameterisations comparison for dataset 1

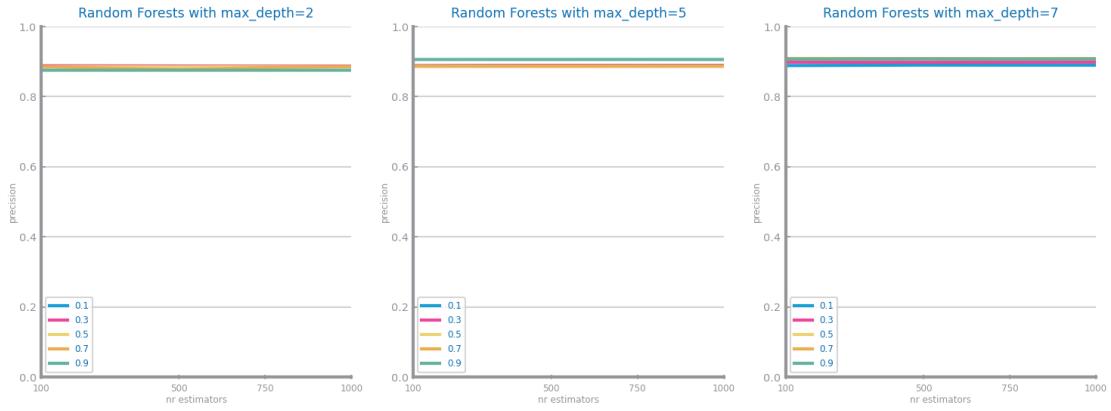


Figure 47 Random Forests different parameterisations comparison for dataset 2

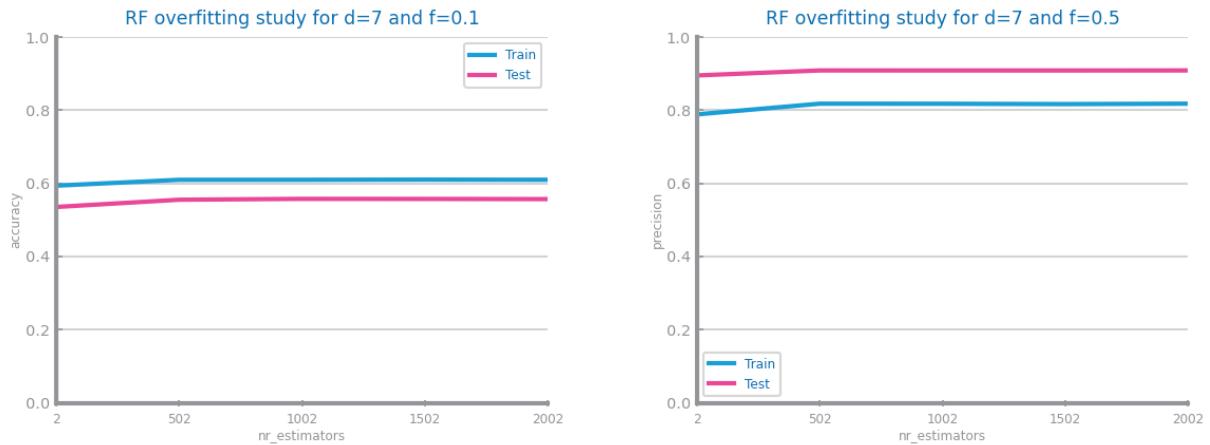


Figure 48 Random Forests overfitting analysis for dataset 1 (left) and dataset 2 (right)

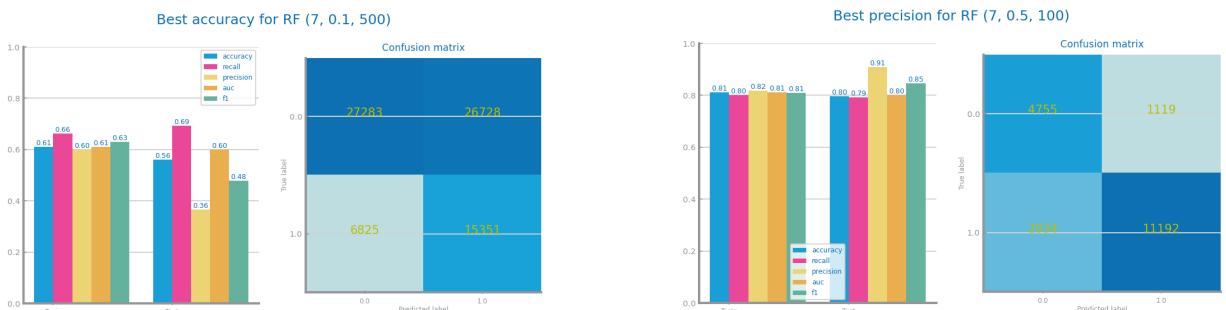


Figure 49 Random Forests best model results for dataset 1 (left) and dataset 2 (right)

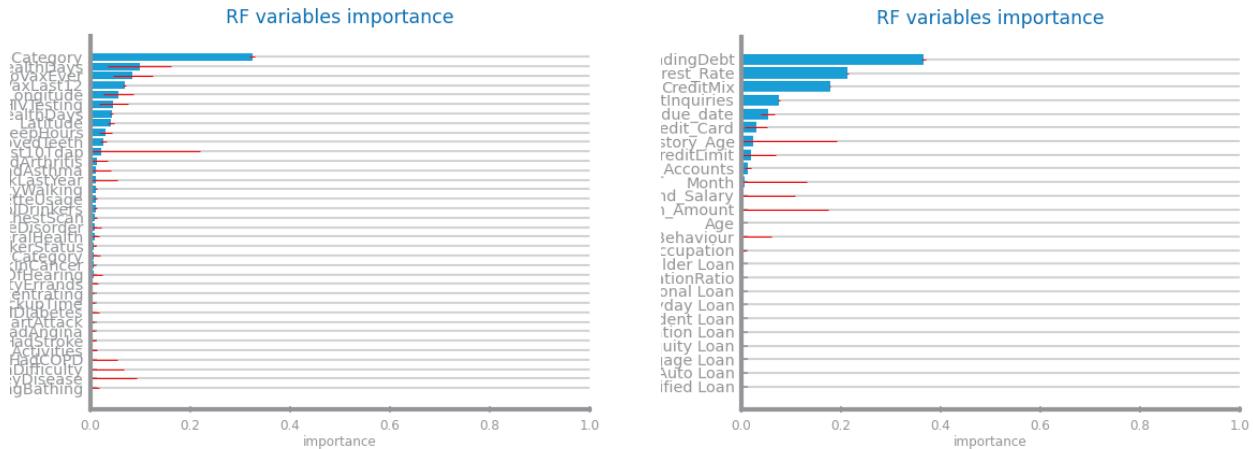


Figure 50 Random Forests variables importance for dataset 1 (left) and dataset 2 (right)

## Gradient Boosting

### DS1

From 502 estimators the model is overfitting (fig55)

Most important vars AgeCategory, Lat & long

### DS2

GB best with 1000 trees ( $d=7$ ,  $lr=0.3$ ). Precision is consistent across depths and learning rates. Recall improves with depth and rate, while some overfitting is detected. Key variables: OutstandingDebt, CreditMix, Delay\_from\_due\_date, with the latter showing inconsistent importance.

Figure 51 Gradient boosting different parameterisations comparison for dataset 1

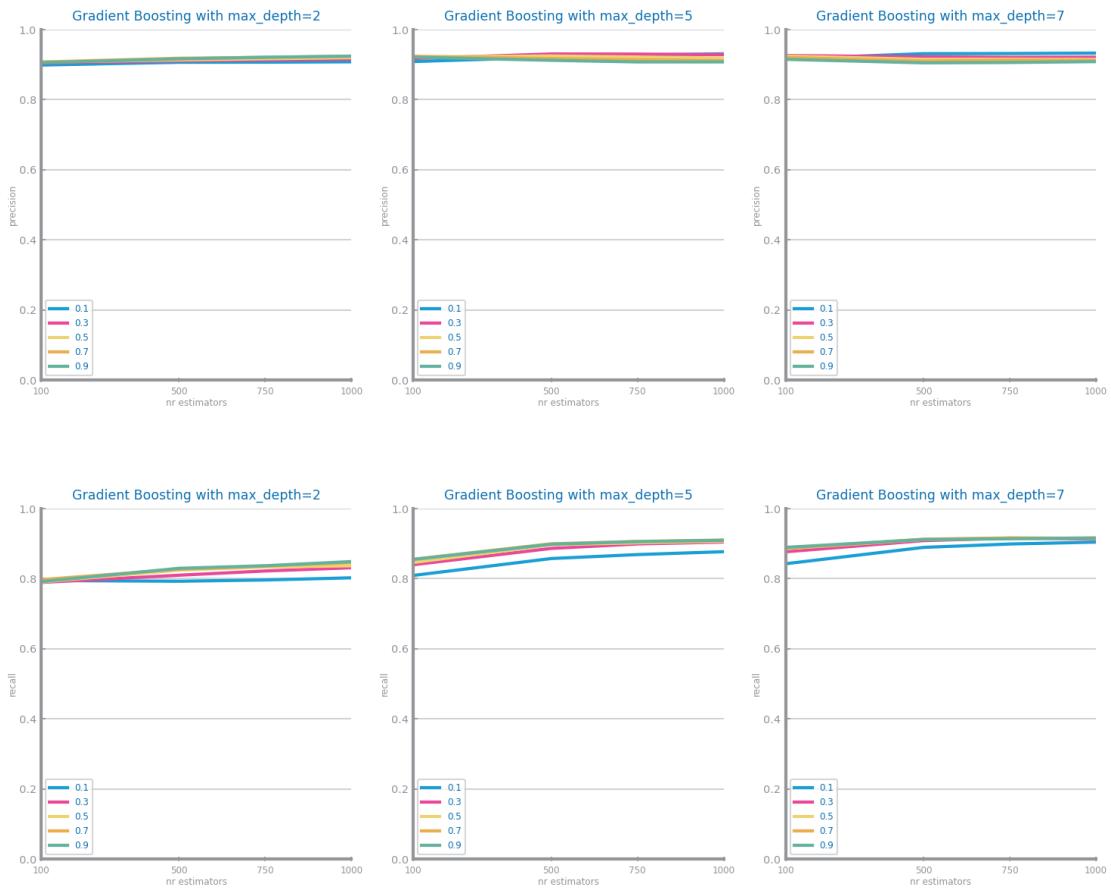


Figure 52 Gradient boosting different parameterisations comparison for dataset 2

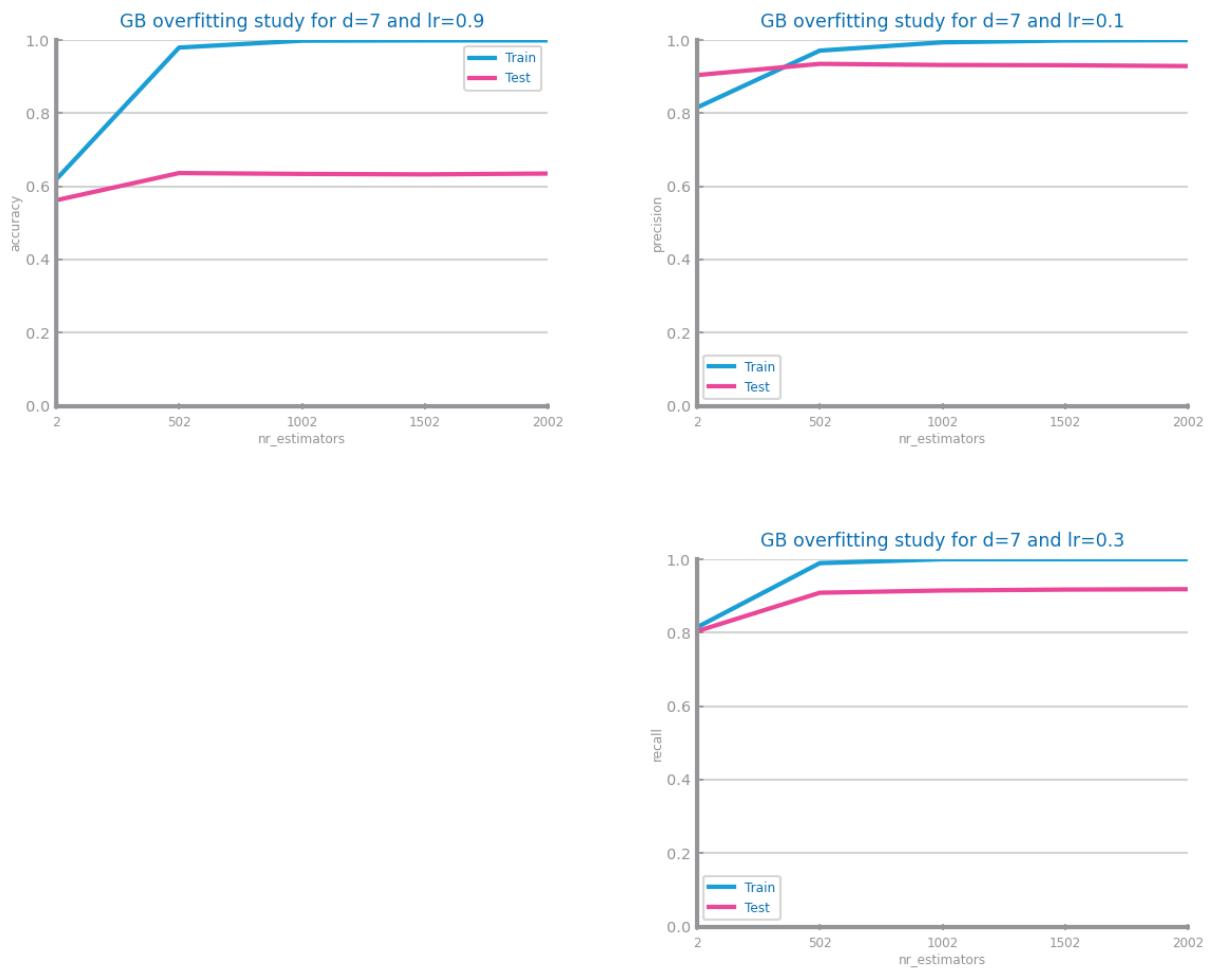


Figure 53 Gradient boosting overfitting analysis for dataset 1 (left) and dataset 2 (right)

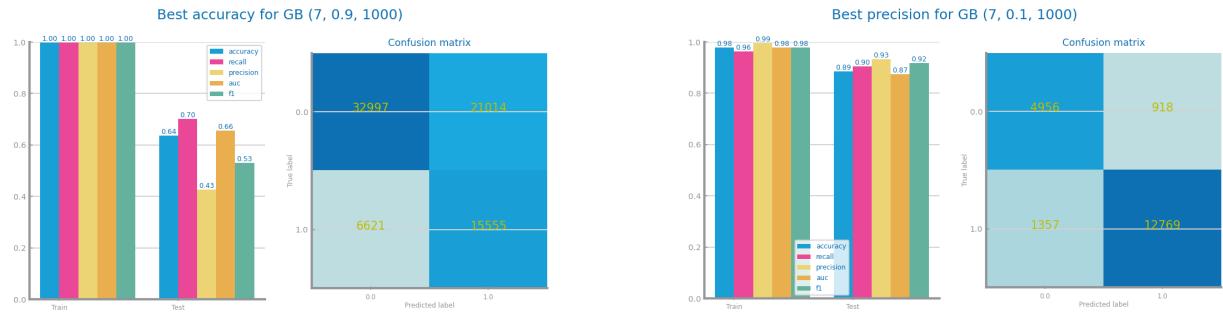


Figure 54 Gradient boosting best model results for dataset 1 (left) and dataset 2 (right)

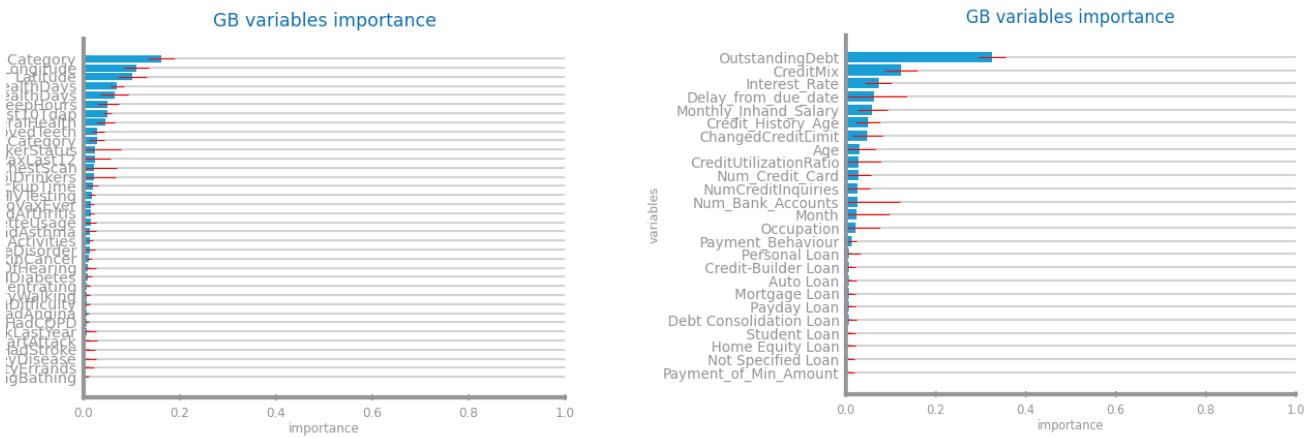


Figure 55 Gradient boosting variables importance for dataset 1 (left) and dataset 2 (right)

## Multi-Layer Perceptrons

### DS1

Irregular overfitting char due to randomness of the training of MLP (fig58)

Accuracy isn't the best measure to optimize in MLP. Everything is considered negative (fig59)

### DS2

Model's robustness is evident with a consistent 0.9 precision across all learning rate types. Adaptive parameterization ensures stable recall, avoiding overfitting and local minimums. No overfitting is detected from the corresponding study.

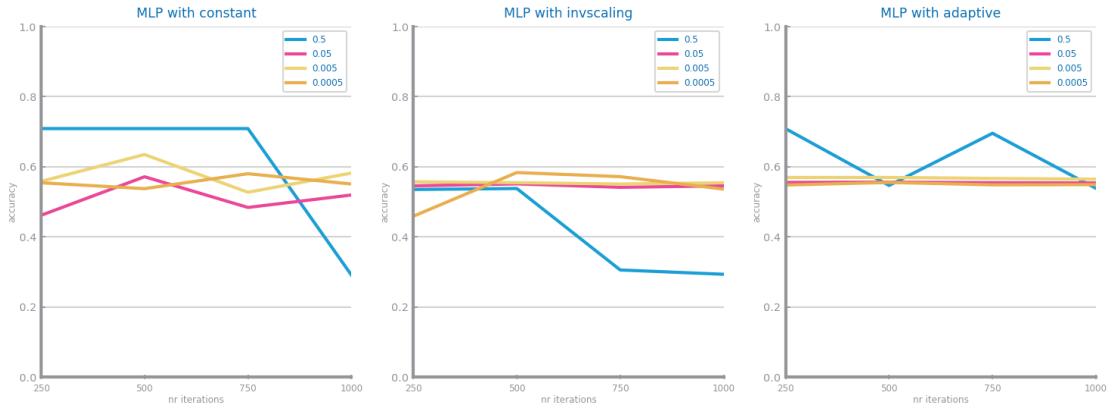


Figure 56 MLP different parameterisations comparison for dataset 1

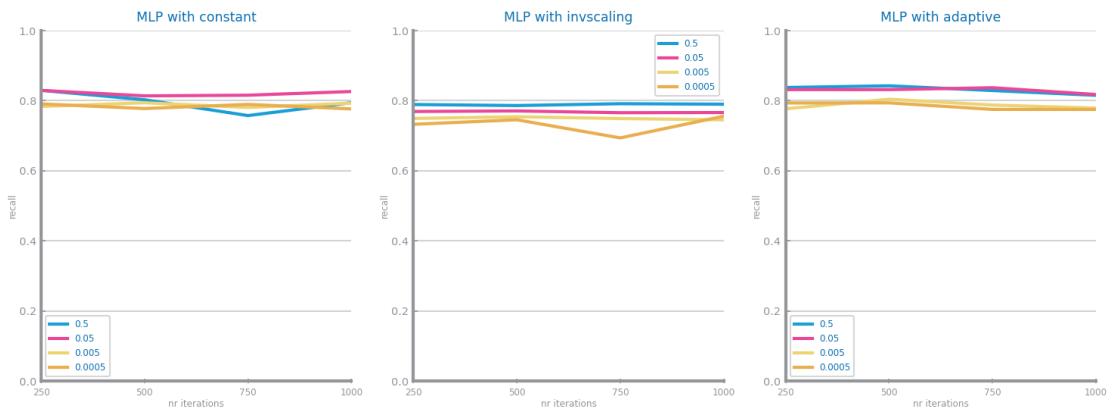
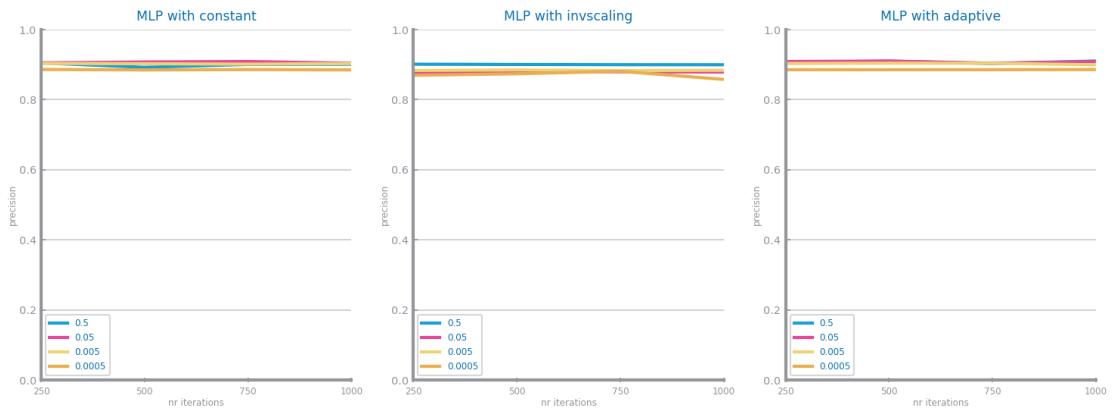


Figure 57 MLP different parameterisations comparison for dataset 2

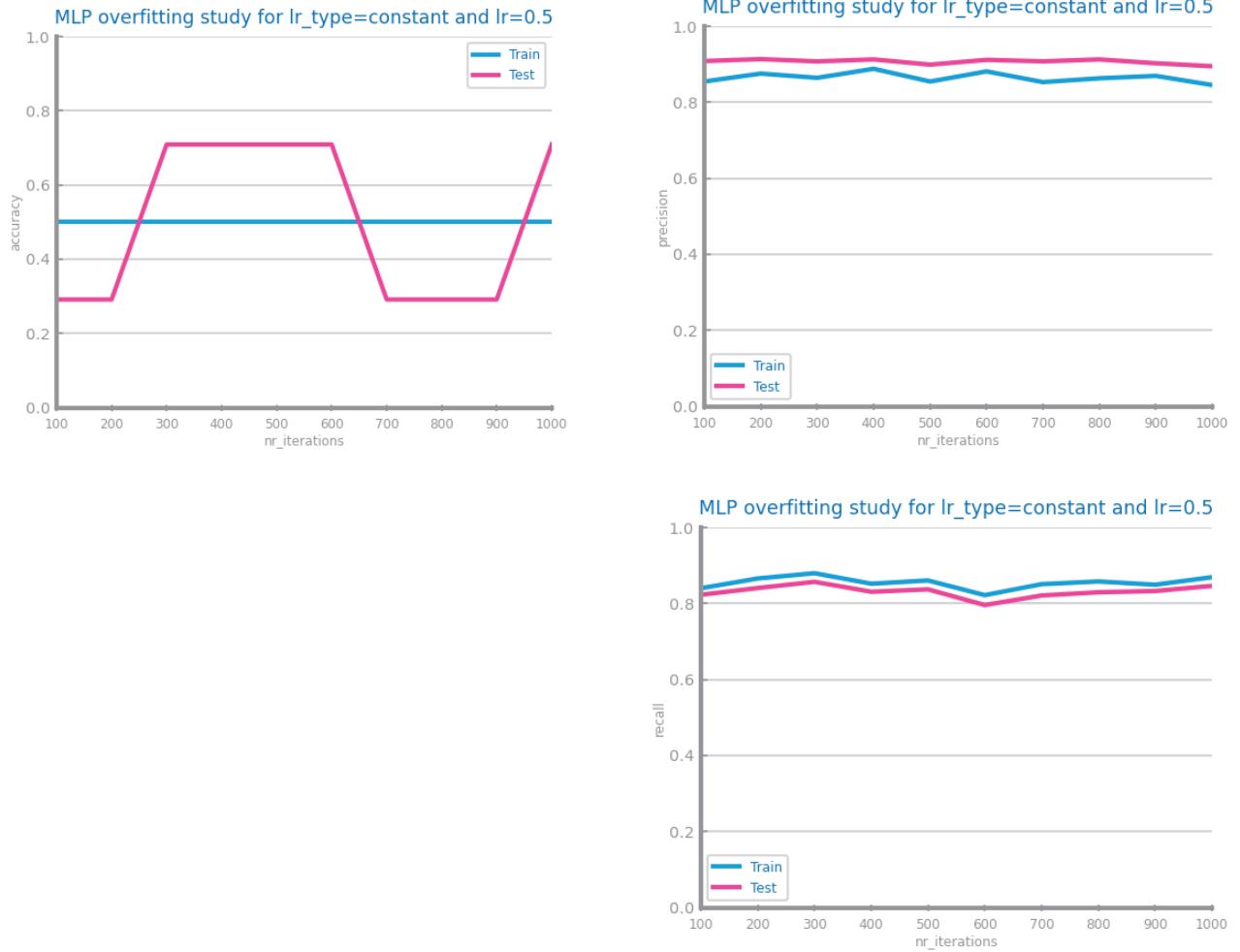


Figure 58 MLP overfitting analysis for dataset 1 (left) and dataset 2 (right)

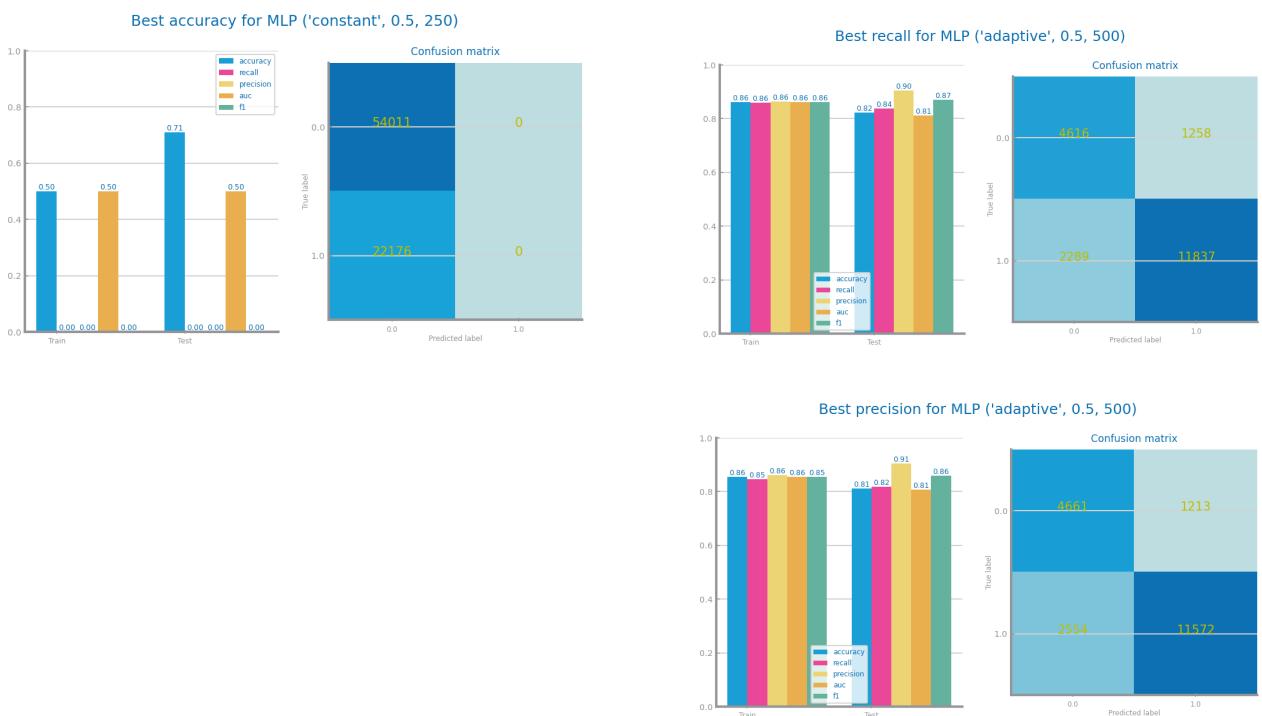


Figure 59 MLP best model results for dataset 1 (left) and dataset 2 (right)

## 4 CRITICAL ANALYSIS

### DS1

AgeCategory turned out to be the most important var followed by latitude and longitude for most models. Would probably be interesting to have more granularity like the exact ages and cities, to understand if it is possible to get better results.

Lots of models entered overfitting, to solve this we could try to use less data on training set, changing training strategy to cross-validation or even try to better fit hyperparameters.

With the results obtained tree models seemed a better choice.

### DS2

Primarily aiming for accurate credit classification, we decided to maximize the precision of our models to reduce the number of False Negatives (which equates to giving a loan to someone with a bad credit score). During the encoding step, we encoded diverse loan types so the models could better distinguish between these. Scaling played a vital role in ensuring equitable feature impact across all models (MLP, DT, RF), preventing dominance of a singular feature. Outlier treatment was critical for MLP due to sensitivity, and had limited impact on DT, while RF's stability relied on ensemble averaging. Balancing classes significantly improved fairness in predictions for MLP and DT, maintaining robustness. These steps collectively optimized models for higher precision and recall, effectively reducing overfitting in the credit score classification process.

We consider MLP and Random Forests among the best models for credit scoring. MLP's structure enables learning intricate, nonlinear patterns via backpropagation and nonlinear activations, ideal for capturing complex data relationships. Simultaneously, Random Forests' ensemble nature offers robustness, reducing overfitting by aggregating decisions from multiple trees, ensuring more reliable predictions in credit scoring tasks.

# TIME SERIES FORECASTING

## 5 DATA PROFILING

### ***Data Dimensionality and Granularity***

#### **DS1**

**Atomic** is weekly.

There is an upward trend and it's cumulative.

#### **DS2**

**Goal** give insights for planning & congestion management

**Atomic granularity** 15-min intervals

Other granularities, Hourly & 4Hours

"Sum" aggregation function

**No Trend**, stable over the months

**Daily Seasonality** - morning & evening rush (*affected by time of the day - fixed & known period*)

**Weekly Cycle** - Peak intervals suggesting regular weekend increases and from 1st to 2nd Peak an occasional 10-days gap (*reflects anomaly or special event*)

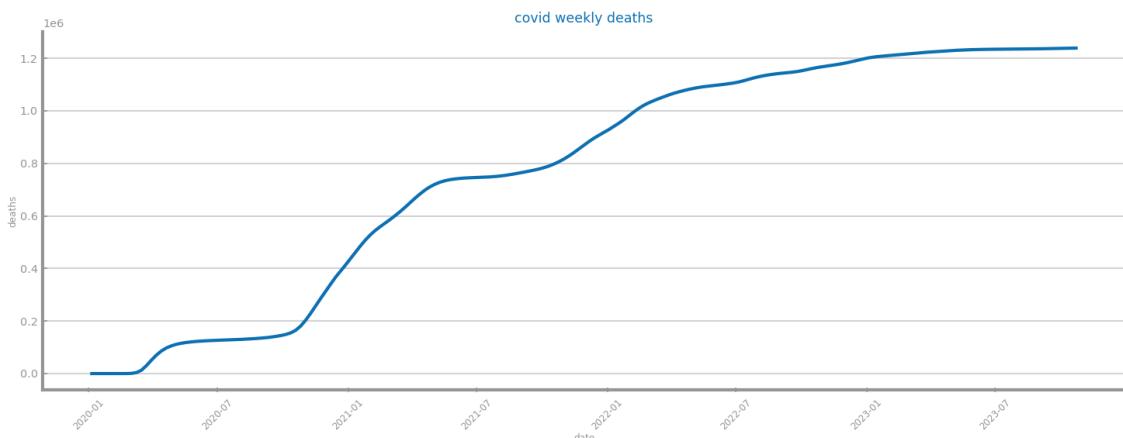


Figure 60 Time series 1 at the most granular detail

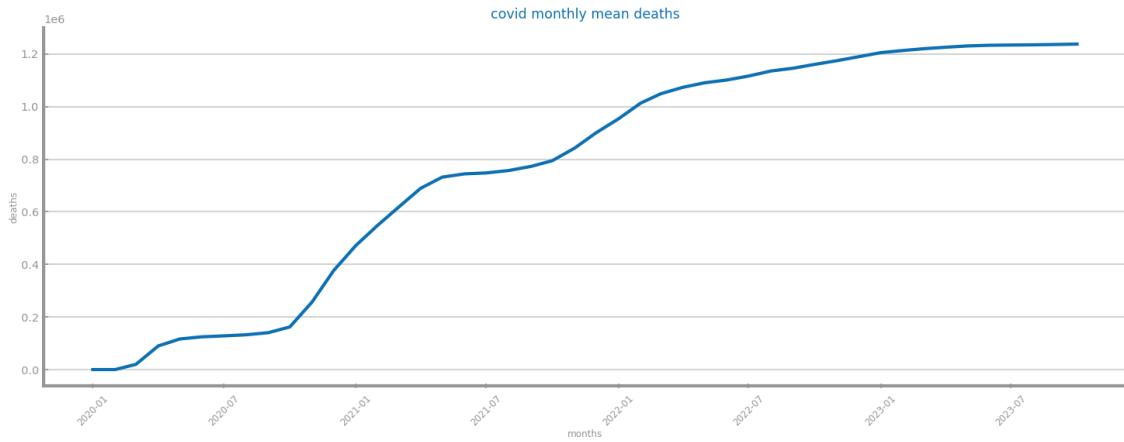


Figure 61 Time series 1 at the second chosen granularity

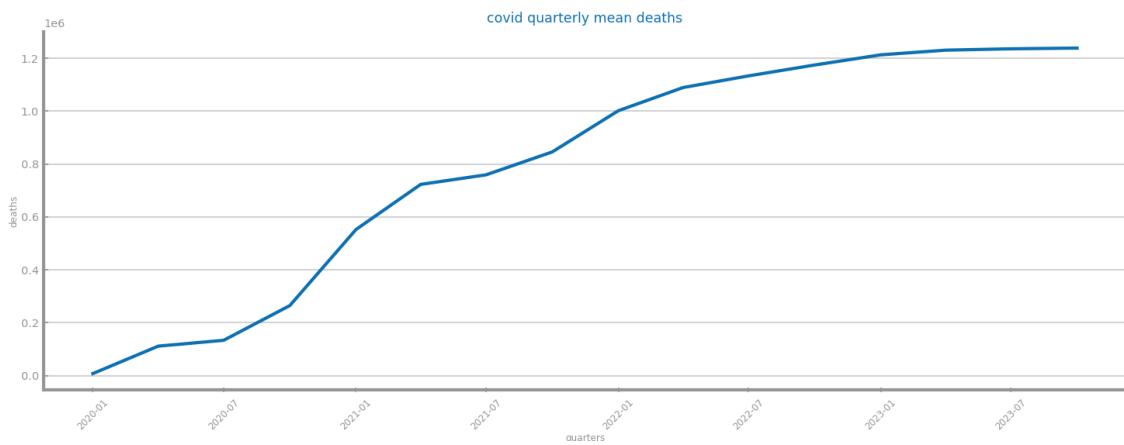


Figure 62 Time series 1 at the third chosen granularity

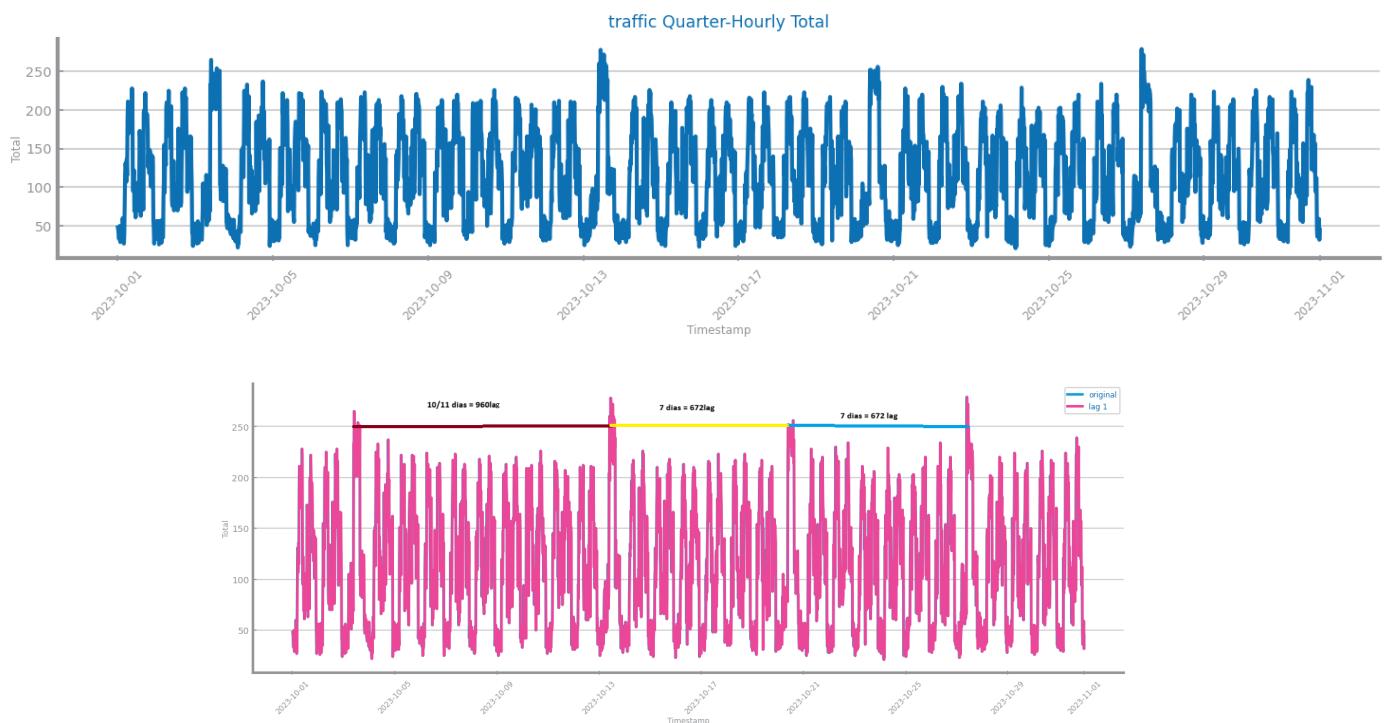
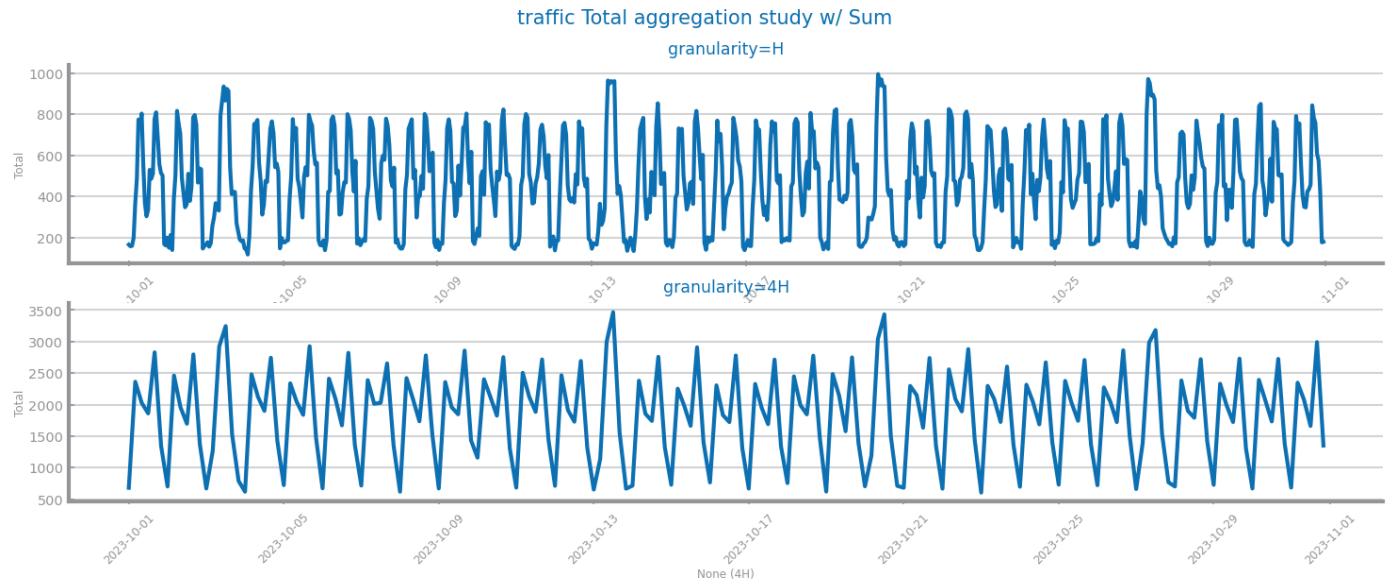


Figure 63 Time series 2 at the most granular detail



*Figure 64 & 65 Time series 2 at the second and third chosen granularity*

## Data Distribution

### DS1

The distributions don't vary in any meaningful way from one granularity to another

### DS2

Lag **96** corresponds to **One Day**

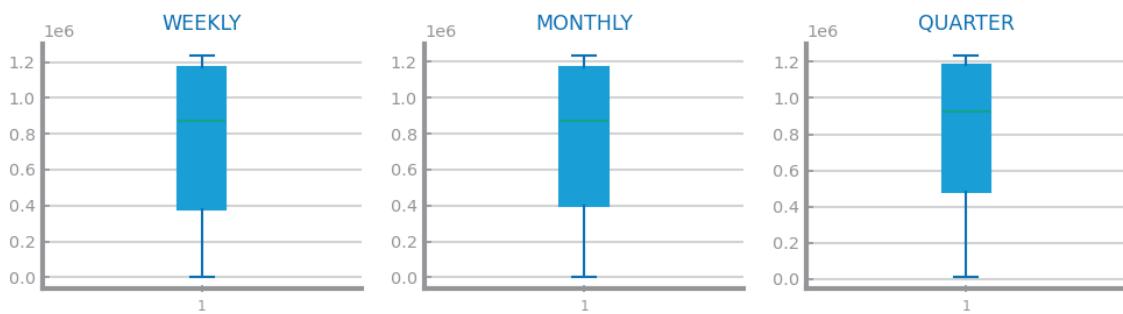
May have a trend, *autocorrelation* high and positive for small lags, and *ac* decreases as lag increases

**3 outliers in 4-Hourly Boxplot**

Autocorrelation is larger for **seasonal lags** (*multiple of seasonal - 7, 14, 192, etc.*)

Highest Peaks tend to be **one week** apart, smaller ones two times a day

No White Noise

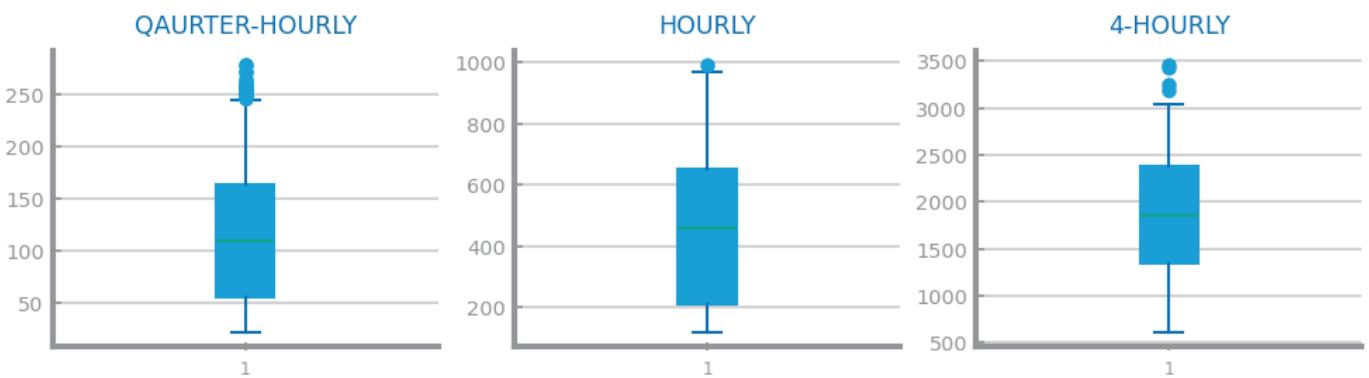


```
count    1.990000e+02
mean     7.743876e+05
std      4.356312e+05
min      0.000000e+00
25%     3.775975e+05
50%     8.691990e+05
75%     1.172049e+06
max     1.238650e+06
Name: deaths, dtype: float64
```

```
count    4.600000e+01
mean     7.750457e+05
std      4.398647e+05
min      2.500000e-01
25%     4.007008e+05
50%     8.714809e+05
75%     1.171477e+06
max     1.238175e+06
Name: deaths, dtype: float64
```

```
count    1.600000e+01
mean     7.944554e+05
std      4.483506e+05
min      7.682846e+03
25%     4.803667e+05
50%     9.235195e+05
75%     1.183809e+06
max     1.238175e+06
Name: deaths, dtype: float64
```

Figure 66 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     456.873656
std      224.577031
min      117.000000
25%     210.750000
50%     459.500000
75%     650.750000
max     995.000000
Name: Total, dtype: float64
  
```

```

count    186.000000
mean     1827.494624
std      736.869773
min      603.000000
25%     1351.000000
50%     1850.000000
75%     2385.750000
max     3466.000000
dtype: float64
  
```

Figure 67 Boxplot(s) for time series 2

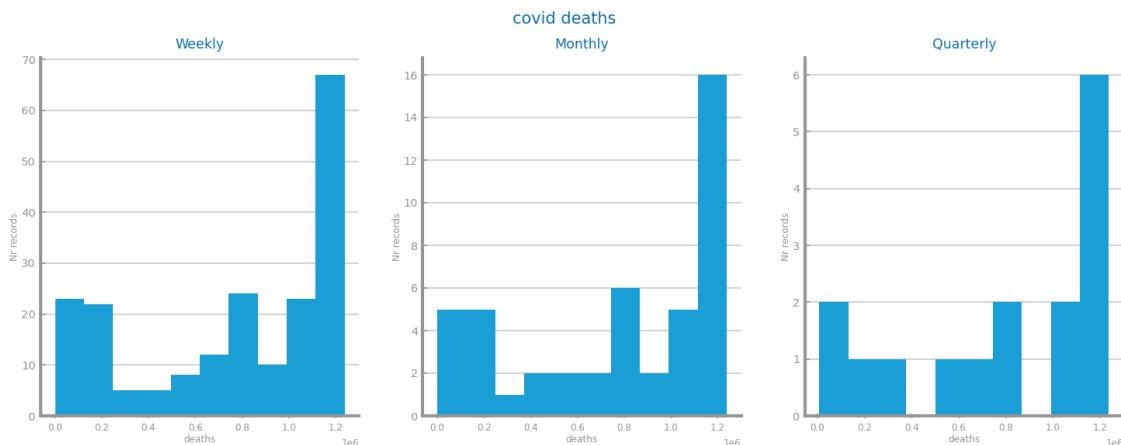


Figure 68 Histogram(s) for time series 1

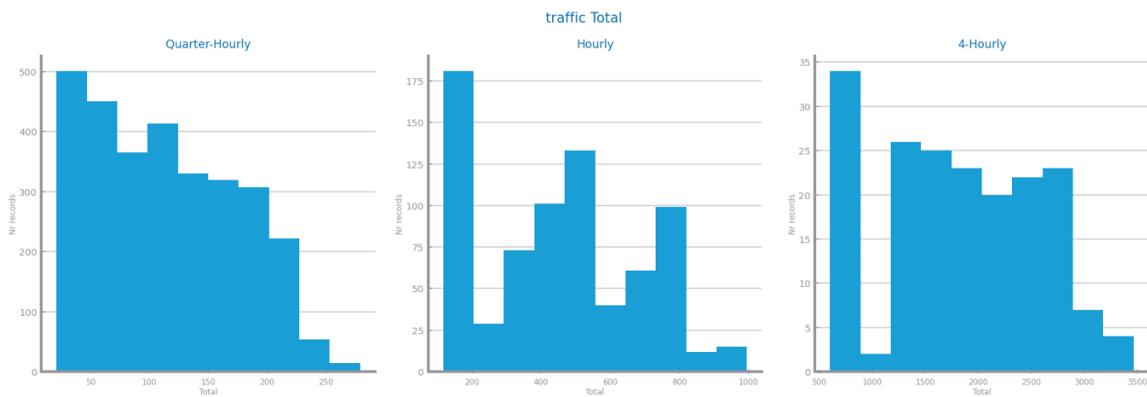


Figure 69 Histogram(s) for time series 2

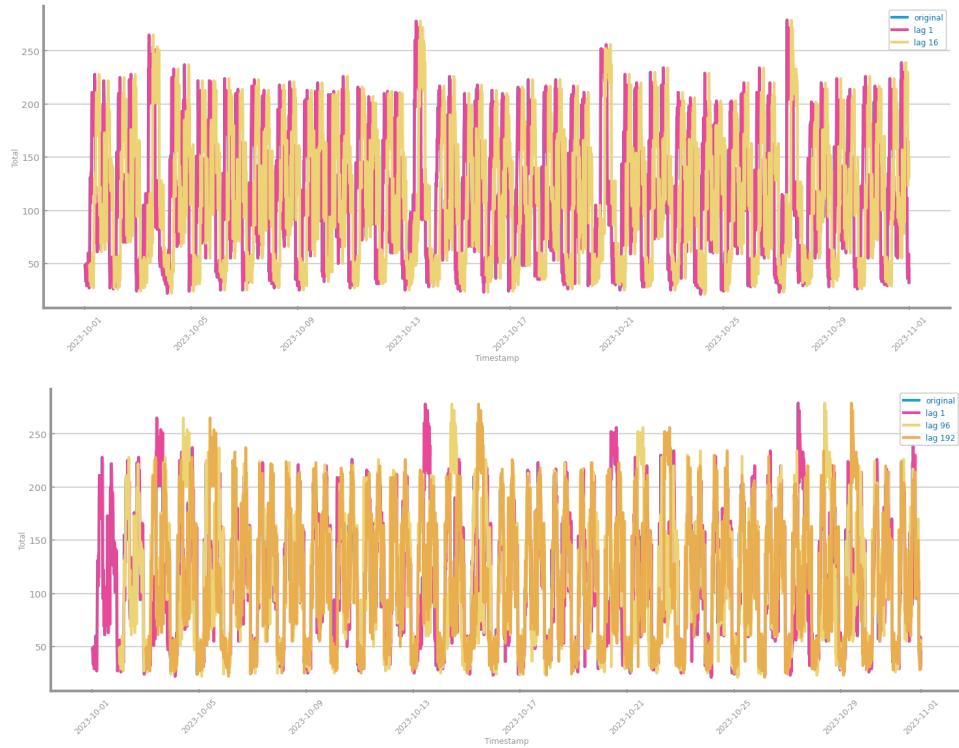


Figure xx Autocorrelation lag-plots (24,16) && (192,96) for time series 2

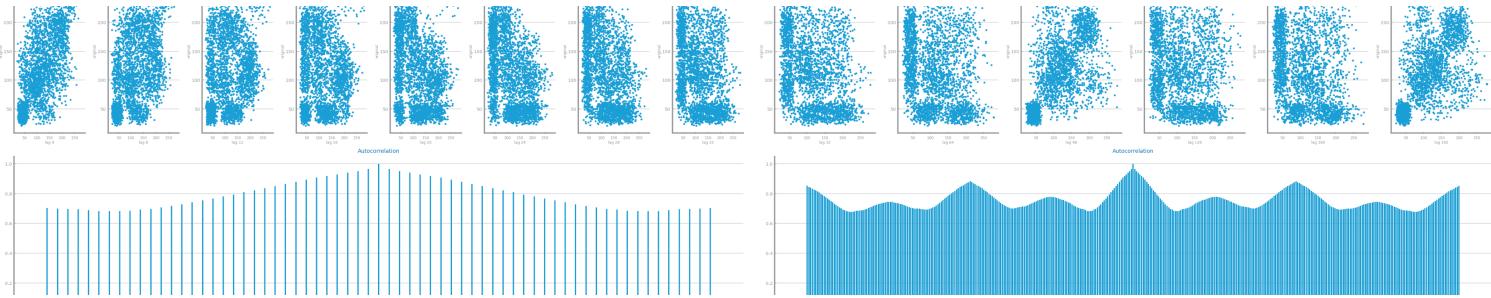


Figure xx Autocorrelation correlogram (32,4) && (192,32) for time series 2

## DS2

**Left fig - lag 0** (highest - correlated w/ itself). Decreases when we increase the lag. In **lag 26**, we see it starts to increase again. “*observations nearby in time are also nearby in value*” - Hyndman

**Right fig** - show some peaks which may be due a **seasonal pattern** in the data. Peaks appear in **lag 96**, and then again in **lag 192**.

There seems to be a combination of seasonality with trend (despite in Fig 62, there seems to be **no trend**)

## Data Stationarity

### DS1

The data is not stationary and it has an upward trend that tapers off.

### DS2

*Fig72 - STL & Additive Decomposition.* No strong up/down trend. It shows no **Seasonal patterns or Cycles**

*Fig73 - mean line flat and constant over time, suggests the average traffic volume **not have a strong up or downward trend.***

Constant Variance.

Show some regular fluctuations within each day, and suggest a *daily pattern*.

*Fig73.2 - p = 0.00 **Stationary** means the series is not defined by a trend, and does not depend on time.*

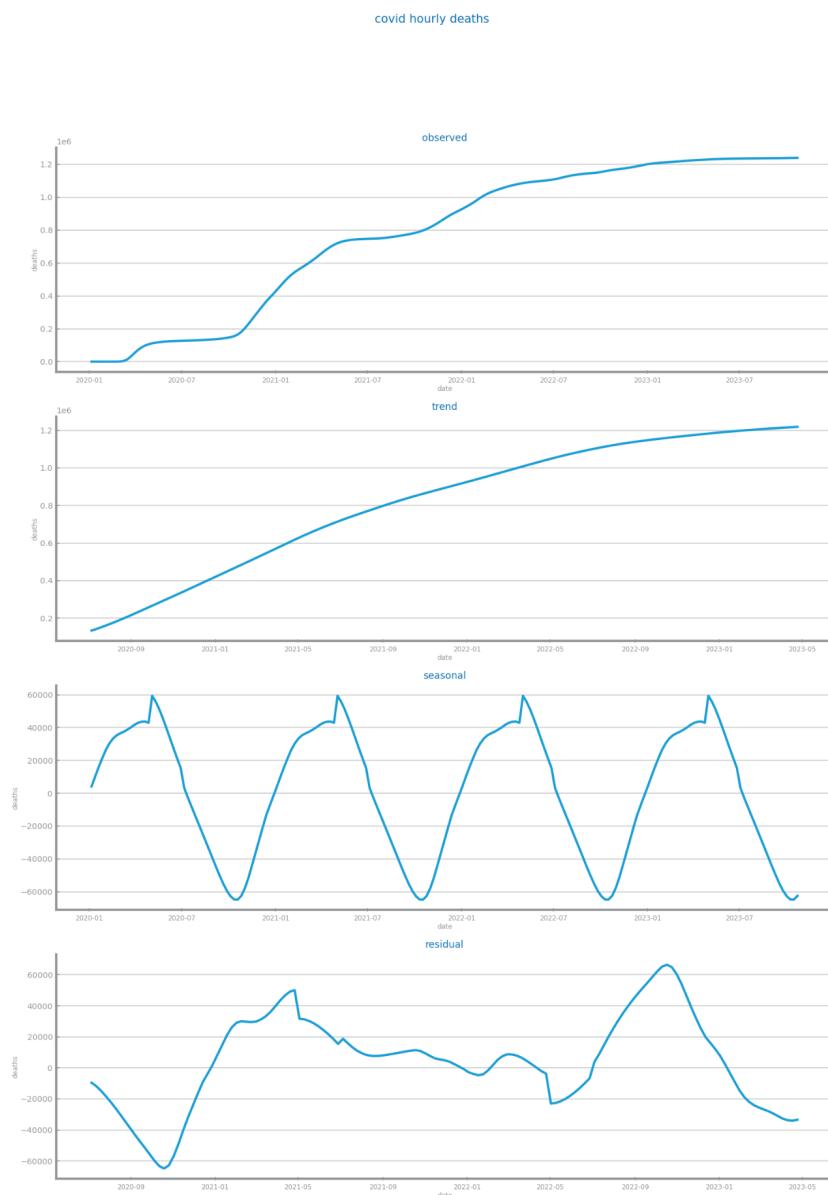


Figure 70 Components study for time series 1

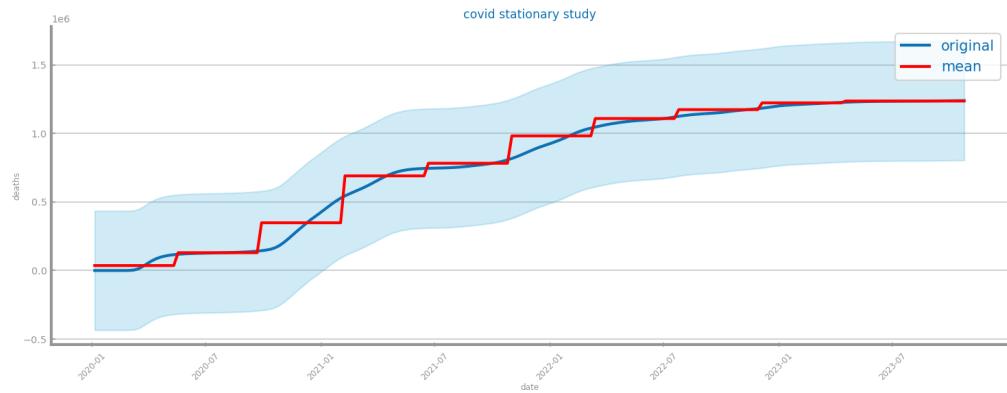
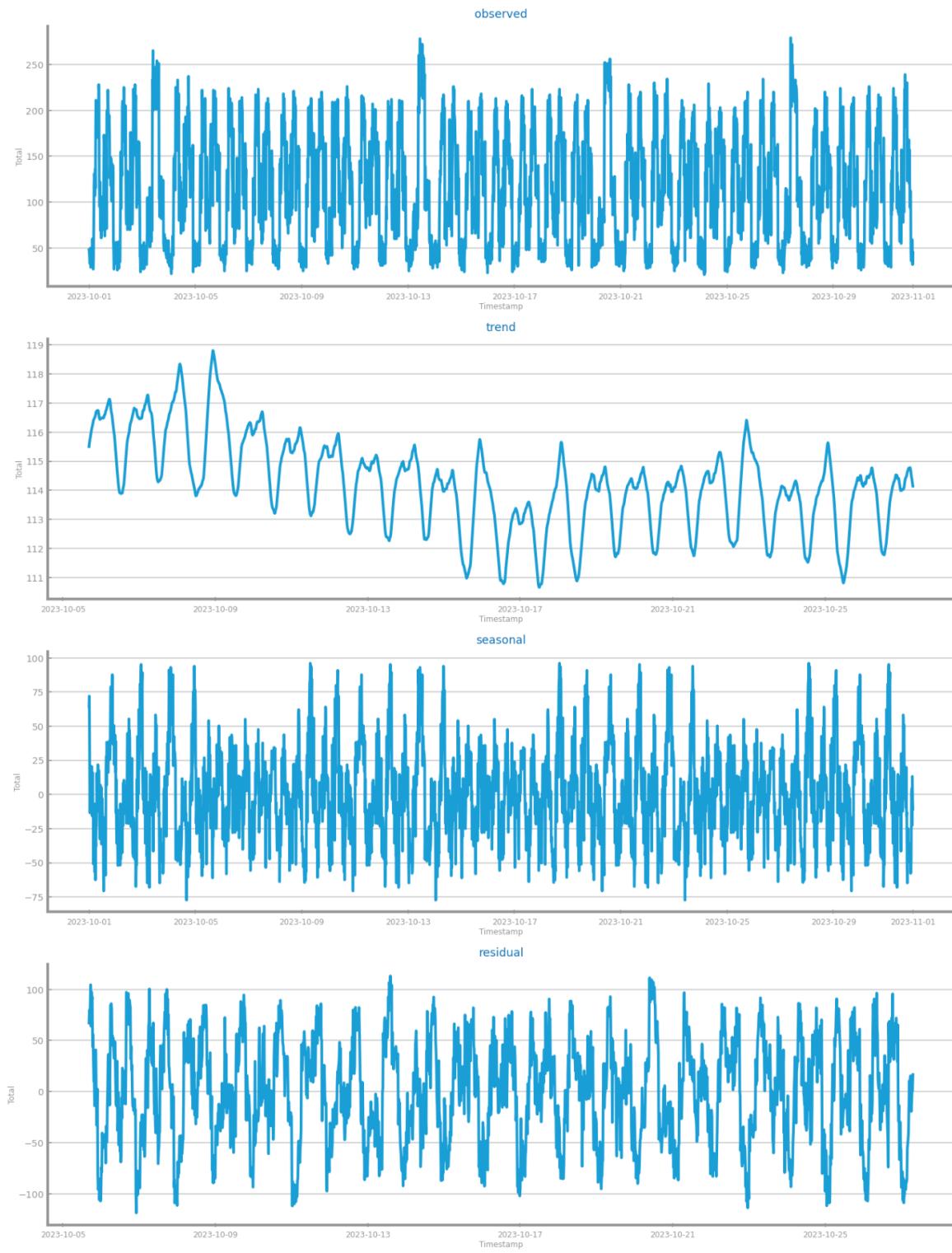


Figure 71 Stationarity study for time series 1

### traffic quarter-hourly Total



*Figure 72 Components study for time series 2*



Figure 73 Stationarity study - no bins - for time series 2



Figure 73.1 Stationarity study - 24 bins - for time series 2

```

ADF Statistic: -14.436
p-value: 0.000
Critical Values:
 1%: -3.439
 5%: -2.866
 10%: -2.569
The series is stationary

```

Figure 73.2 Stationarity study - Augmented Dickey-Fuller test - for time series 2

## 6 DATA TRANSFORMATION

### Aggregation

#### DS1

Chose atomic aggregation due to no significant improvements and data retention preference.

#### DS2

**Atomic, Hourly (second) and 4-Hourly (third)** from Profiling. Aggregation function “sum” (retains total volume & introduces less error)

Opted for **Atomic** as plot shows pattern/info loss w/ other aggregations

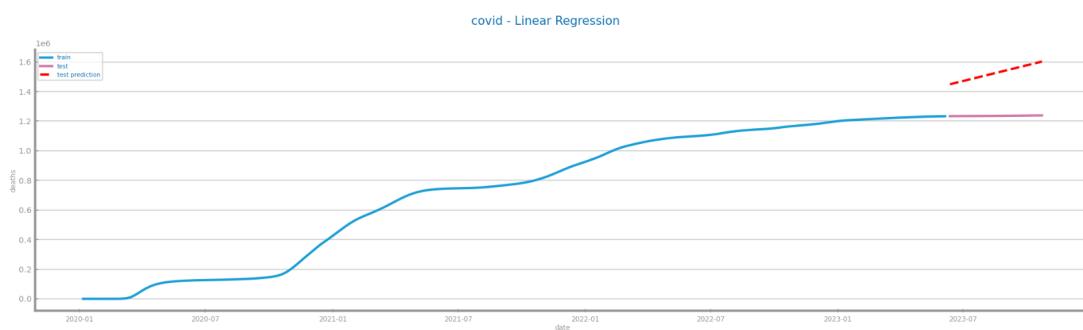


Figure 74.1 Forecasting plots after different aggregations on time series 1 (Weekly)

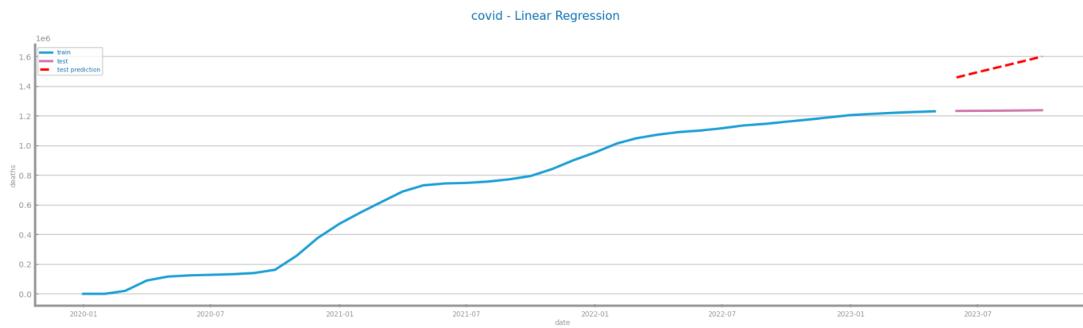


Figure 74.2 Forecasting plots after different aggregations on time series 1 (Monthly)

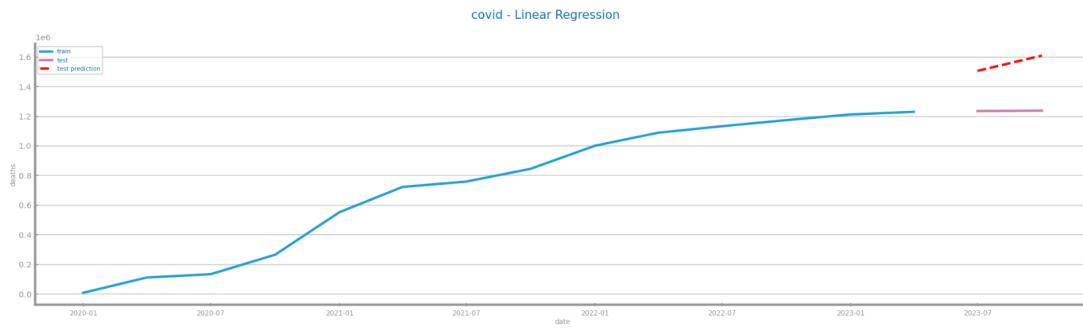


Figure 74.3 Forecasting plots after different aggregations on time series 1 (Quarterly)

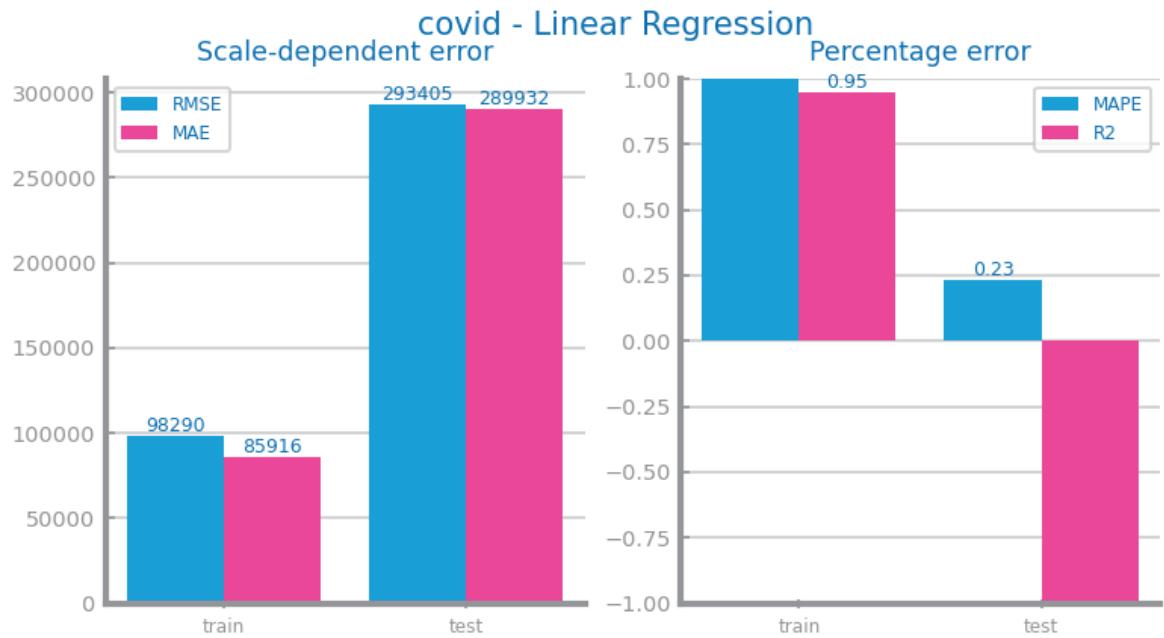


Figure 75.1 Forecasting results after different aggregations on time series 1 (Weekly)

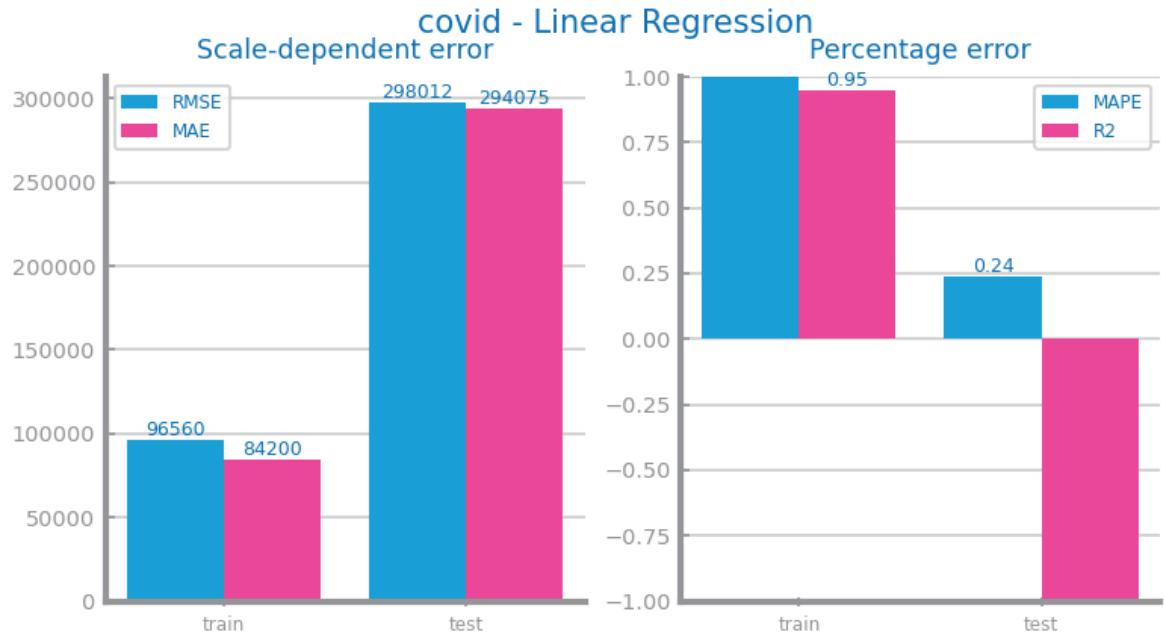


Figure 75.2 Forecasting results after different aggregations on time series 1 (Monthly)

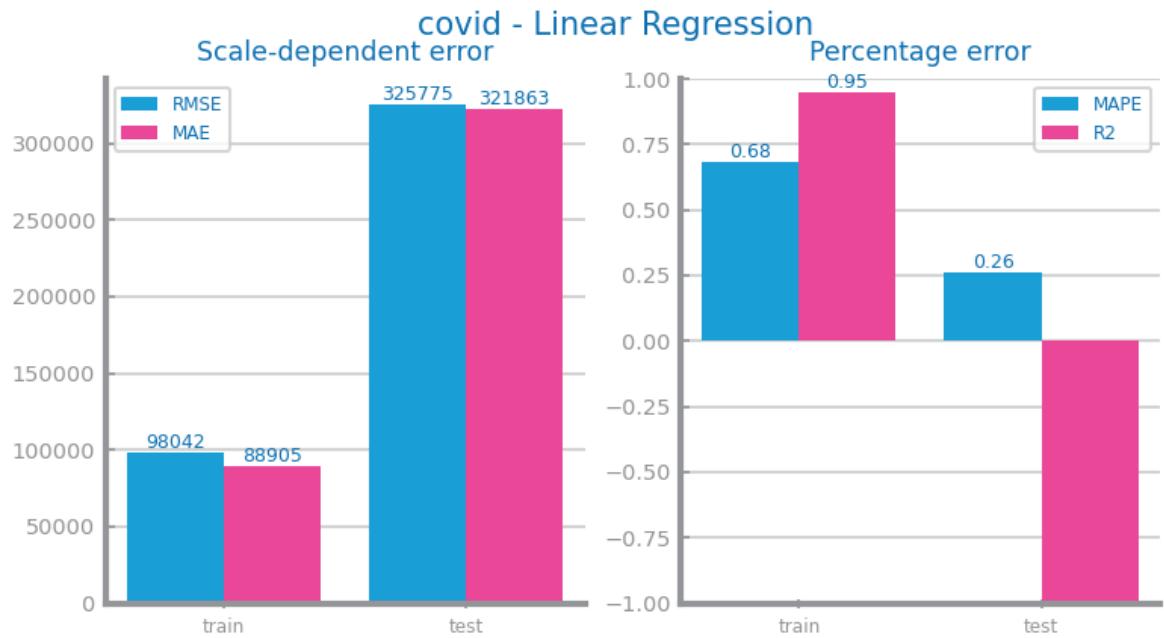


Figure 75.3 Forecasting results after different aggregations on time series 1 (Quarterly)

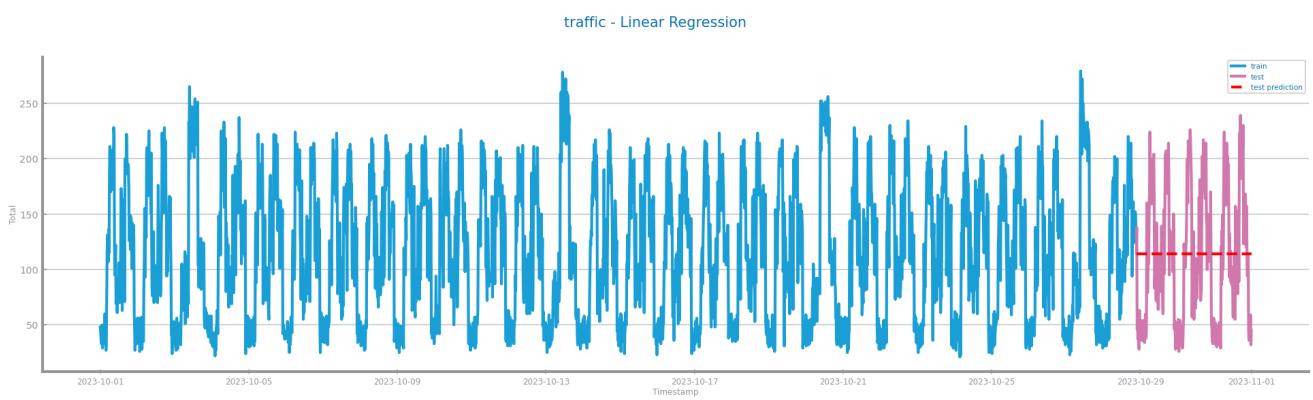


Figure 76 Forecasting plots after different aggregations - ATOMIC - on time series 2

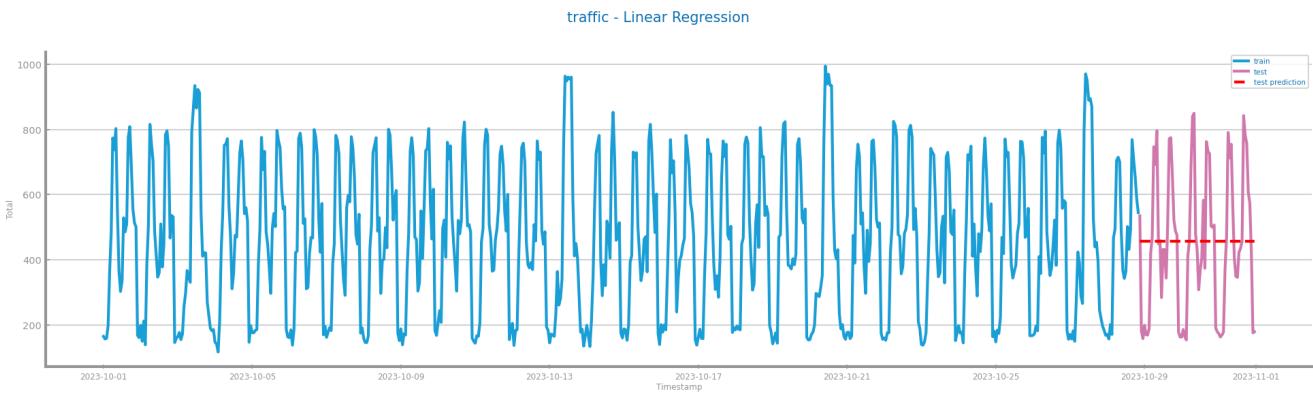


Figure 76 Forecasting plots after different aggregations - SECOND - on time series 2

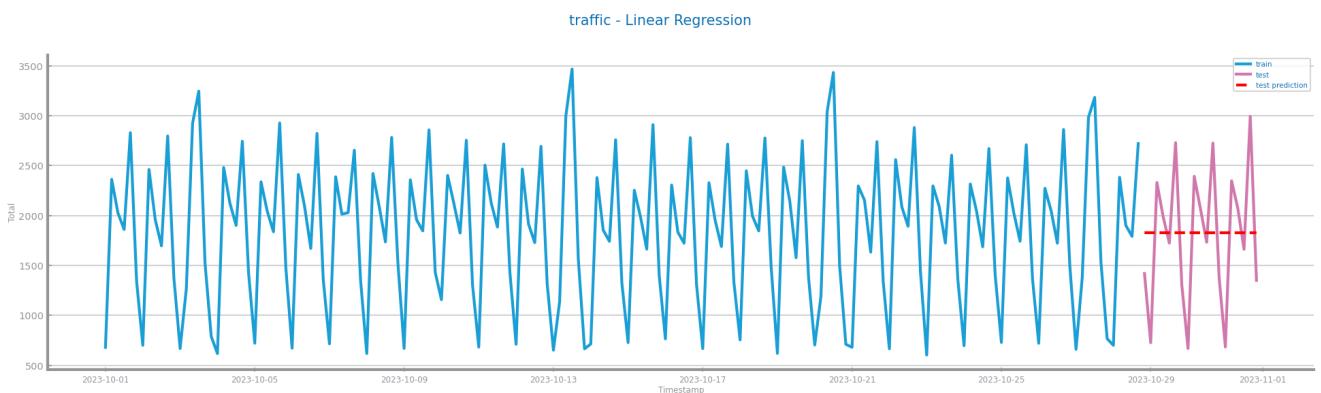


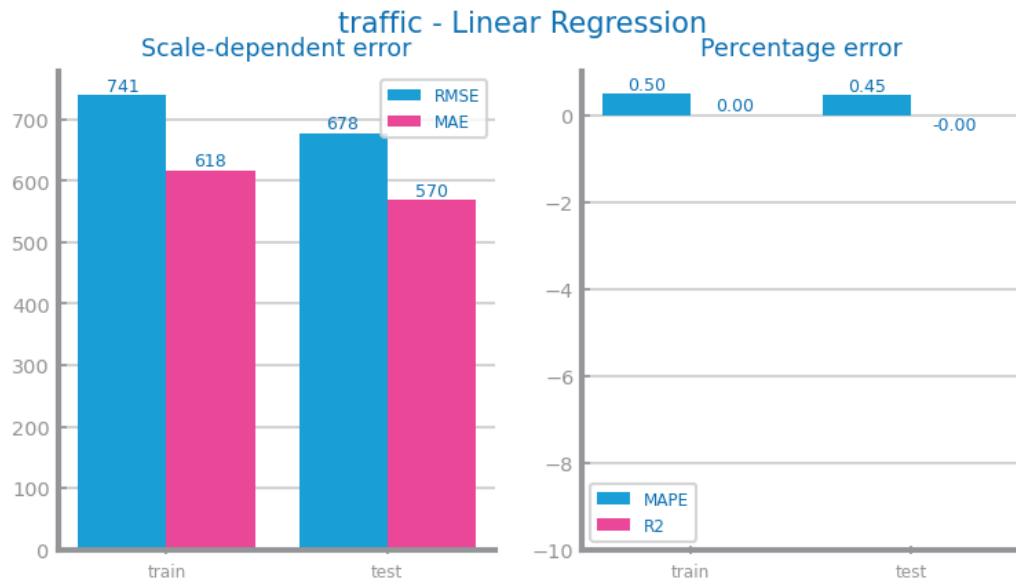
Figure 76 Forecasting plots after different aggregations - THIRD - on time series 2



Figure 77 Forecasting results after different aggregations - ATOMIC - on time series 2



Figure 77 Forecasting results after different aggregations - SECOND - on time series 2



*Figure 77 Forecasting results after different aggregations - THIRD - on time series 2*

## Smoothing

Only done in the Training Set

Used Backward Fill to impute MV in both

### DS1

Retained Original as window didn't improve results; data is cumulative

### DS2

Applied short-window moving average

WIN\_SIZE=16 same as 4 Hours

Fig72 showed lots of noise & variance

Smooth short-term fluctuations & clarified weekly patterns

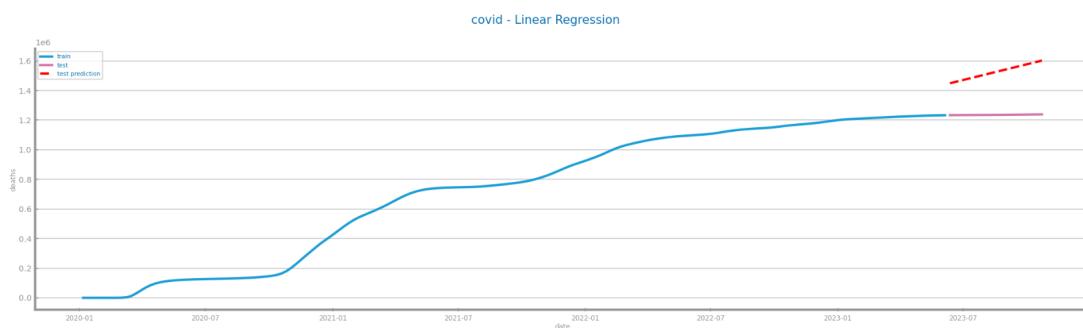


Figure 78 Forecasting plots after different smoothing parameterisations on time series 1

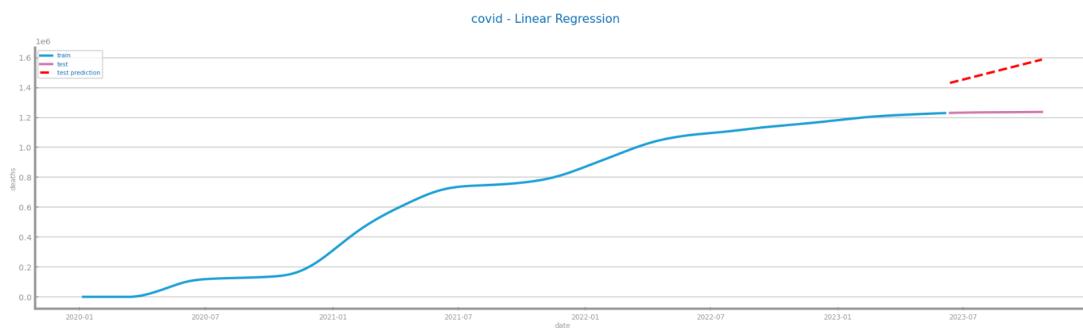


Figure 78 Forecasting plots after different smoothing parameterisations on time series 1

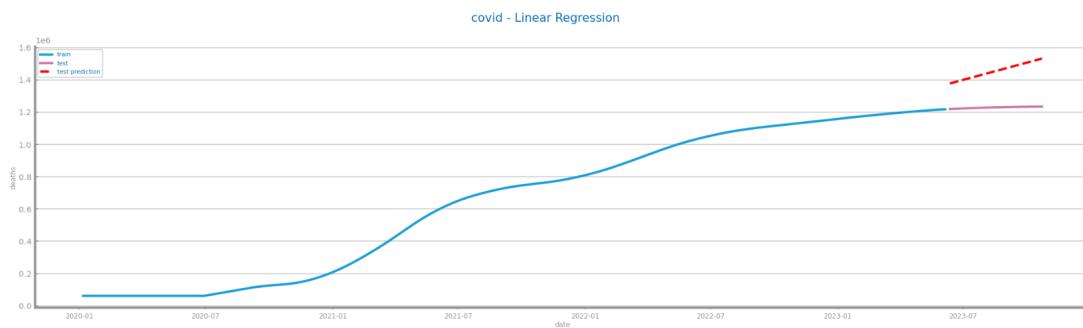
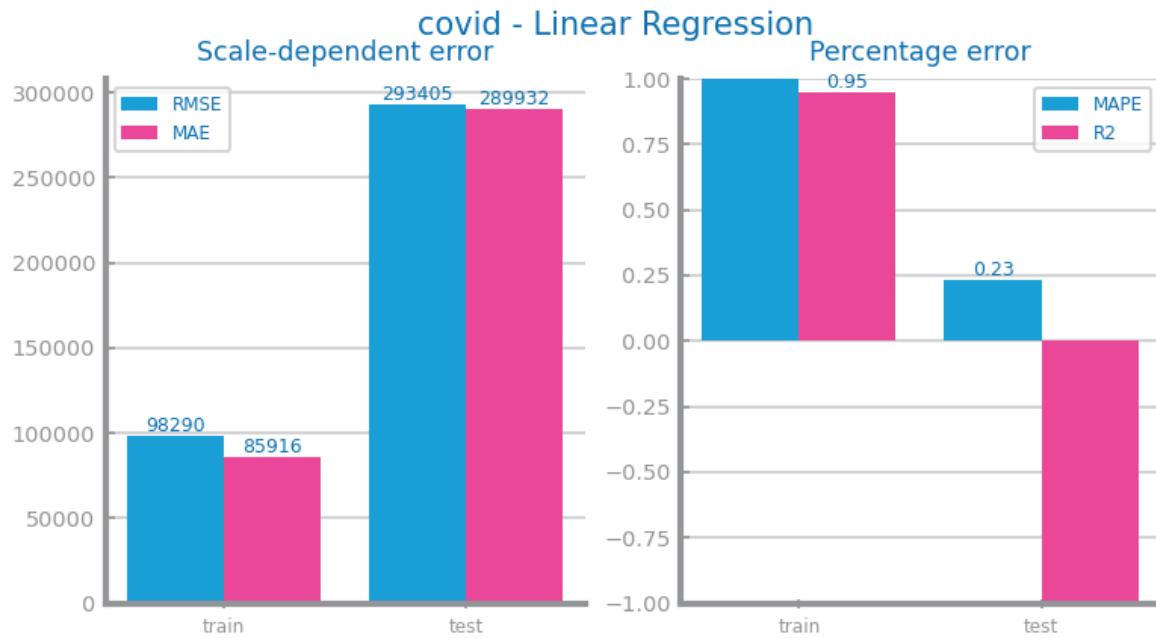
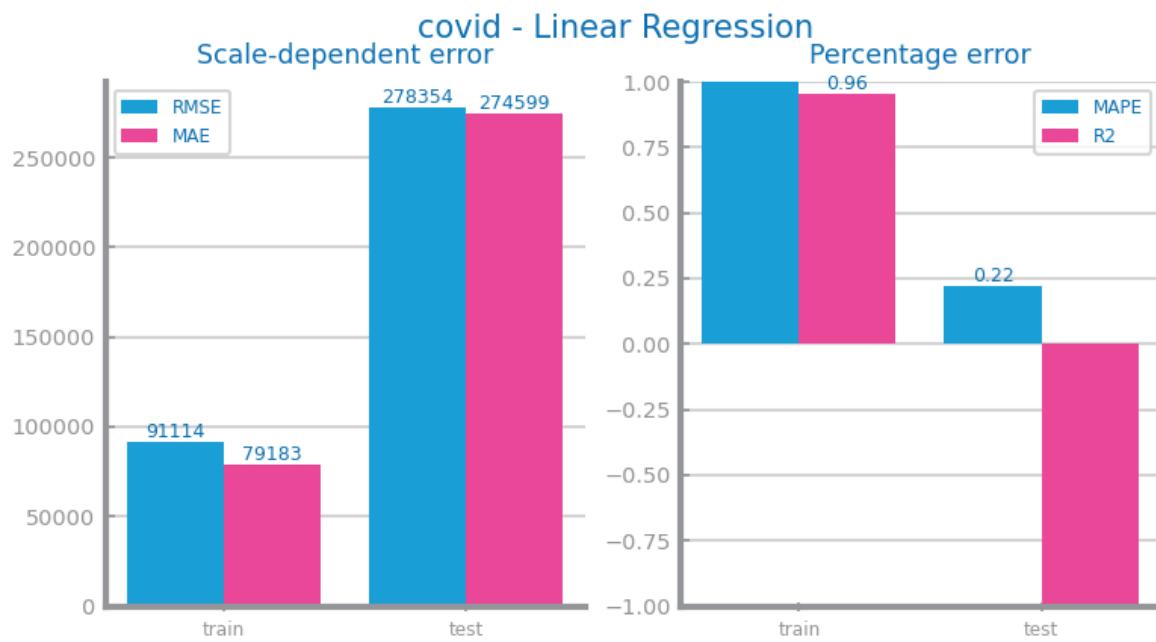


Figure 78 Forecasting plots after different smoothing parameterisations on time series 1



*Figure 79 Forecasting results after different smoothing parameterisations on time series 1*



*Figure 79 Forecasting results after different smoothing parameterisations on time series 1*

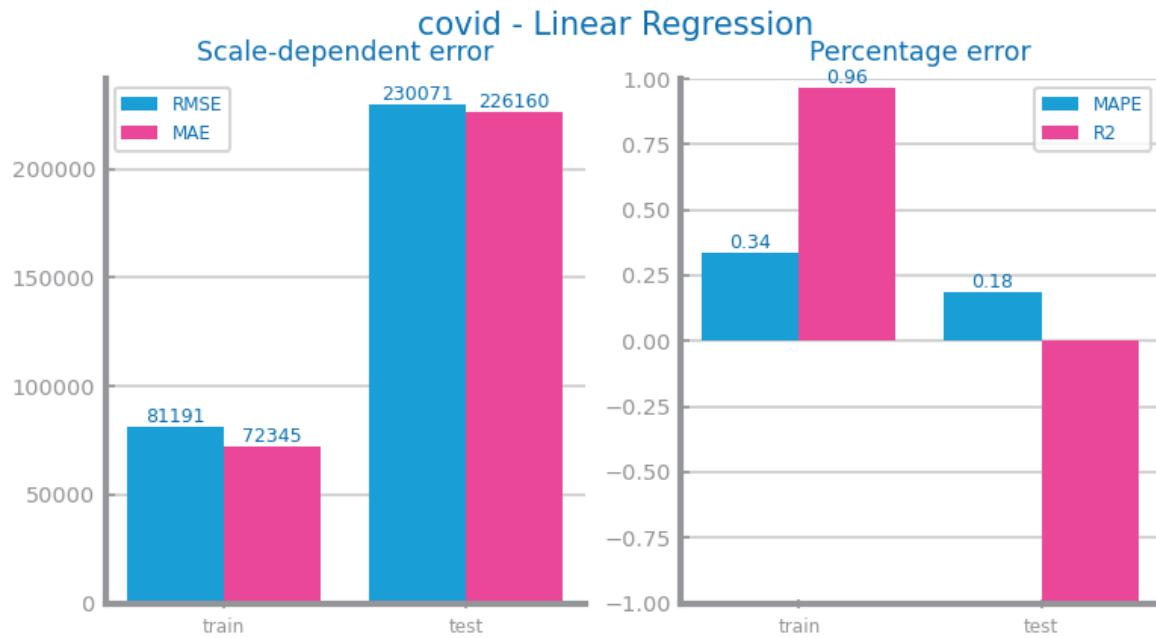


Figure 79 Forecasting results after different smoothing parameterisations on time series 1

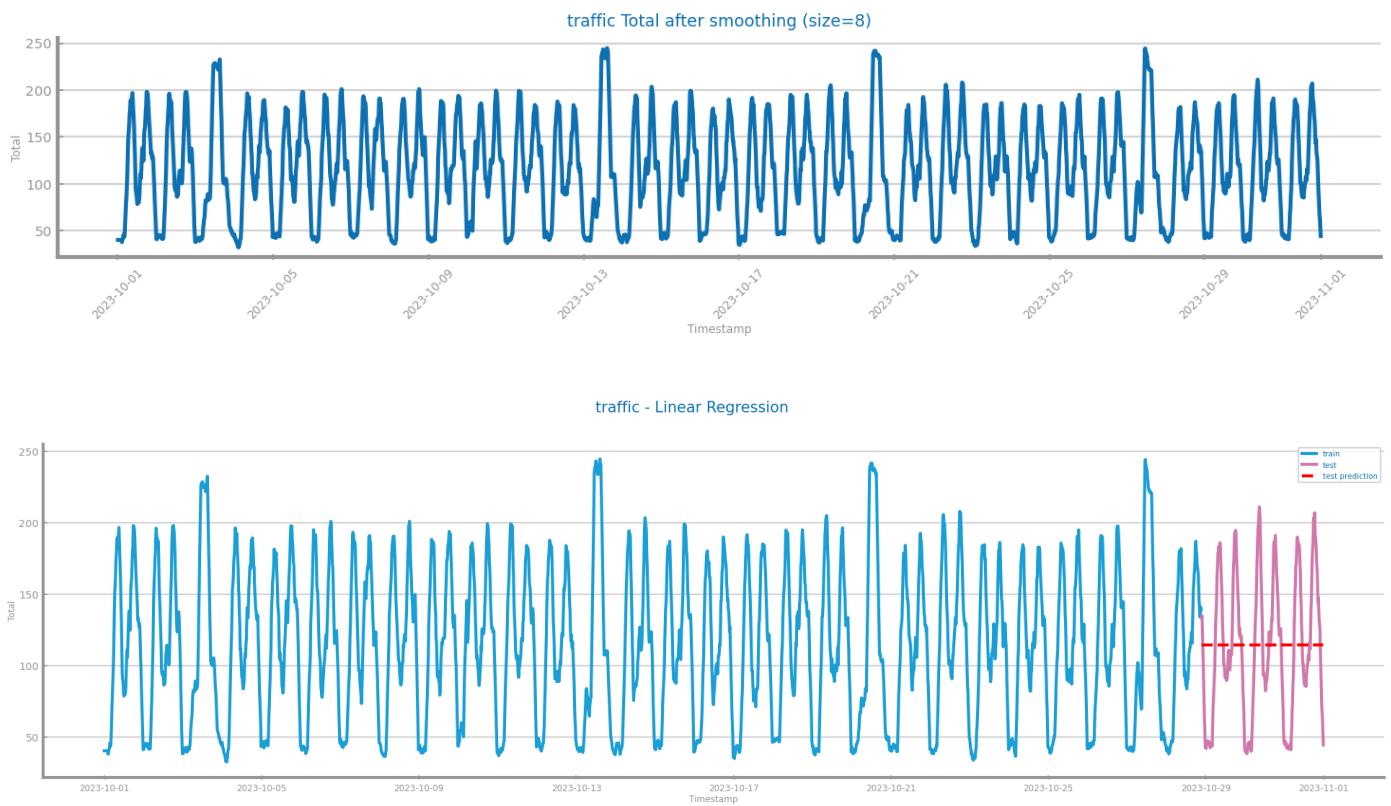


Figure 80 Forecasting plots after different smoothing parameterisations - WIN\_SIZE (8) - on time series 2

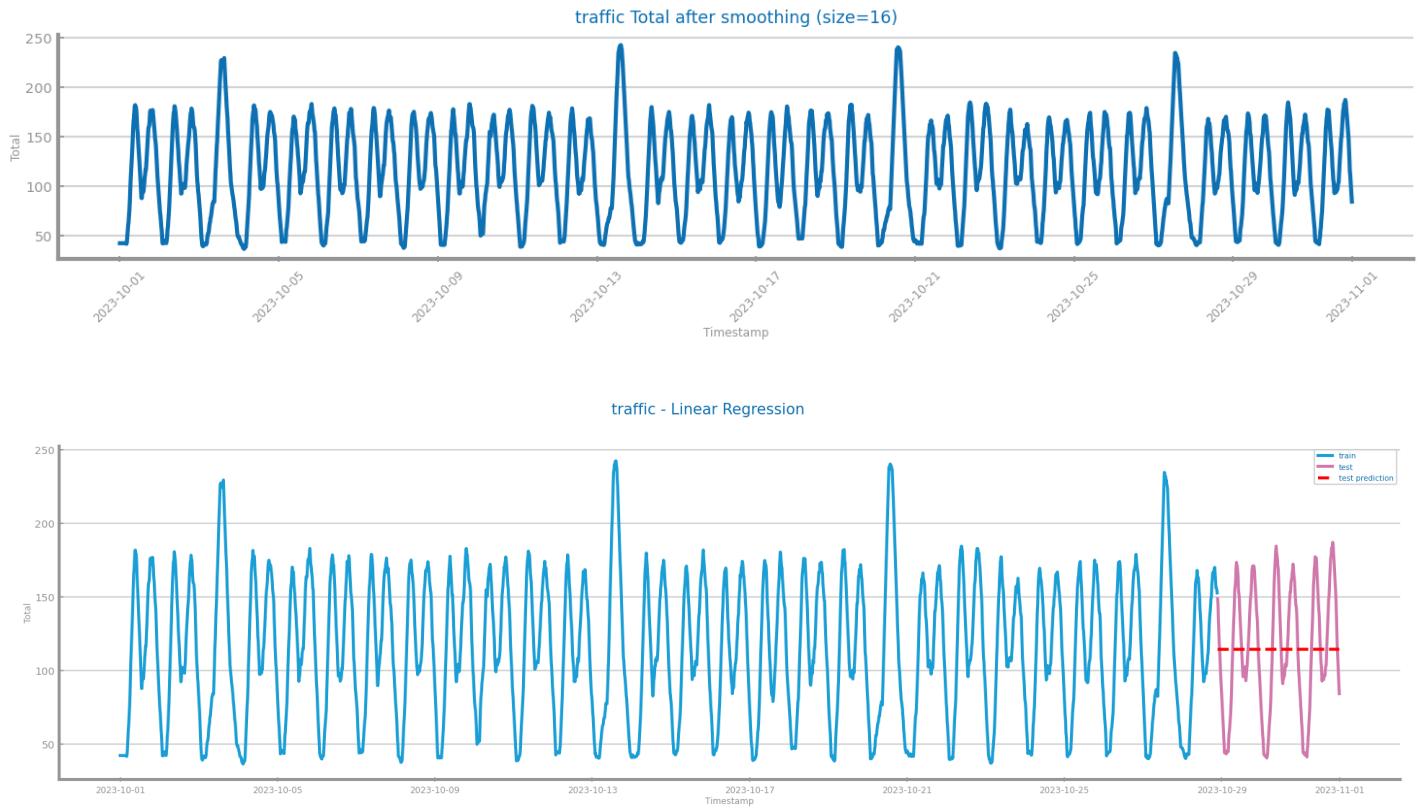


Figure 80.1 Forecasting plots after different smoothing parameterisations - WIN\_SIZE (16) - on time series 2

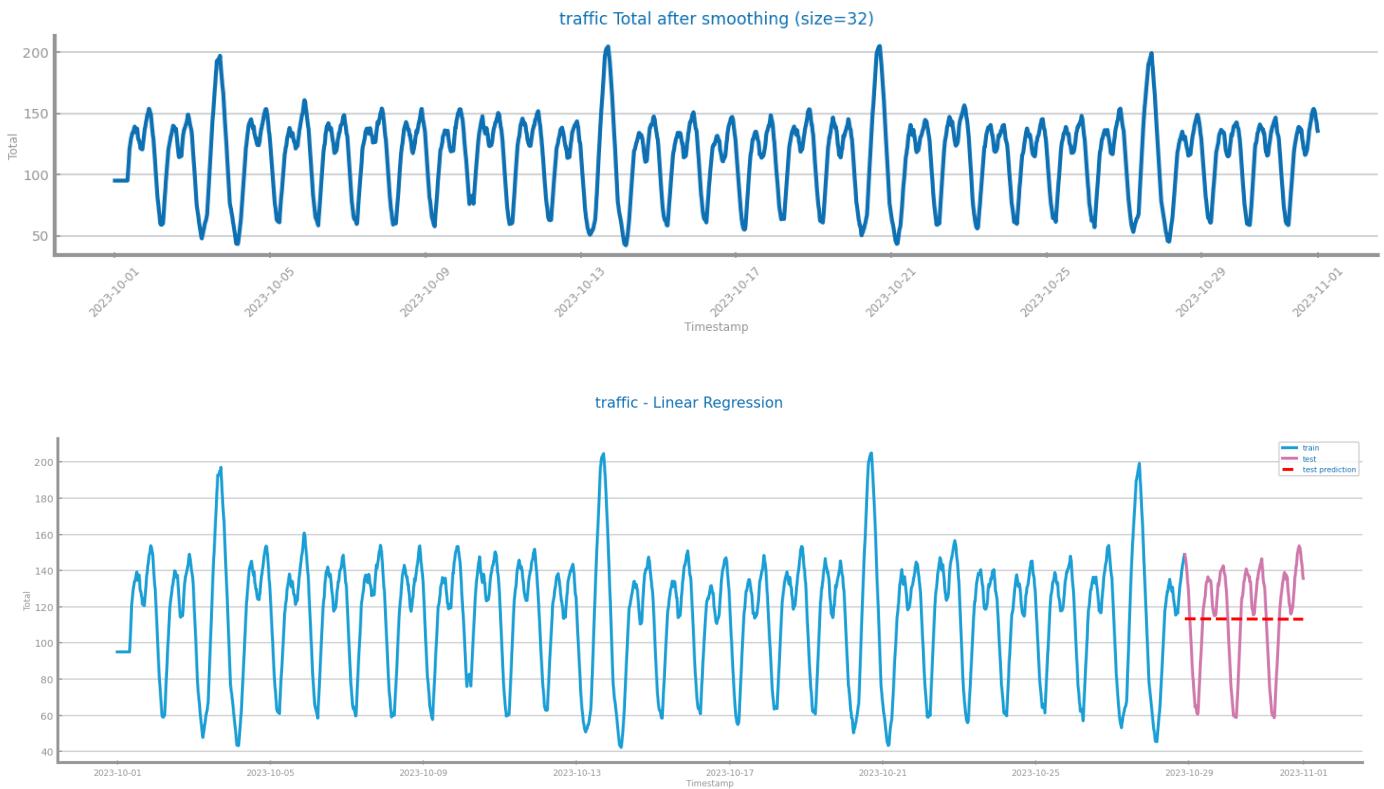
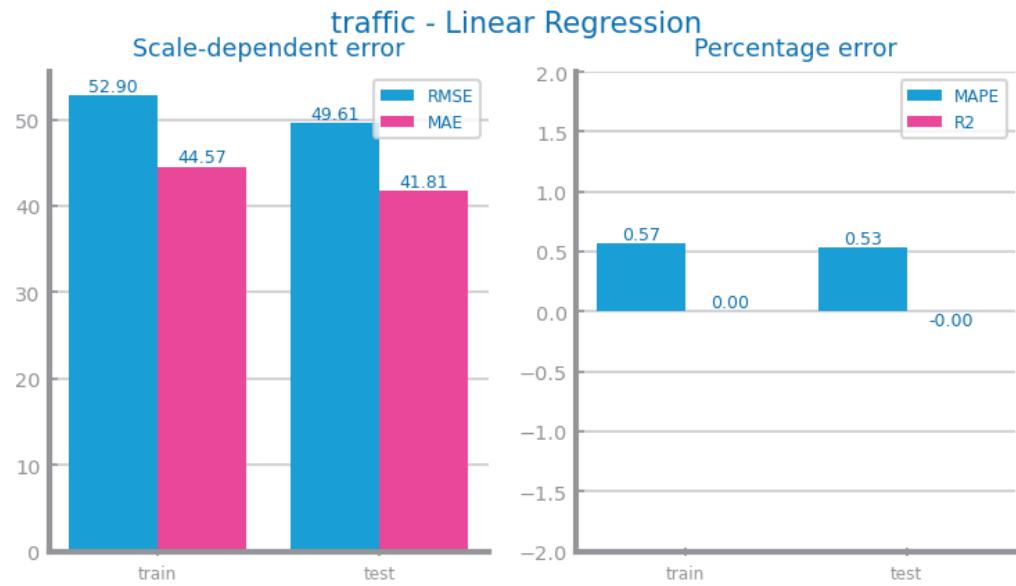
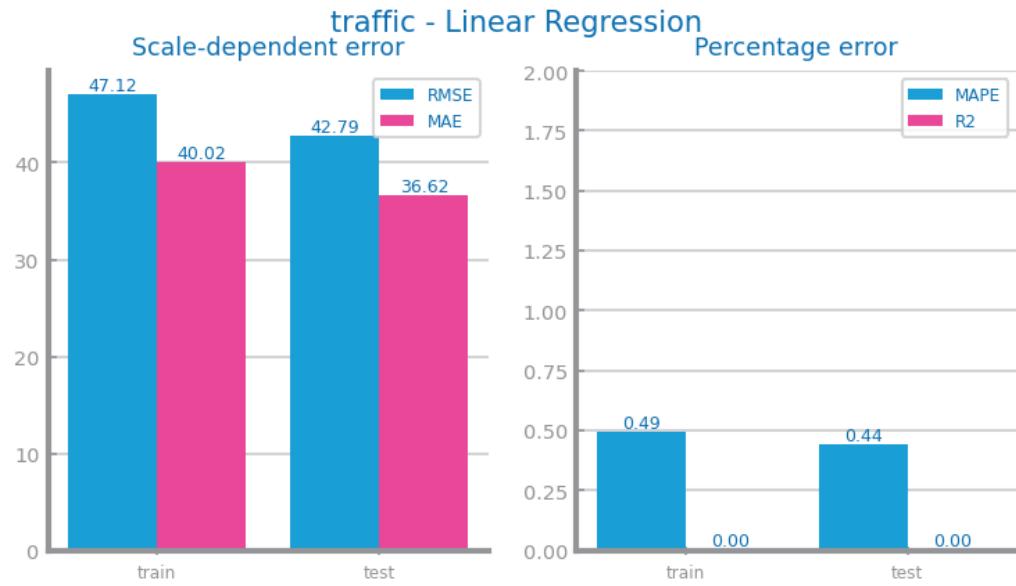


Figure 80.2 Forecasting plots after different smoothing parameterisations - WIN\_SIZE (32) - on time series 2



*Figure 81 Forecasting results after different smoothing parameterisations - WIN\_SIZE (8) - on time series 2*



*Figure 81.1 Forecasting results after different smoothing parameterisations - WIN\_SIZE (16) - on time series 2*



*Figure 81.2 Forecasting results after different smoothing parameterisations - WIN\_SIZE (32) - on time series 2*

## Differentiation

### DS1

Used to remove the Series dependence on time

Second differentiation was adopted

### DS2

Not applied

Scale-dependent error decreased, MAPE has non-viable values; due to the granularity we chose to proceed the transformation

The plot gives no information, introduces noise

Proceed w/ the result from *Smoothing*

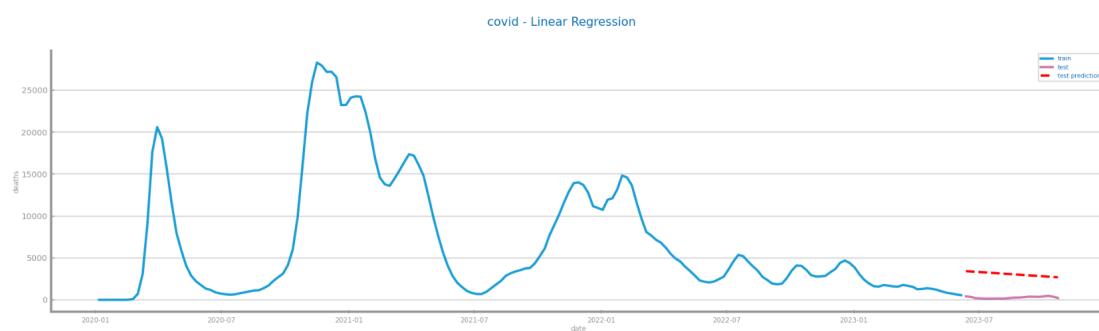
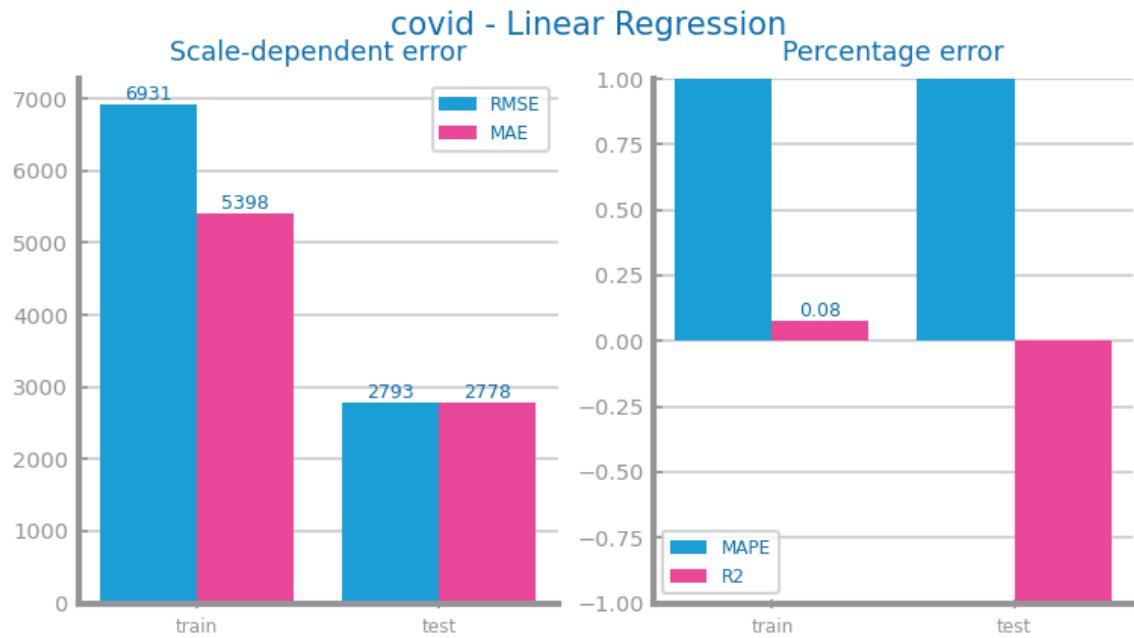


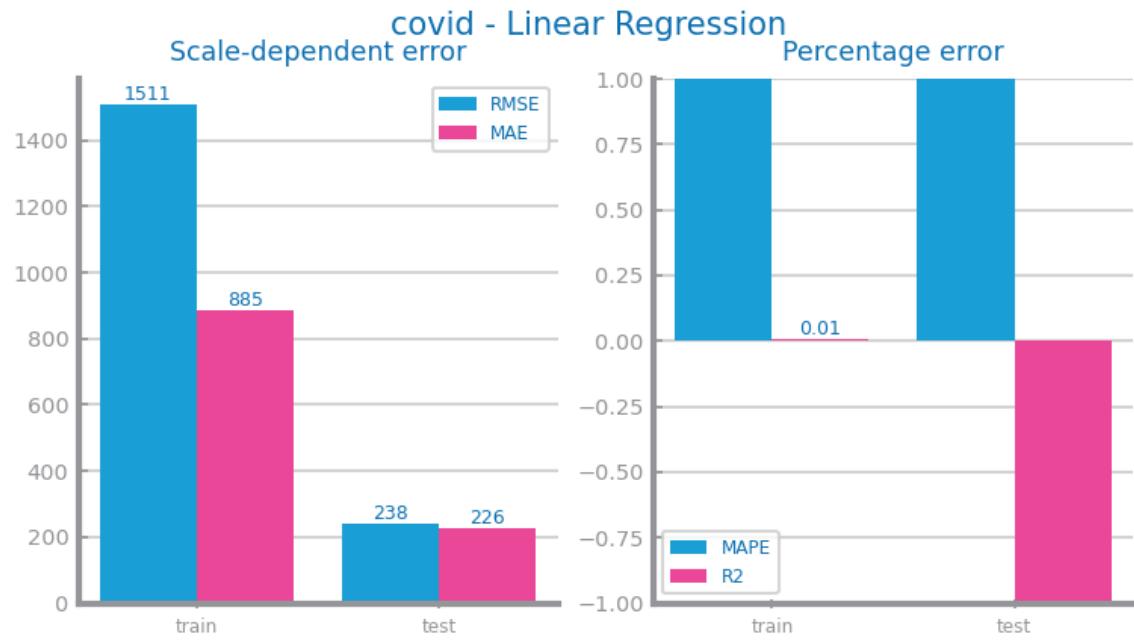
Figure 82.1 Forecasting plots after first differentiation of time series 1



Figure 82.2 Forecasting plots after second differentiation of time series 1



*Figure 83.1 Forecasting results after first and second differentiation of time series 1*



*Figure 83.2 Forecasting results after first and second differentiation of time series 1*

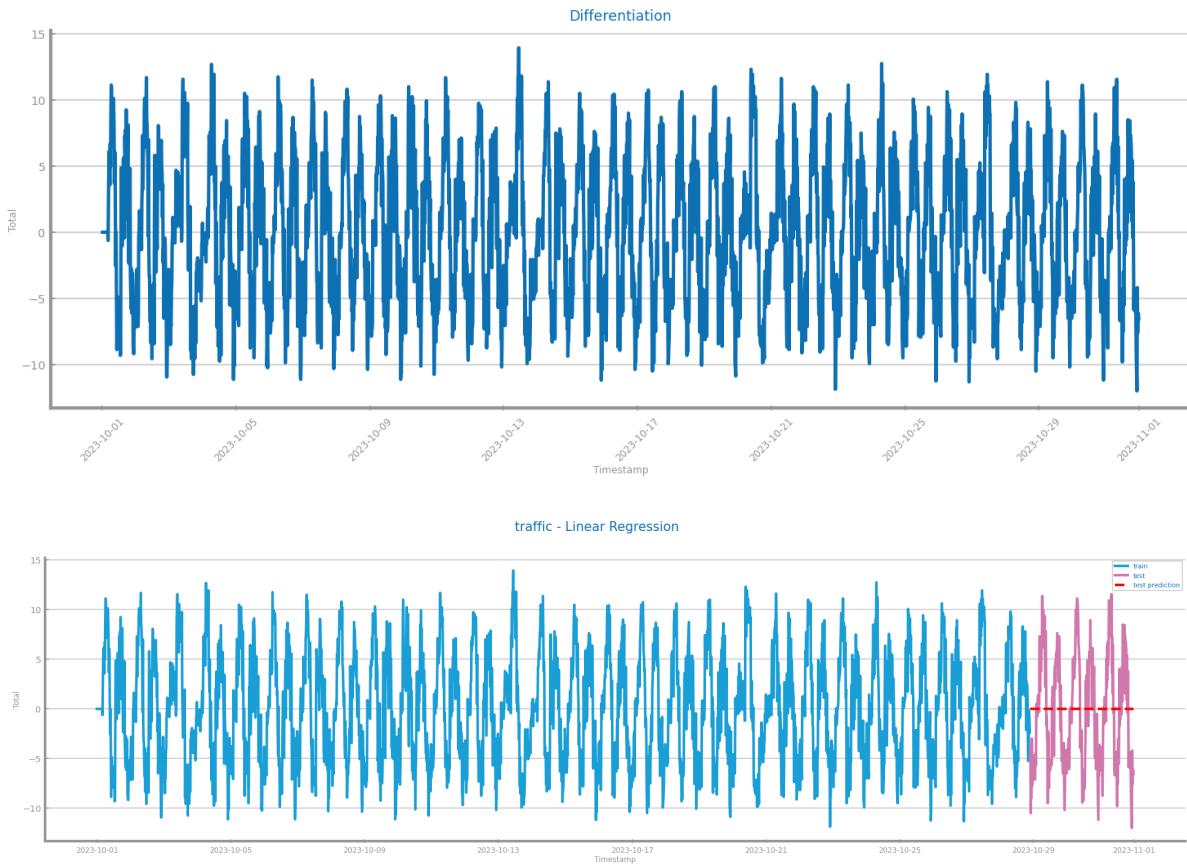


Figure 84 Forecasting plots after first differentiation of time series 2

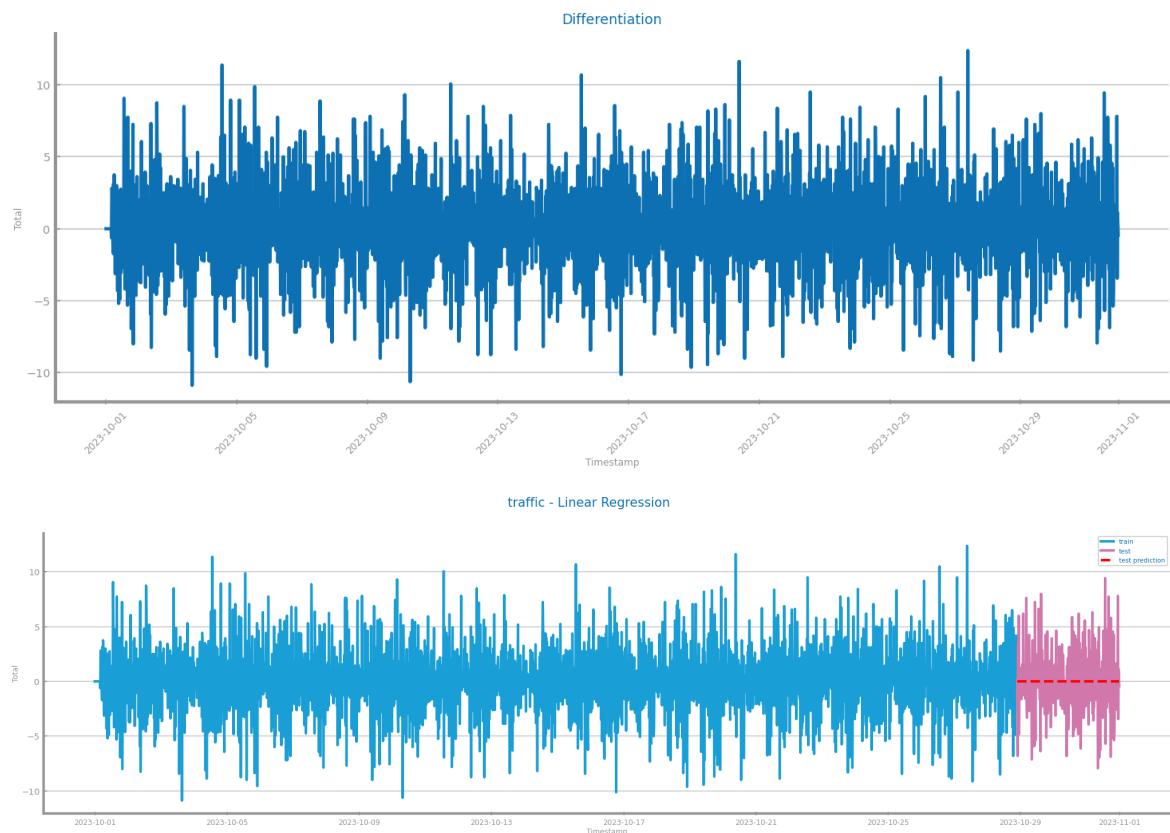


Figure 84.1 Forecasting plots after second differentiation of time series 2



```
{'RMSE': [5.474, 5.591], 'MAE': [4.595, 4.803]} {'MAPE': [2888839254183.7, 53230202313.469], 'R2': [1.53400e-05, -0.00168]}
```

*Figure 85 Forecasting results after first differentiation of time series 2*



```
{'RMSE': [3.033, 2.998], 'MAE': [2.267, 2.283]} {'MAPE': [99057114225.309, 357566406282.854], 'R2': [2.74627e-07, -9.4198e-06]}
```

*Figure 85.1 Forecasting results after second differentiation of time series 2*

## Other transformations (Scaling)

No Missing Values in both (not applied)

Used for LSTM suitability; by speeding learning process & better convergence

### DS1

Minor result changes; Used for LSTM

### DS2

LSTM benefits from *Log transformation*

ARIMA can stabilize the variance, which we can benefit from

Greatly improved MAPE and scale-dependent error



Figure 82 Forecasting plots after applying other transformations over time series 1

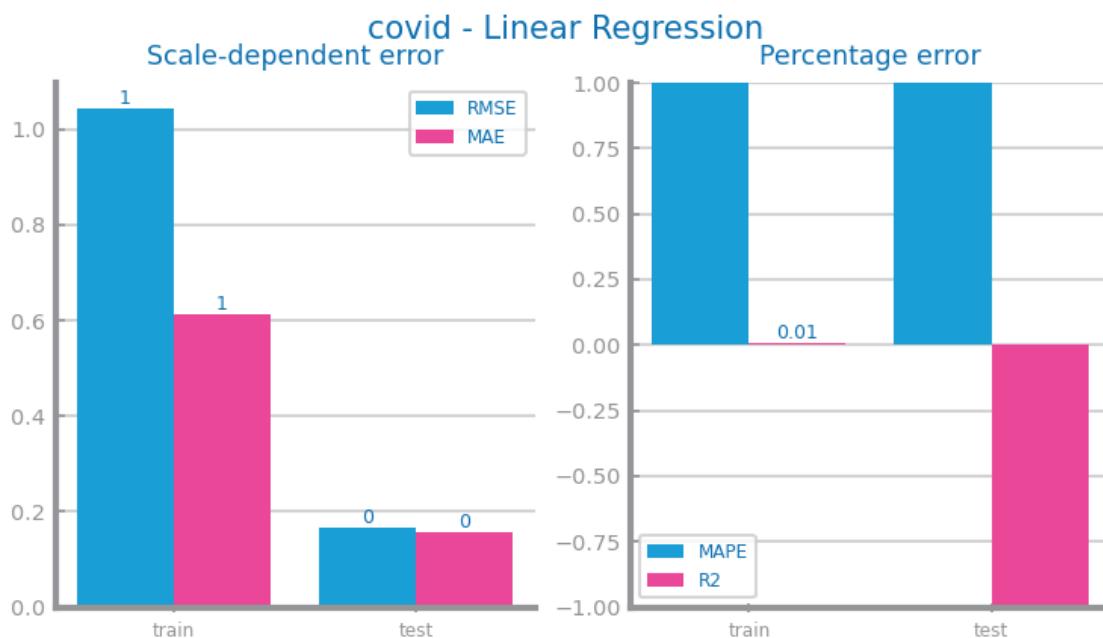


Figure 83 Forecasting results after applying other transformations over time series 1

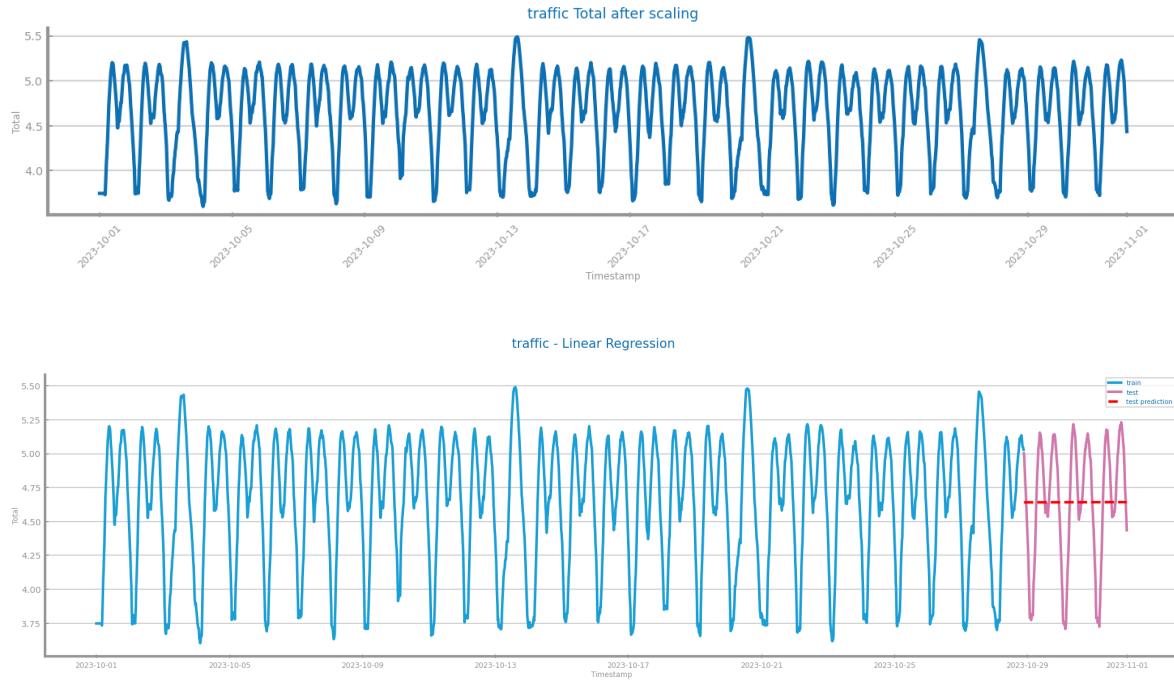


Figure 84 Forecasting plots after applying other transformations over time series 2



Figure 85 Forecasting results after applying other transformations over time series 2

## 7 MODELS' EVALUATION

### DS1

We only used the second derivative and scaling for transformations, and we measured our models accuracy by the R2.

### DS2

Didn't based in Scale-dependent error most of the time

MAPE guides us through the *Transformation*

Maximize R2 from now on

Linear Regression didn't capture patterns/relationships in the data effectively, was practically **0**, as expected. Not the appropriate model to base our decision over which operation leads to better results in *Transformation*

### Simple Average Model

### DS1

Expected results due lack of parameters

### DS2

Ignore trends, seasonality, and cycles; low R2

Unsuitable due to the **high variability** around the mean, as shown *Profiling Fig73*

It doesn't perform well

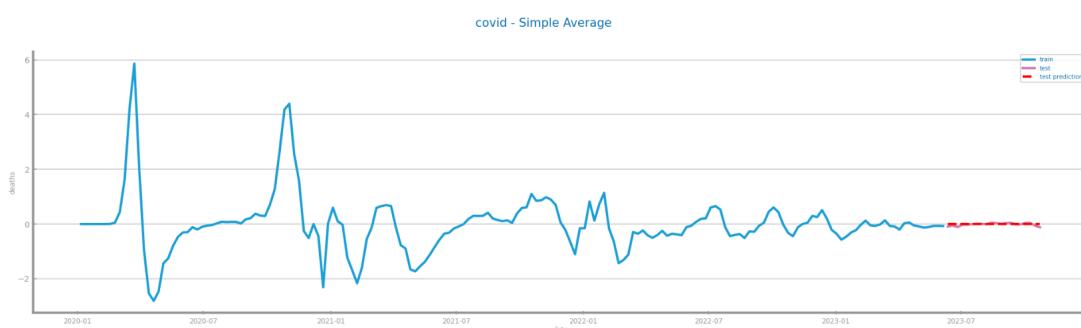


Figure 86 Forecasting plots obtained with Simple Average model over time series 1

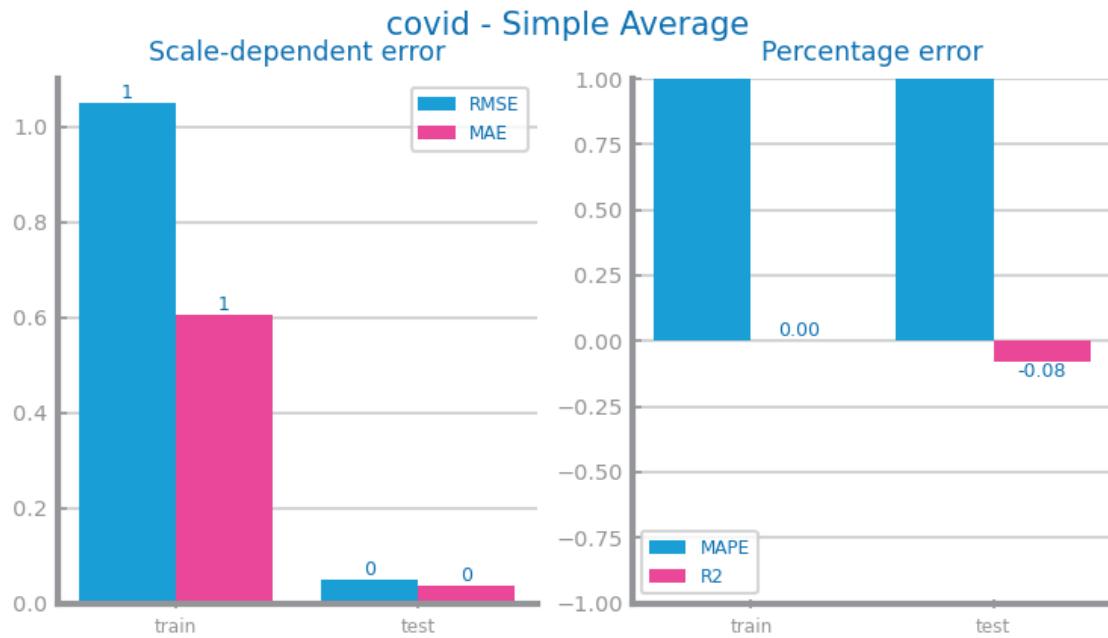


Figure 86 Forecasting results obtained with Simple Average model over time series 1

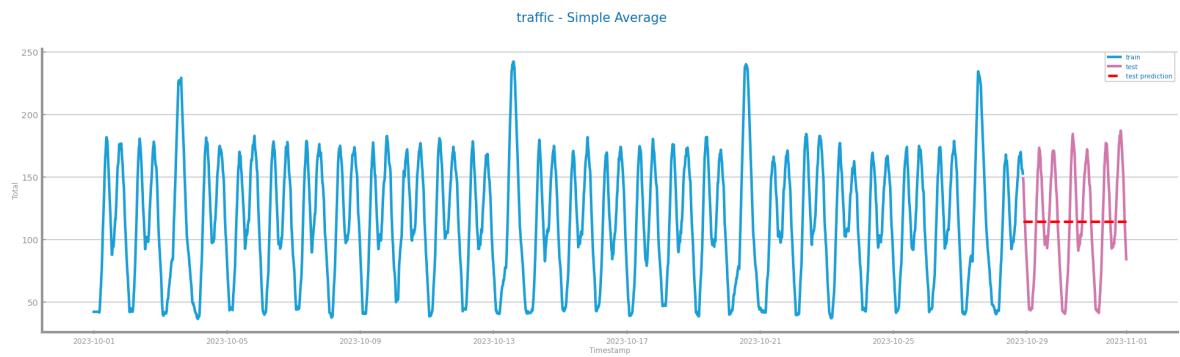


Figure 87 Forecasting plots obtained with Simple Average model over time series 2

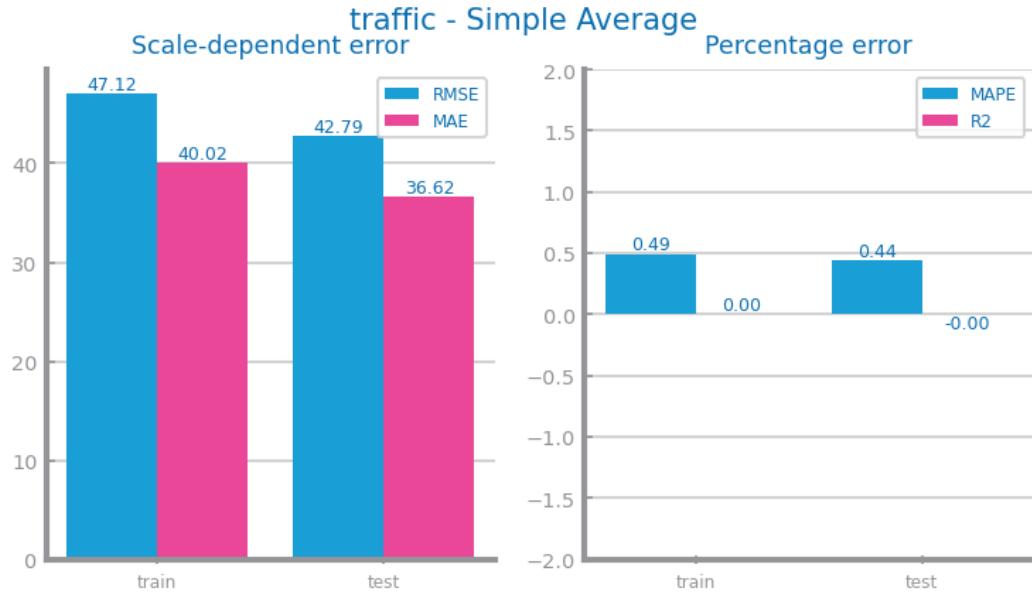


Figure 86 Forecasting results obtained with Simple Average model over time series 2

## Persistence Model

### DS1

The optimistic model showed better results than the realistic, so the data follows a one-set-behind model closer than a long term one.

### DS2

As expected really good results  $R^2$  for the optimistic model (low MAPE error and almost perfect  $R^2$ )

Assumes the future will be the same as the present

(The optimistic persistence model showed way better results than the realistic one and we can see that the realistic one only predicted a line)

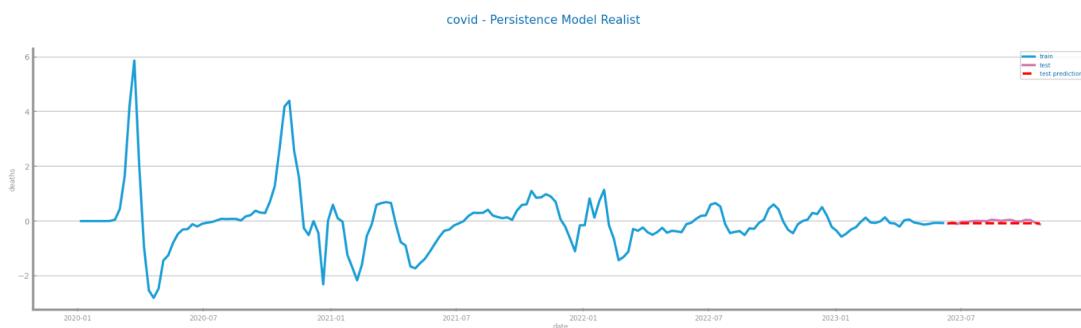
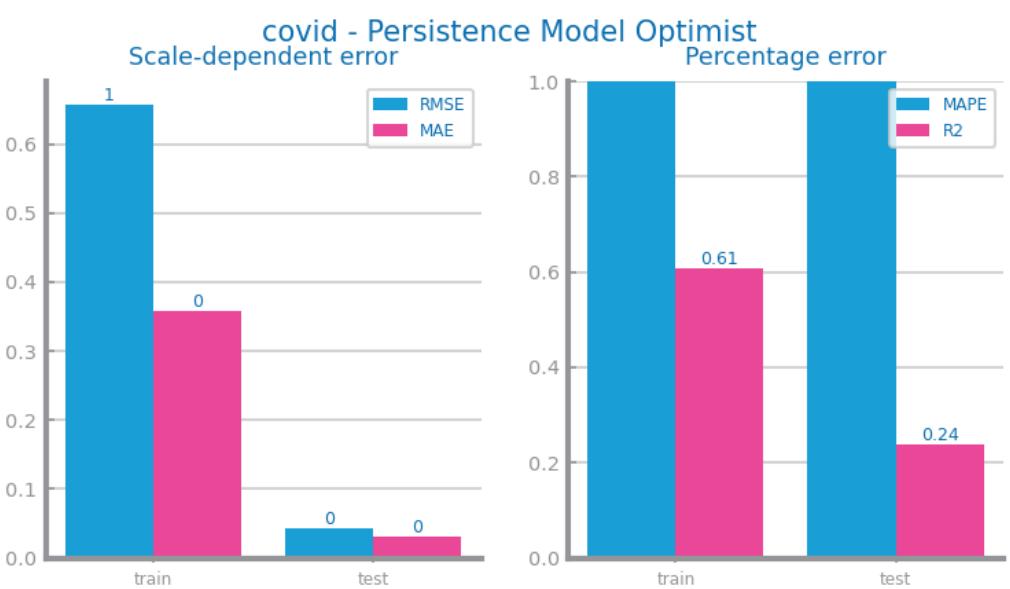
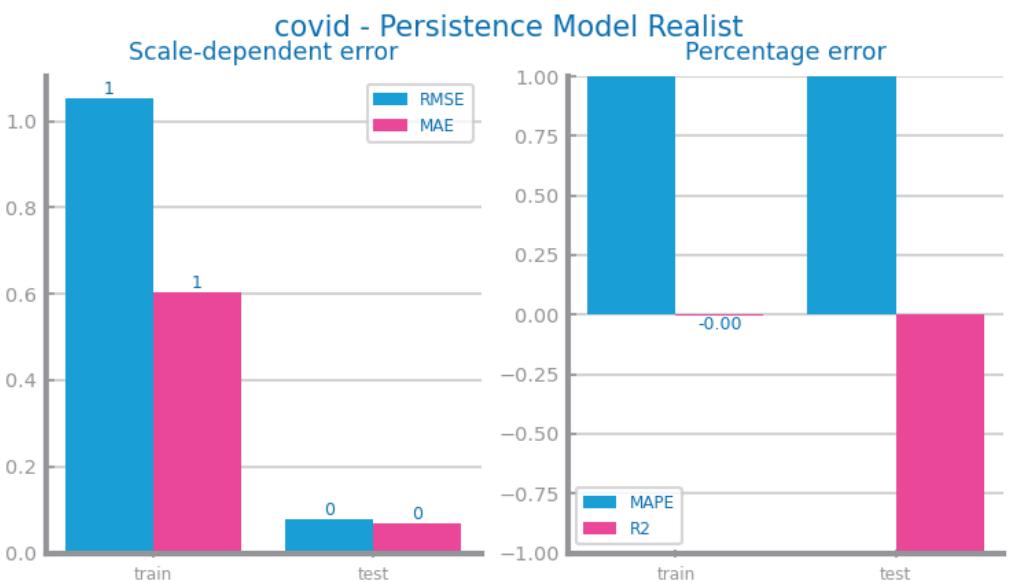


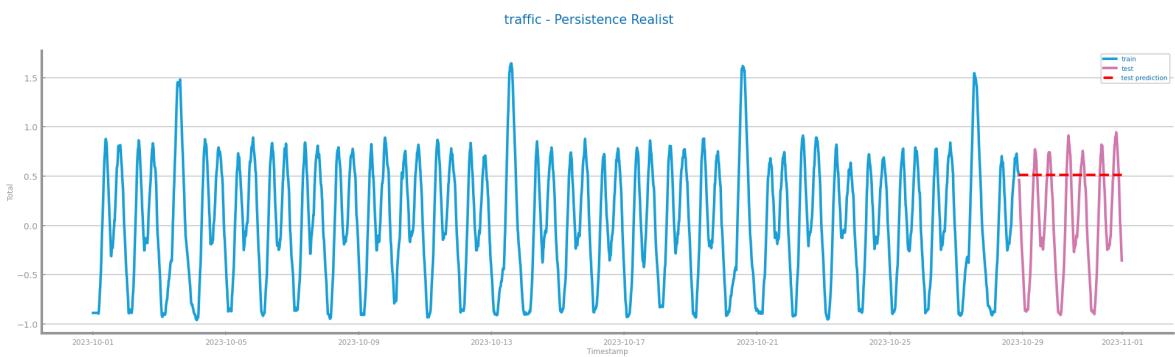
Figure 88 Forecasting plots obtained with Persistence model (long term) over time series 1



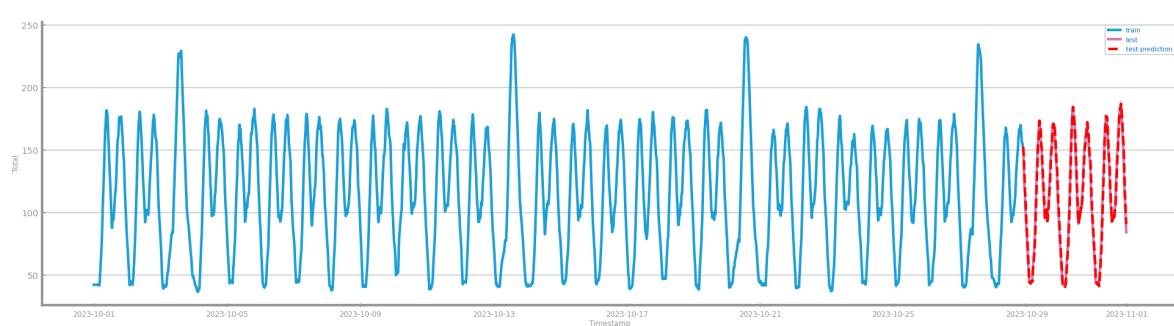
Figure 89 Forecasting plots obtained with Persistence model (one-set-behind) over time series 1



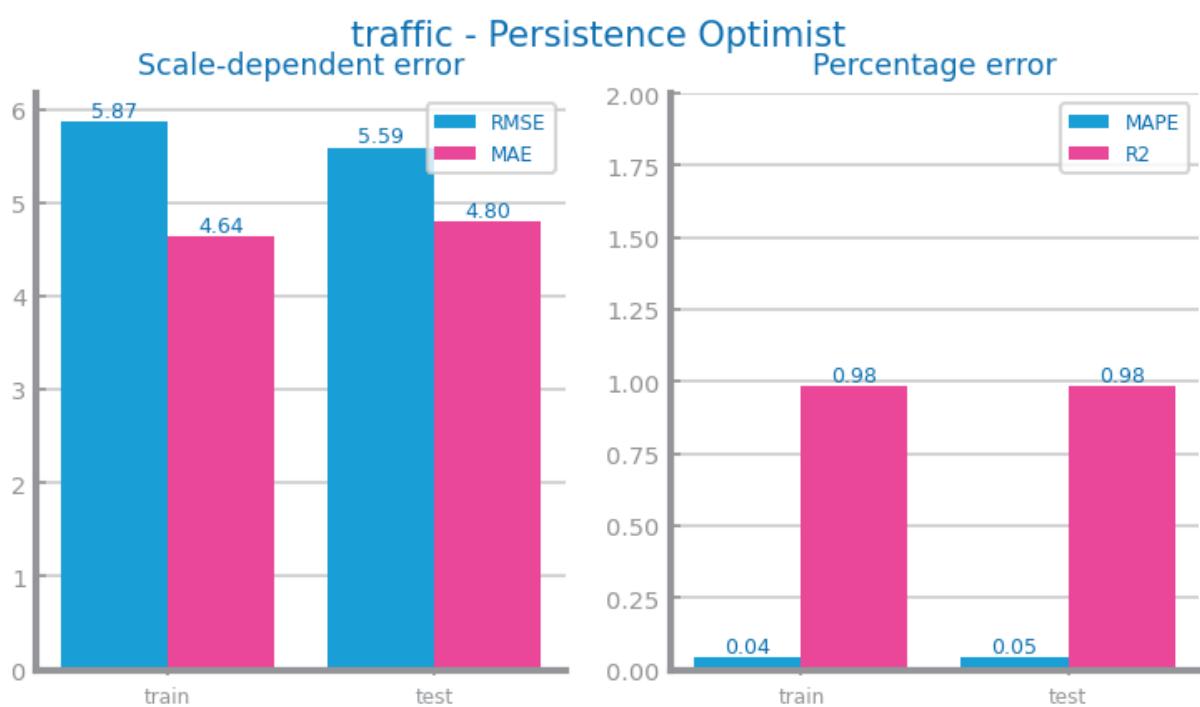
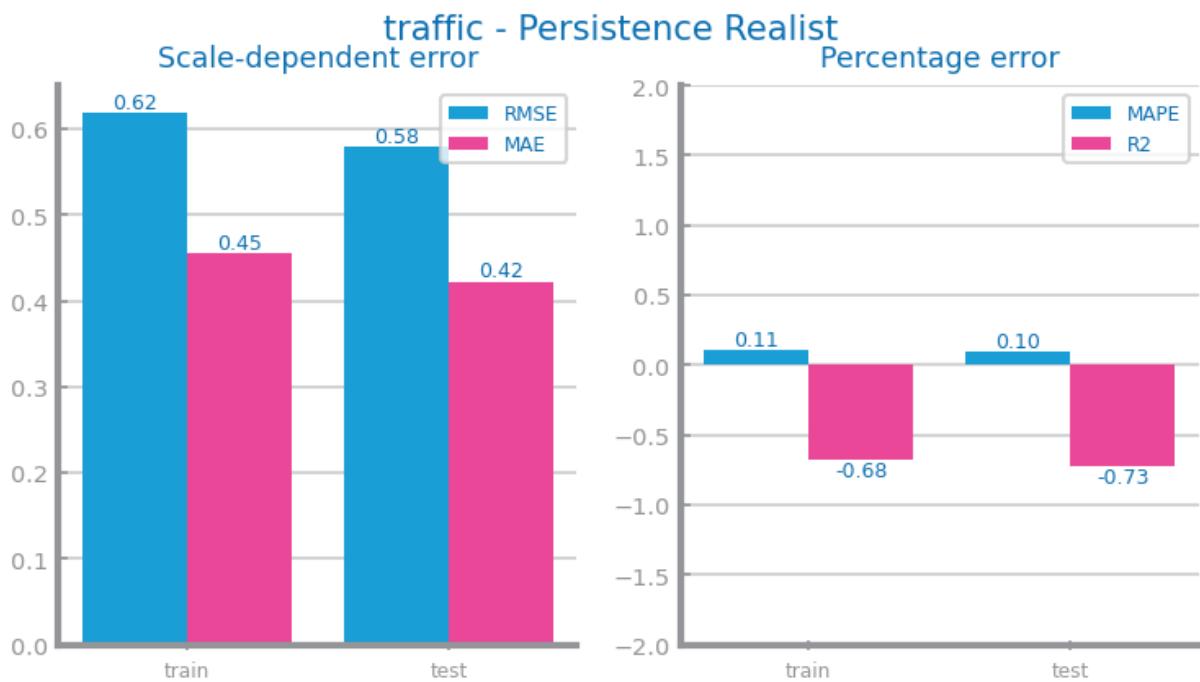
*Figure 86 Forecasting results obtained with Persistence model in both situations over time series 1*



*Figure 88 Forecasting plots obtained with Persistence model (long term) over time series 2*



*Figure 89 Forecasting plots obtained with Persistence model (one-set-behind) over time series 2*



*Figure 86 Forecasting results obtained with Persistence model in both situation over time series 2*

## **Rolling Mean Model**

### **DS1**

WIN-SIZE from 1 to 179

Displayed a cyclic pattern in  $R^2$  values, overall performance was not optimal

### **DS2**

Adjustable parameter WIN\_SIZE

**81** as best; diff sizes were tests; start with upward pattern and then stabilize for  $R^2$

Despite low MAPE suggesting high predictive accuracy,  $R^2$  of **0** indicates the model failed to account variability in the data

Mean-based approaches are unsuitable here, as the technique compute its mean value for predicting the next one

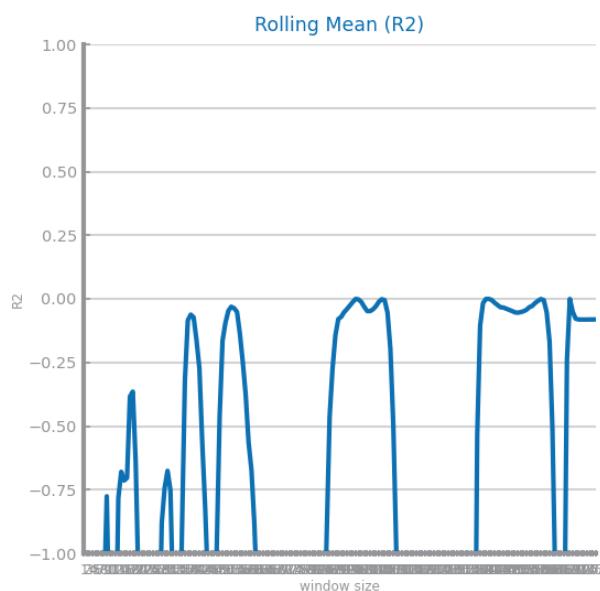


Figure 90 Forecasting study over different parameterisations of the rolling mean algorithm over time series 1



Figure 91 Forecasting plots obtained with the best parameterisation of rolling mean algorithm, over time series 1

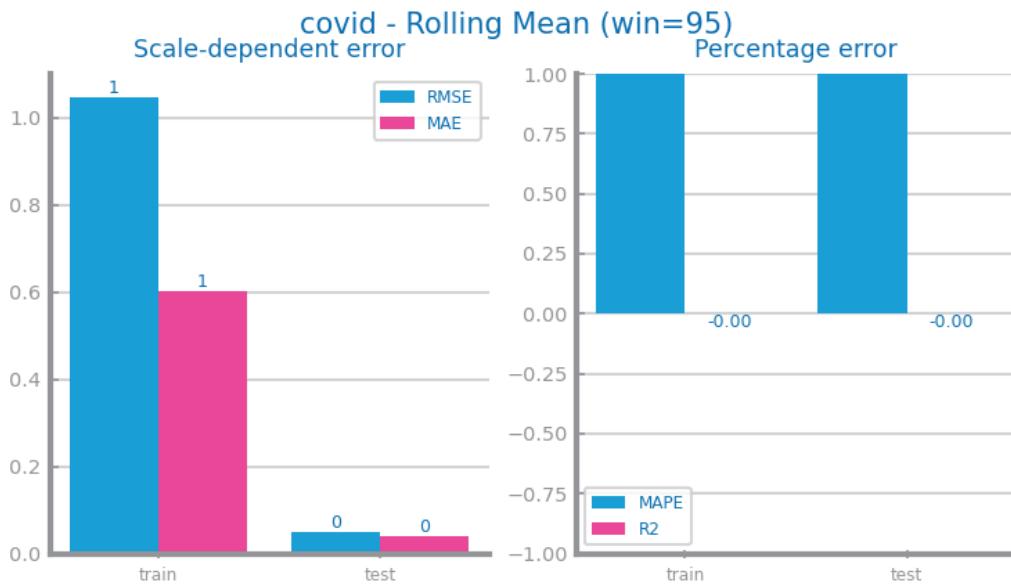


Figure 92 Forecasting results obtained with the best parameterisation of rolling mean algorithm, over time series 1

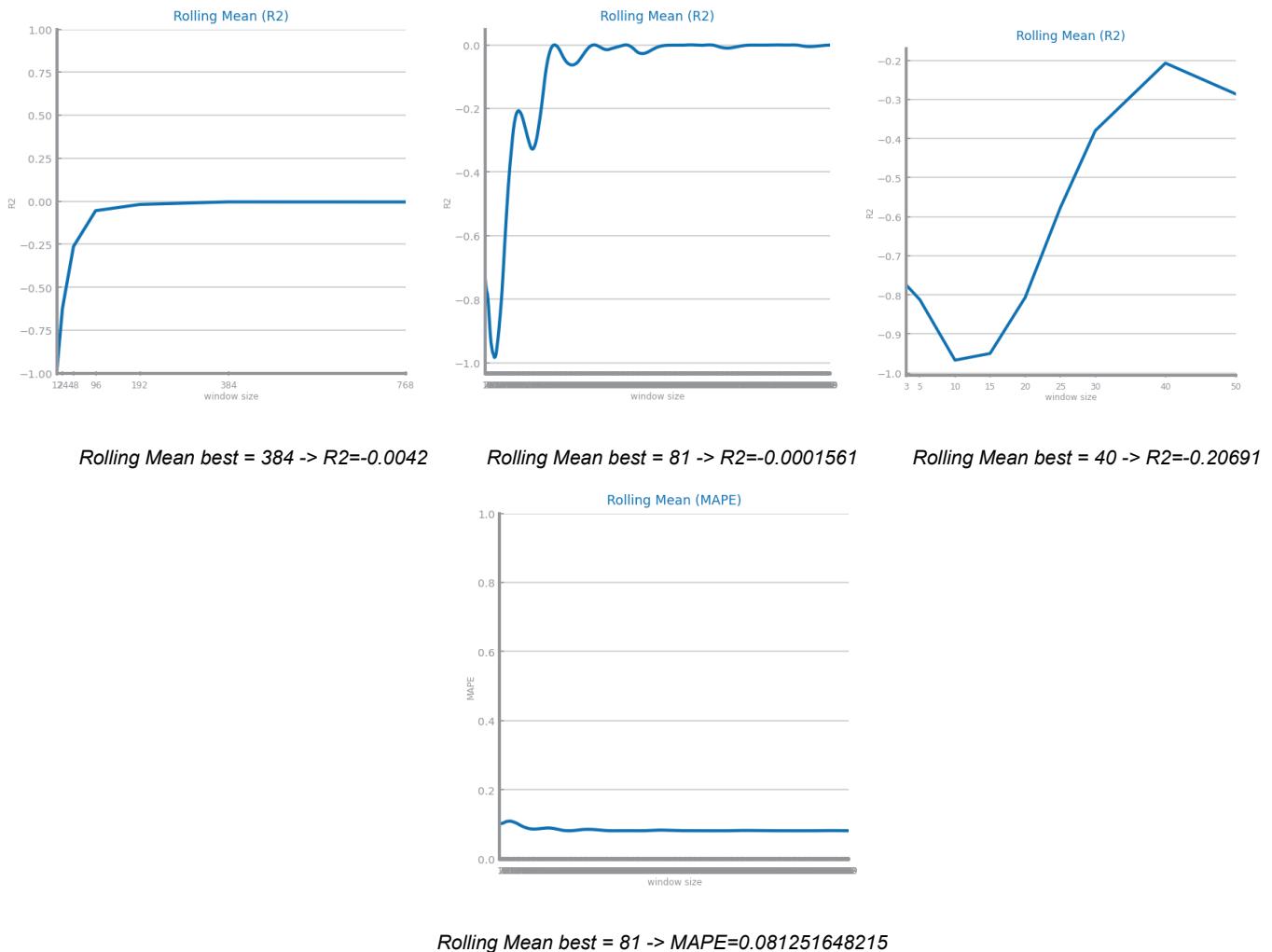
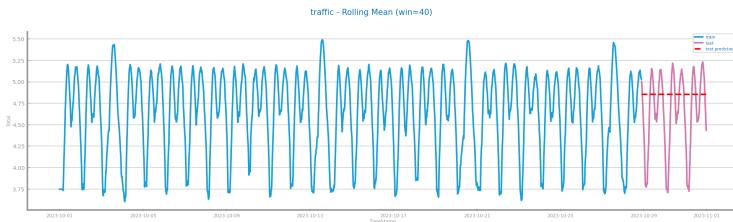
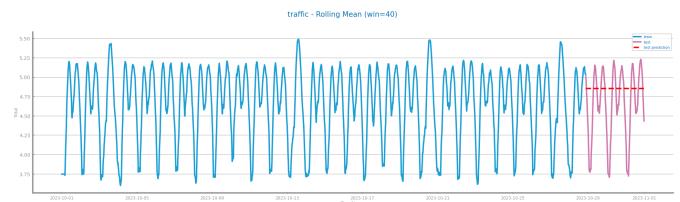
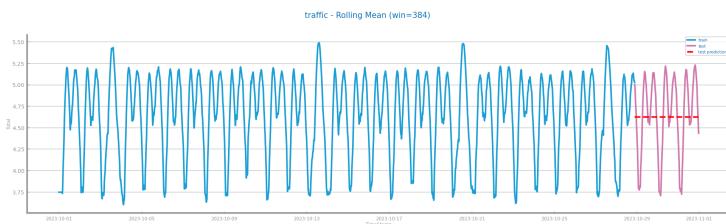


Figure 93 Forecasting study over different parameterisations of the rolling mean algorithm over time series 2



384

81

40

Figure 94 Forecasting plots obtained with the best parameterisation of rolling mean algorithm, over time series 2

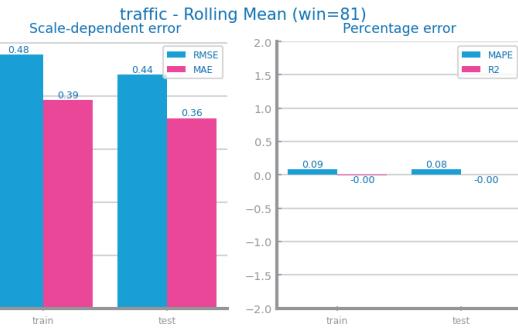
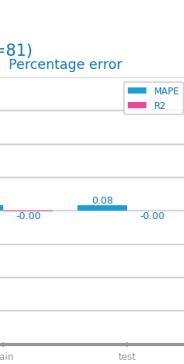
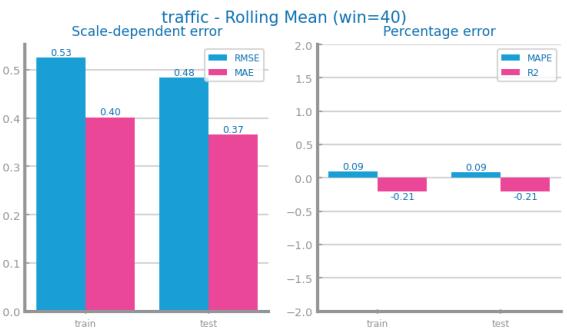
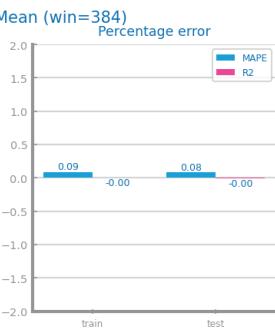


Figure 95 Forecasting results obtained with the best parameterisation of rolling mean algorithm, over time series 2

## ARIMA Model

### DS1

Best parameters w/ low d values indicate stationarity: d [0,1,2], p [1,2,3,5,7,10], q [1,3,5,7]

### DS2

Study (2,8,4):

**Std residuals** have no pattern, structure is well captured *assumptions are good*

**Q-Q plot**, almost all points in reference line, residuals normal distributed

no White Noise, correlations present

Despite study good feedback, **best model (2,0,7)** was short, perfect R2 train but significant MAPE error & low R2 test; Expected better due to Stationary

Plot test prediction doesn't follow the test

===== Dep. Variable: deaths No. Observations: 179 Model: ARIMA(3, 1, 2) Log Likelihood -146.766 Date: Wed, 03 Jan 2024 AIC 305.532 Time: 17:07:57 BIC 324.622 Sample: 01-06-2020 HQIC 313.273 Covariance Type: opg =====						
ar.L1	0.2143	0.124	1.733	0.083	-0.028	0.457
ar.L2	0.6932	0.131	5.309	0.000	0.437	0.949
ar.L3	-0.4819	0.069	-7.002	0.000	-0.617	-0.347
ma.L1	-0.0228	5.171	-0.004	0.996	-10.157	10.111
ma.L2	-0.9768	5.062	-0.195	0.845	-10.781	8.828
sigma2	0.2973	1.523	0.195	0.845	-2.687	3.282
Ljung-Box (L1) (Q):	0.02	Jarque-Bera (JB):	787.04			
Prob(Q):	0.88	Prob(JB):	0.00			
Heteroskedasticity (H):	0.04	Skew:	-0.31			
Prob(H) (two-sided):	0.00	Kurtosis:	13.28			

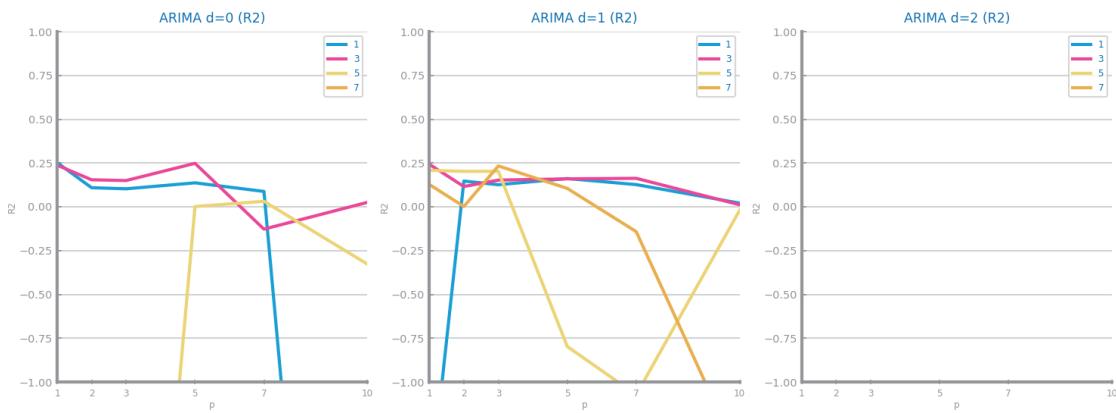
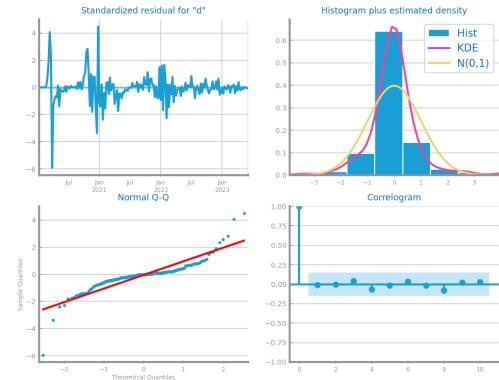


Figure 96 Forecasting study over different parameterisations of the ARIMA algorithm over time series 1

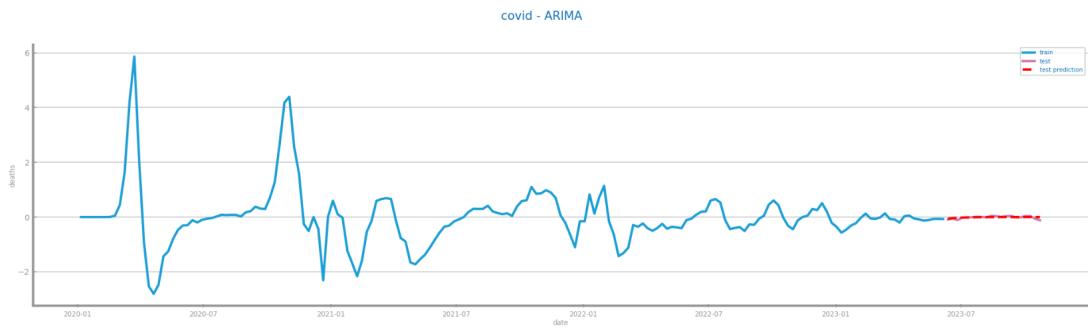


Figure 97 Forecasting plots obtained with the best parameterisation of ARIMA algorithm, over time series 1

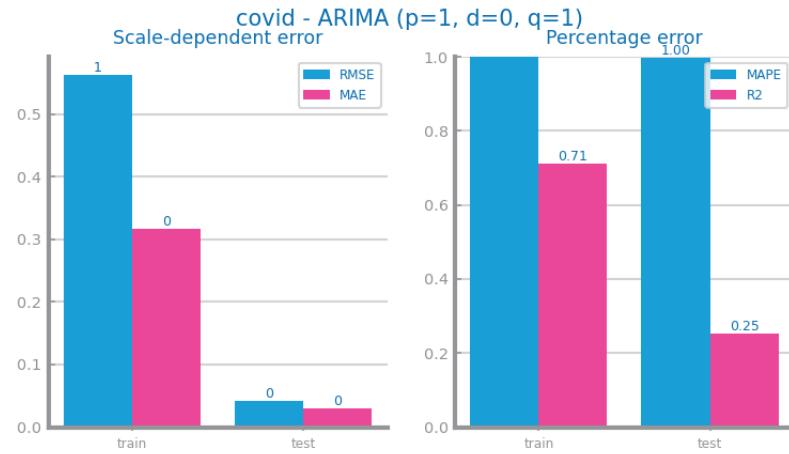
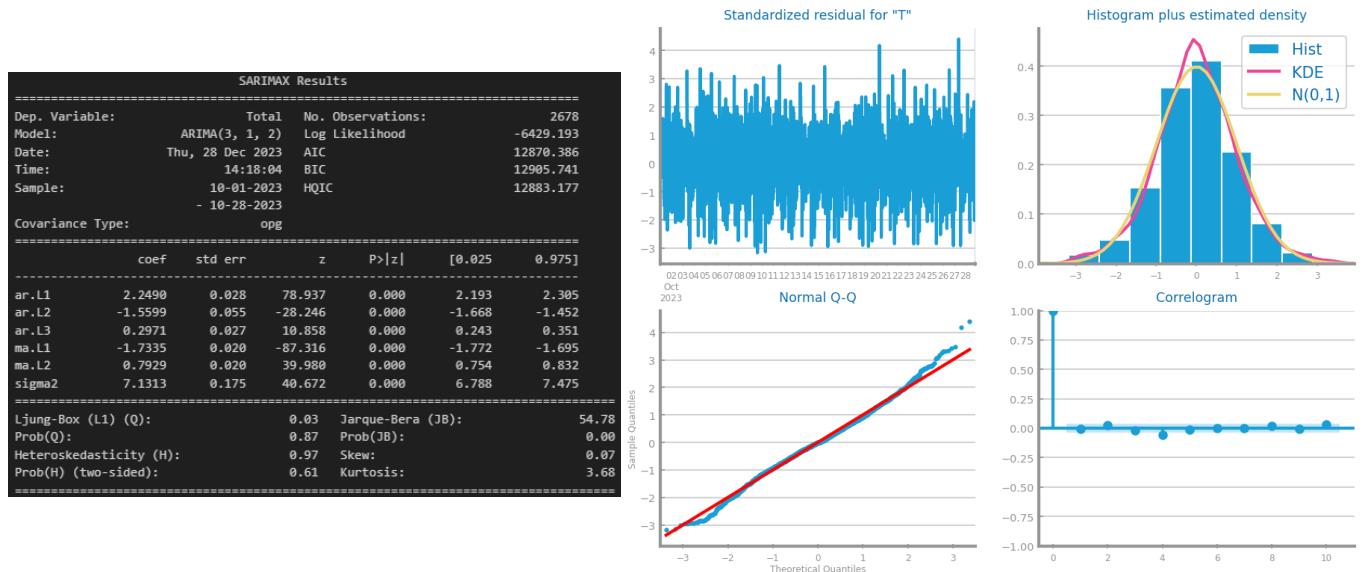


Figure 98 Forecasting results obtained with the best parameterisation of ARIMA algorithm, over time series 1

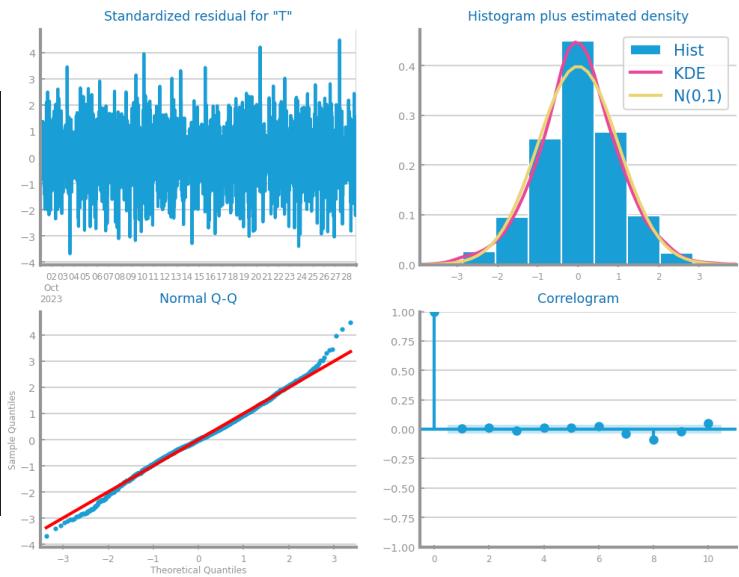


```

SARIMAX Results
=====
Dep. Variable: Total No. Observations: 2678
Model: ARIMA(4, 1, 9) Log Likelihood -6330.456
Date: Sat, 30 Dec 2023 AIC 12688.911
Time: 12:32:15 BIC 12771.405
Sample: 10-01-2023 HQIC 12718.756
Covariance Type: opg

coef std err z P>|z| [0.025 0.975]
ar.L1 0.9561 0.045 21.252 0.000 0.868 1.044
ar.L2 -0.3674 0.042 -8.733 0.000 -0.450 -0.285
ar.L3 0.8198 0.036 22.742 0.000 0.749 0.890
ar.L4 -0.5619 0.026 -21.905 0.000 -0.612 -0.512
ma.L1 -0.4335 0.049 -8.759 0.000 -0.531 -0.337
ma.L2 0.3149 0.036 8.729 0.000 0.244 0.386
ma.L3 -0.7145 0.033 -21.874 0.000 -0.779 -0.651
ma.L4 0.3102 0.029 10.812 0.000 0.254 0.366
ma.L5 0.1866 0.026 7.255 0.000 0.136 0.237
ma.L6 0.0764 0.027 2.839 0.005 0.024 0.129
ma.L7 0.1986 0.027 7.244 0.000 0.145 0.252
ma.L8 0.2098 0.027 7.789 0.000 0.157 0.263
ma.L9 0.0320 0.034 0.954 0.340 -0.034 0.098
...

```



```

SARIMAX Results
=====
Dep. Variable: Total No. Observations: 2678
Model: ARIMA(2, 8, 4) Log Likelihood -9906.263
Date: Sat, 30 Dec 2023 AIC 19826.525
Time: 12:33:35 BIC 19867.754
Sample: 10-01-2023 HQIC 19841.443
Covariance Type: opg

coef std err z P>|z| [0.025 0.975]
ar.L1 -1.7628 0.012 -149.432 0.000 -1.786 -1.740
ar.L2 -0.7981 0.012 -67.738 0.000 -0.821 -0.775
ma.L1 -1.9970 7.991 -0.250 0.803 -17.660 13.666
ma.L2 -0.0011 8.000 -0.000 1.000 -15.681 15.679
ma.L3 1.9970 7.998 0.250 0.803 -13.679 17.673
ma.L4 -0.9989 7.989 -0.125 0.900 -16.656 14.659
sigma2 96.3951 771.000 0.125 0.901 -1414.736 1607.526

Ljung-Box (L1) (Q): 652.69 Jarque-Bera (JB): 27.71
Prob(Q): 0.00 Prob(JB): 0.00
Heteroskedasticity (H): 1.05 Skew: -0.01
Prob(H) (two-sided): 0.46 Kurtosis: 3.50

```

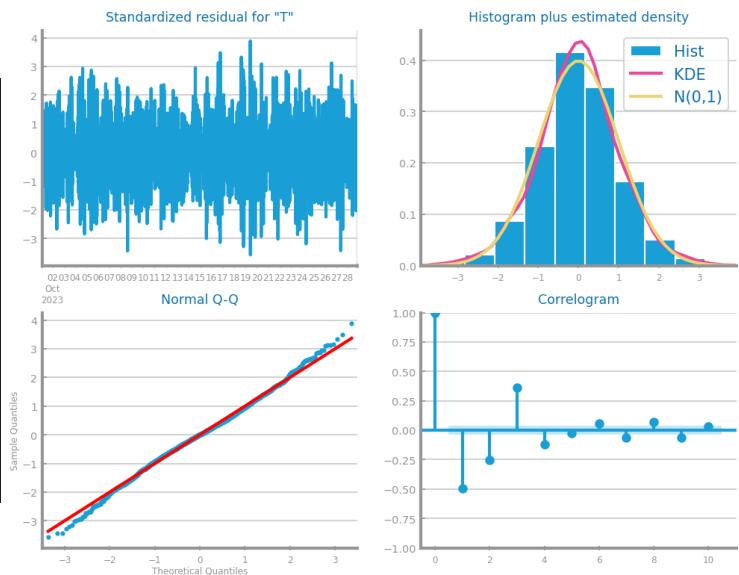
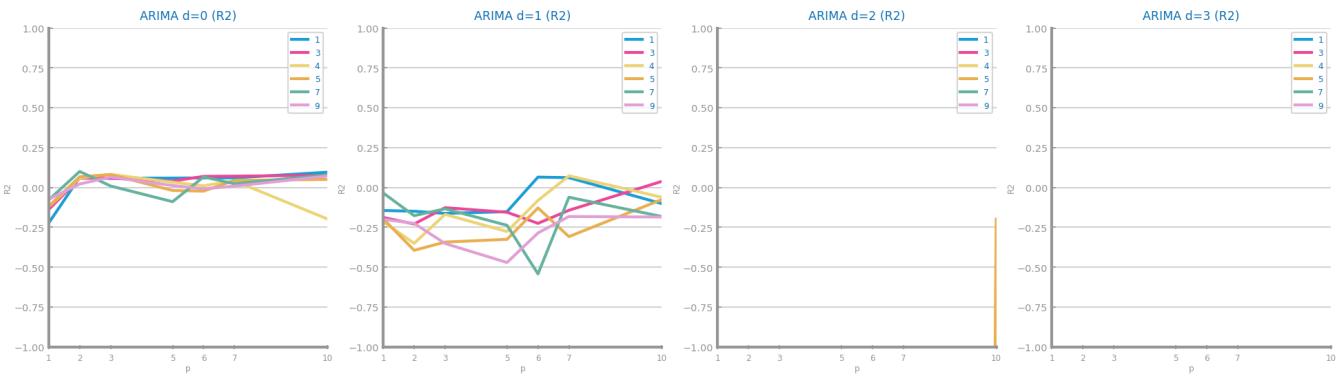


Figure 99 Forecasting study over different parameterisations of the ARIMA algorithm over time series 2



ARIMA best results achieved with  $(p,d,q)=(2, 0, 7) \Rightarrow \text{measure}=0.10$

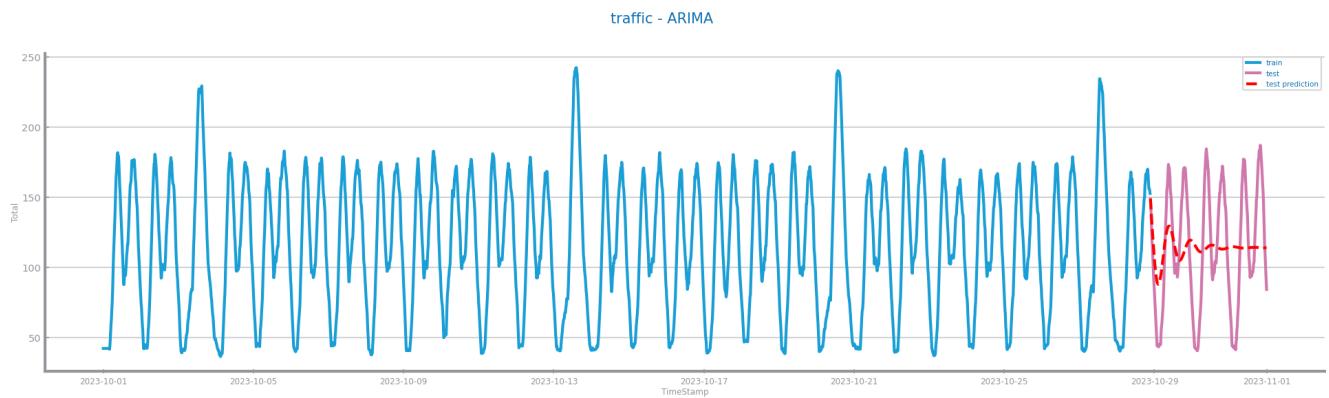


Figure 100 Forecasting plots obtained with the best parameterisation of ARIMA algorithm, over time series 2

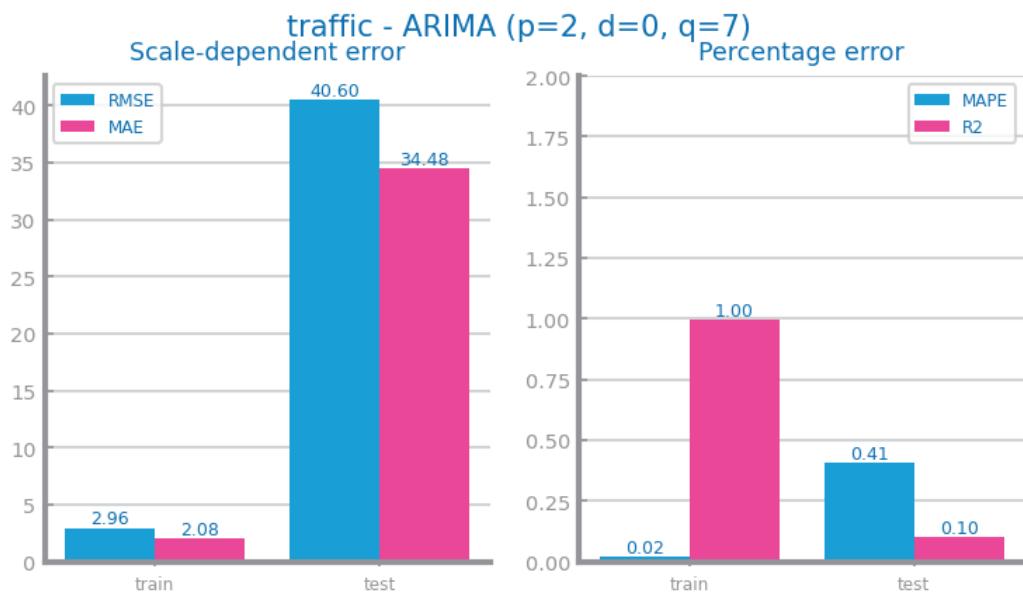


Figure 101 Forecasting results obtained with the best parameterisation of ARIMA algorithm, over time series 2

## LSTMs Model

### DS1

Use values in Fig102 for LSTM; Perform better in small sequence lengths due to data tapering off

### DS2

Best result w/ sequence <8, consistent  $R^2$  by capturing variability

Best model (**4, 25, 300**); Perfect  $R^2$  & low MAPE error in train & test; Scale-dependent errors practically **0**

Test predictions near 0 in plot, indicating it does not reflect pattern capture (*problem in Transformation or w/ the dataset*)

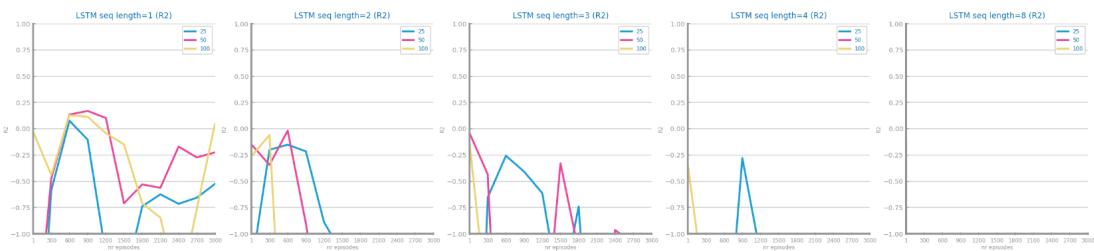


Figure 102 Forecasting study over different parameterisations of LSTMs over time series 1

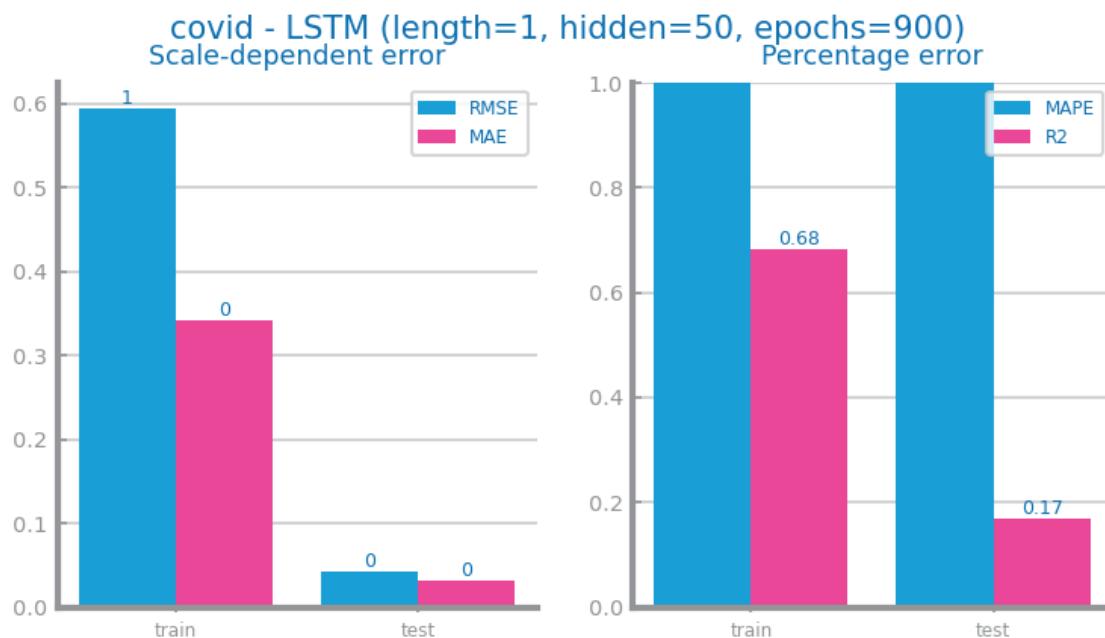


Figure 103 Forecasting plots obtained with the best parameterisation of LSTMs, over time series 1



Figure 104 Forecasting results obtained with the best parameterisation of LSTMs, over time series 1

`tensor(18.4214, grad_fn=<MseLossBackward0>)`

LSTM best results achieved with length=4 hidden units=25 and nr episodes=300) ==> measure=1.00

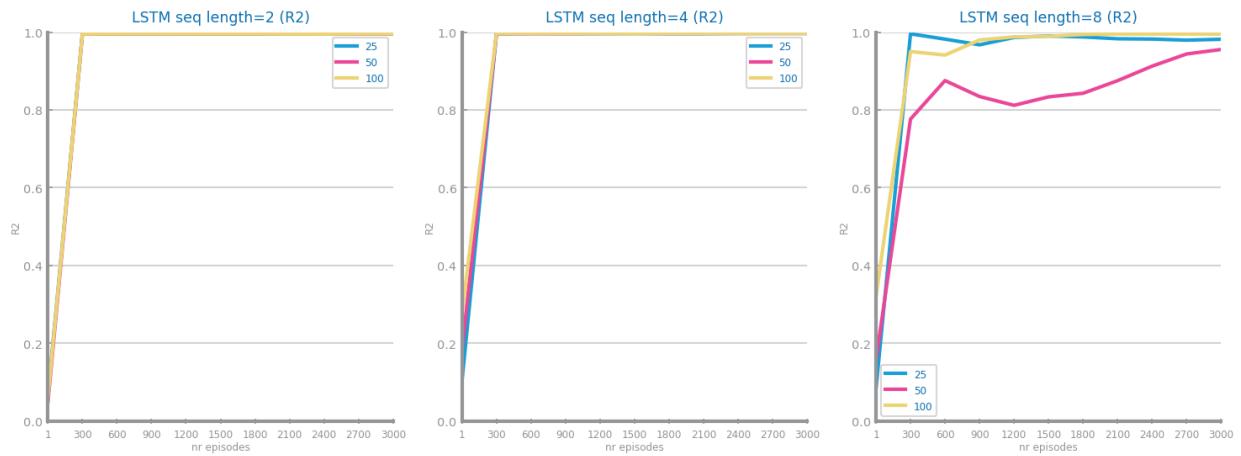
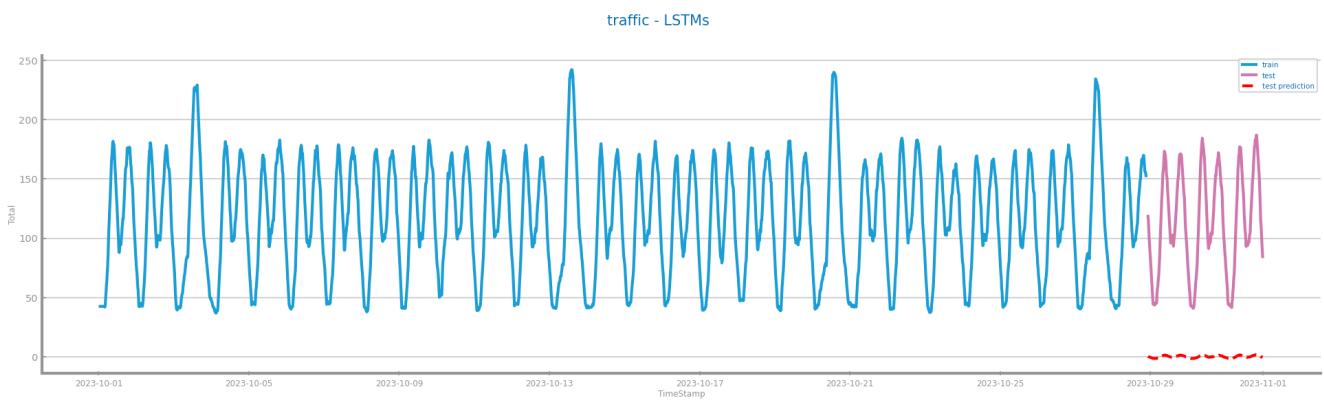


Figure 105 Forecasting study over different parameterisations - length, hidden\_units, nr\_episodes - of the LSTMs over time series 2

Figure 106 Forecasting plots obtained with the best parameterisation of LSTMs, over time series 2



Figure 107 Forecasting results obtained with the best parameterisation of LSTMs, over time series 2



## 8 CRITICAL ANALYSIS

### DS1

From the data profiling done on this data set we can infer that the data isn't stationary however its upwards trend is really fast in the first points of the data and then slows down considerably, we also inferred this by the differentiation done on the transformation section where either the first or the second derivative had really intense activity on the first half of the series, and settles down in the later portion. This drop off in the death-rates (first derivative) and transmission-rate(second derivative\*some mortality function) could be justified by the vaccination of the population reaching a point of herd immunity, but there wouldn't be any way to predict this with a univariate time series, like the one we have, and that explains the poor performance of our models since the series takes such an unpredictable turn this is further corroborated by the fact that the best LSTMs, ARIMA and persistence models were the ones where they had a small seasonality/memory function.

### DS2

From *Profiling*, it's possible to conclude the Series is **Stationary**, w/ a constant mean & high variability

A more granular approach would mean the loss of information

*SAM* and *Linear Regression* were not suited, due the high **variability** around the mean, so none performed well

The *Persistence Model* excelled for *Optimistic* projections, accurately predicting stable values (*value would be the same as the last* - true given Profiling), but faltered in *Realistic* scenarios, for near future, **due to instability**

*Rolling Mean* (*mean-based* problem) by using the mean value to predict next ones, given **the mean was constant**, as stated before, trying to predict almost the same values, resulting in a poor prediction and evaluation

Despite modest test outcomes, *ARIMA* stands as a robust option with minimal *MAPE* error, indicating reliable performance. Conversely, while *LSTM* demonstrates promising results, it encounters issues in prediction testing, warranting further investigation.