ML Raport

AutoPrep

January 13, 2025

Abstract

This raport has been generated with AutoPrep.

Contents

1 Overview

1.1 System

System	Darwin
Machine	arm64
Processor	arm
Architecture	64bit
Python Version	3.10.5
Physical Cores	8
Logical Cores	8
CPU Frequency (MHz)	3204
Total RAM (GB)	16.0000
Available RAM (GB)	6.0100
Total Disk Space (GB)	228.2700
Free Disk Space (GB)	13.0500

Table 1: System overview.

1.2 Dataset

Task detected for the dataset: binary classfication.
Table 46 presents an overview of the dataset including the number of samples, features, and their types.

Number of samples	1047
Number of features	13
Number of numerical features	6
Number of categorical features	7

Table 2: Dataset Summary.

Distribution of the target classes in terms of the number of observations and their percentages is presented in Table 25

class	**number of observations**	**fraction**
0	665	0.6351
1	382	0.3649

Table 3: Target class distribution.

Table 47 presents the distribution of missing values in the dataset.

feature	number of observations	fraction
pclass	0	0.0000
name	0	0.0000
sex	0	0.0000
age	207	0.1977
sibsp	0	0.0000
parch	0	0.0000
ticket	0	0.0000
fare	1	0.0010
cabin	813	0.7765
embarked	1	0.0010
boat	672	0.6418
body	948	0.9054
home___dest	453	0.4327

Table 4: Missing values distribution.

Table 48 presents the description of features in the dataset.

feature	type	dtype	space usage
pclass	numerical	uint8	9.4 kB
name	categorical	object	96.4 kB
sex	categorical	category	9.7 kB
age	numerical	float64	16.8 kB
sibsp	numerical	uint8	9.4 kB
parch	numerical	uint8	9.4 kB
ticket	categorical	object	75.1 kB
fare	numerical	float64	16.8 kB
cabin	categorical	object	42.1 kB
embarked	categorical	category	9.7 kB
boat	categorical	object	46.4 kB
body	numerical	float64	16.8 kB
home___dest	categorical	object	64.5 kB

Table 5: Features dtypes description.

Table 49 and Table 29 present the description of numerical and categorical features in the dataset.

feature	count	mean	std	min	25%	50%	75%	max
pclass	1047.0000	2.2970	0.8369	1.0000	2.0000	3.0000	3.0000	3.0000
age	840.0000	29.5327	14.2658	0.1667	21.0000	28.0000	38.6250	80.0000
sibsp	1047.0000	0.5205	1.0500	0.0000	0.0000	0.0000	1.0000	8.0000
parch	1047.0000	0.3954	0.8942	0.0000	0.0000	0.0000	0.0000	9.0000
fare	1046.0000	33.5472	51.8097	0.0000	7.9250	14.5000	31.2750	512.3292
body	99.0000	160.8990	98.3519	1.0000	73.5000	156.0000	255.5000	328.0000

Table 6: Numerical features description.

index	count	unique	top	freq
name	1047	1046	Connolly, Miss. Kate	2
sex	1047	2	male	677
ticket	1047	773	CA. 2343	9
cabin	234	161	B57 B59 B63 B66	5
embarked	1046	3	S	737
boat	375	25	13	34
home___dest	594	317	New York, NY	50

Table 7: Categorical features description.

2 Eda

This part of the report provides basic insides to the data and the informations it holds..

2.1 Target variable and missing values

Figure 30 shows the distribution of the target variable.

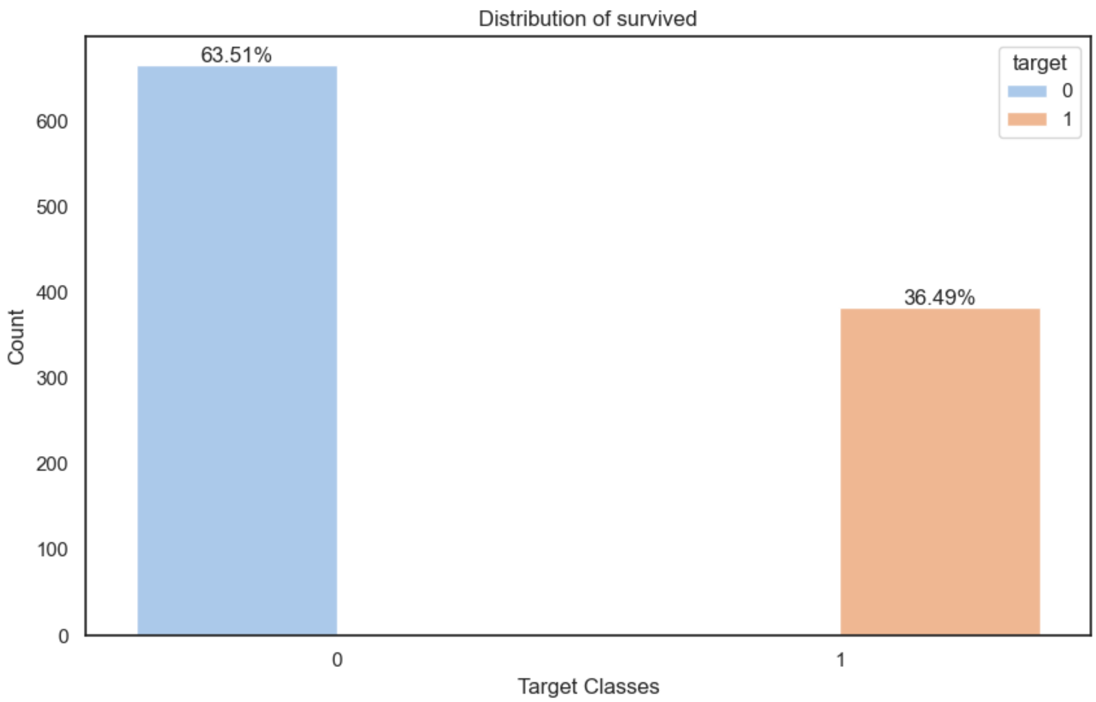

Figure 1: Target distribution.

Figure 2 shows the distribution of missing values in the dataset.

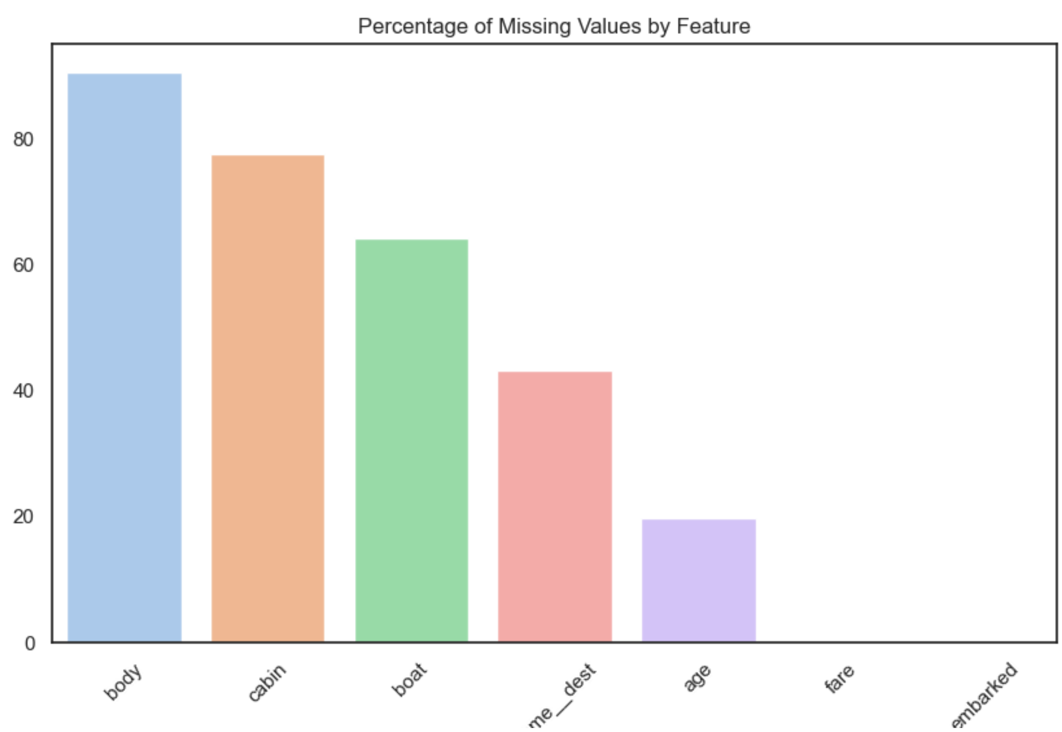

Figure 2: Missing values.

2.2 EDA for categorical features

The distribution of categorical features is presented on barplot(s) below.

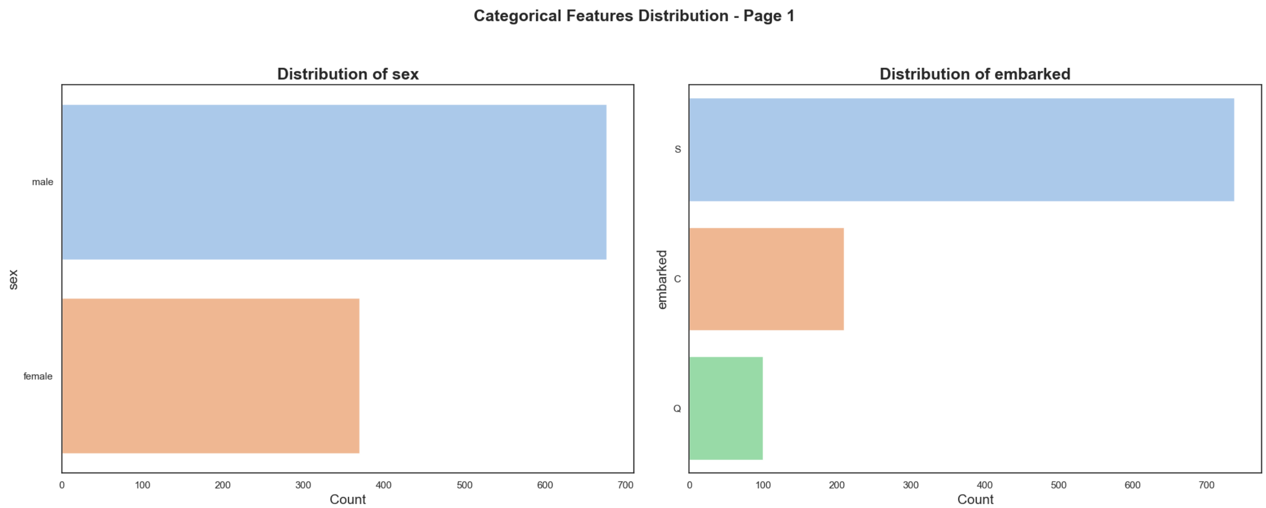

Figure 3: Categorical Features Distribution - Page 1

2.3 EDA for numerical features

The distribution of numerical features is presented on histogram(s) below.

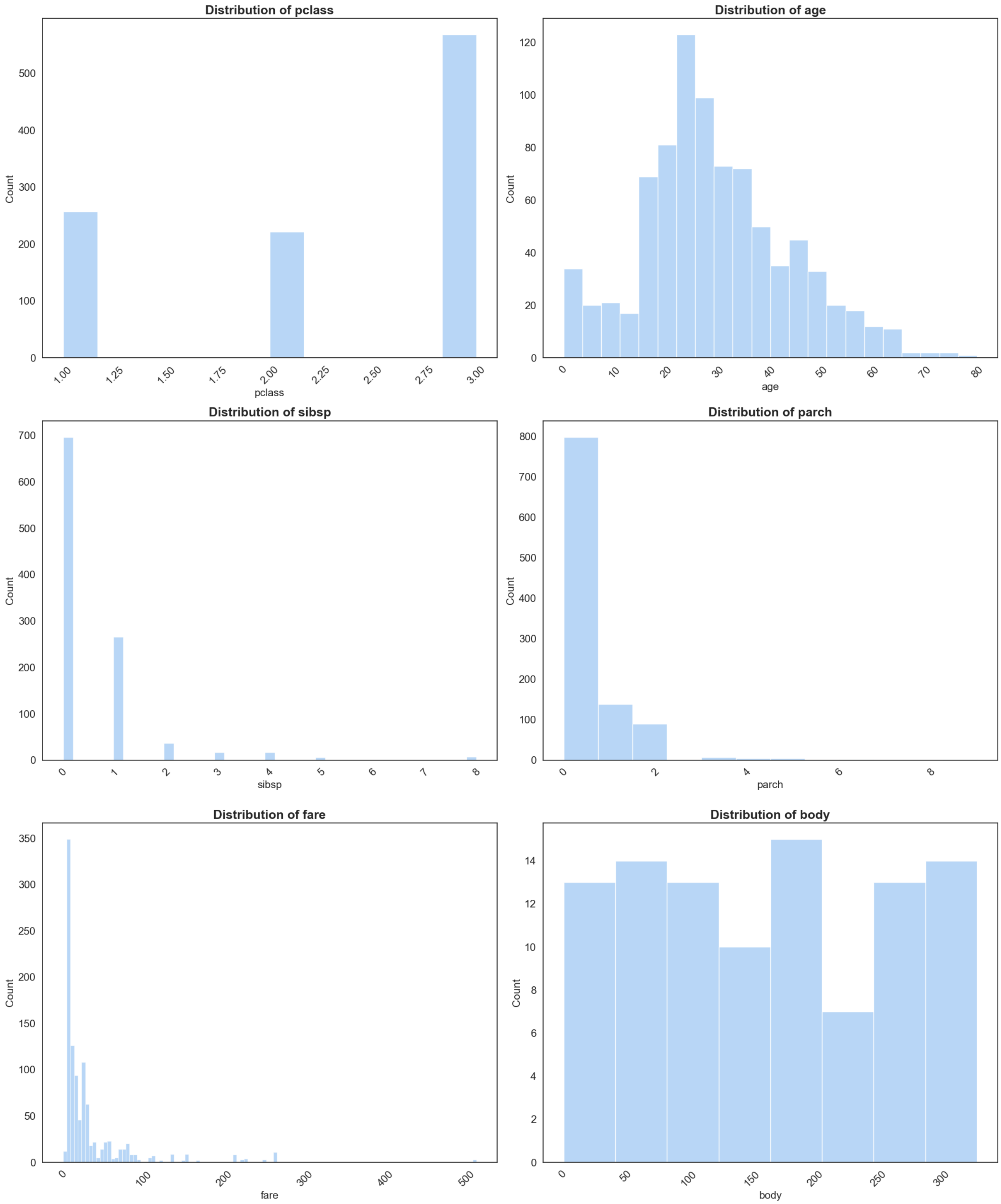

Figure 4: Numerical Features Distribution - Page 1

Figure 33 shows the correlation between features.

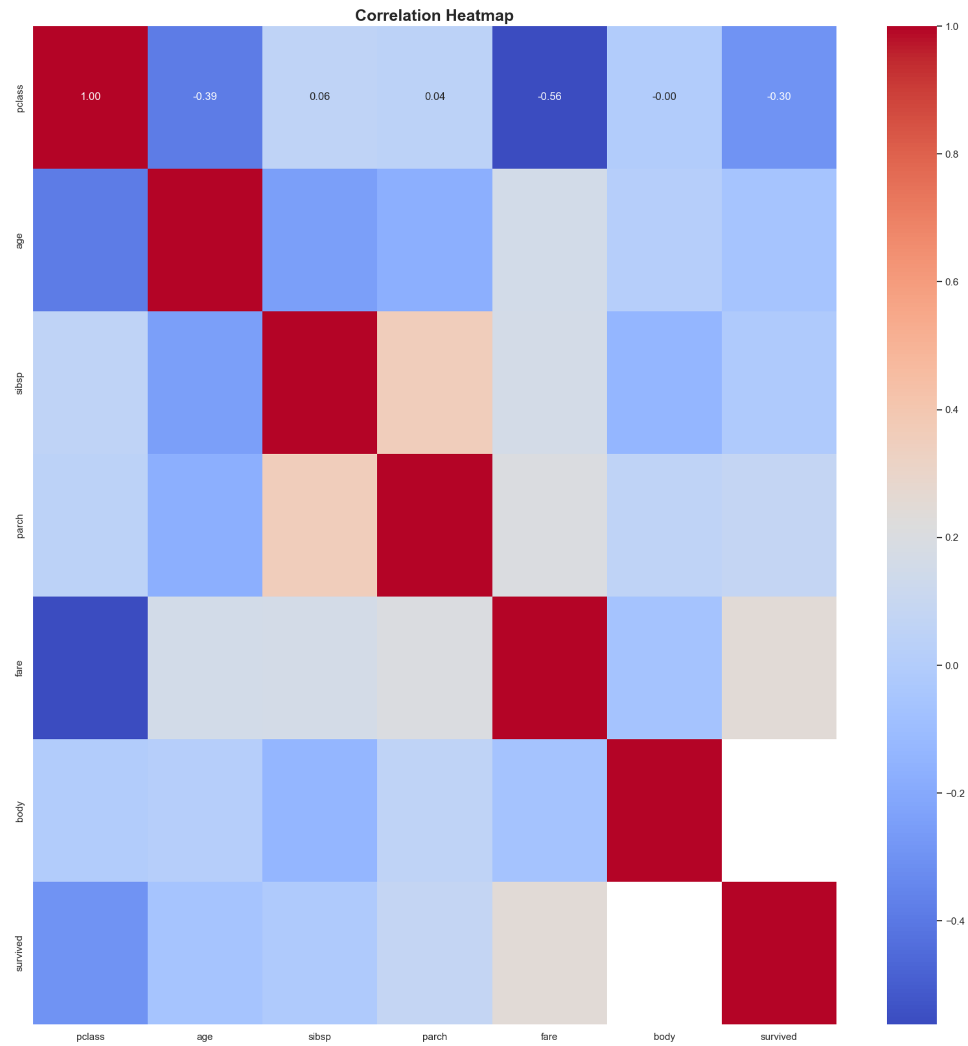

Figure 5: Correlation heatmap.

The boxplot of numerical features is presented on chart(s) below.

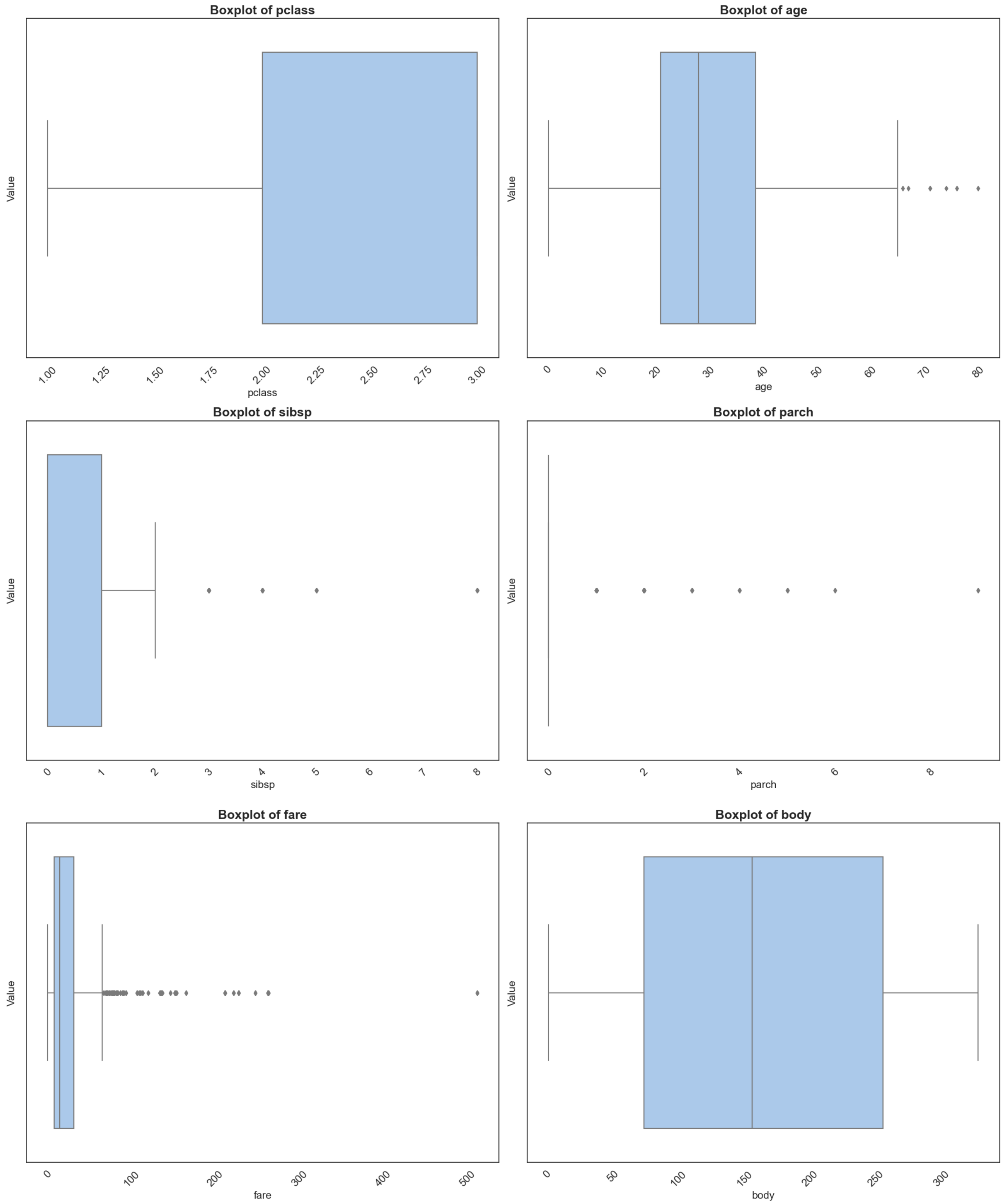

Figure 6: Boxplot page 1

3 Preprocessing

This part of the report presents the results of the preprocessing process. It contains required, as well as non required, steps listed below.

Required preprocessing steps:

- Missing data imputation

- Removing columns with 100% unique categorical values
- Categorical features encoding
- Scaling
- Removing columns with 0 variance
- Detecting highly correlatd features

Additional preprocessing steps:

- Feature selection methods : Correlation with the target or Random Forest feature importance
- Dimention reduction techniques: PCA, VIF, UMAP

Preprocessing process was configured to select up to 3 best unique preprocessing pipelines. Pipelines were scored based on a simple model. Tables below show detailed description of the best pipelines as well as all step combinations that were examined.

index	**steps**
0	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler
1	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, CorrelationSelector
2	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, FeatureImportanceRegressSelector
3	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, FeatureImportanceClassSelector
4	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, PCADimentionReducer
5	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, CorrelationSelector, PCADimentionReducer
6	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, FeatureImportanceRegressSelector, PCADimentionReducer
7	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, FeatureImportanceClassSelector, PCADimentionReducer
8	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, UMAPDimentionReducer
9	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, CorrelationSelector, UMAPDimentionReducer
10	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, FeatureImportanceRegressSelector, UMAPDimentionReducer
11	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, FeatureImportanceClassSelector, UMAPDimentionReducer
12	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, VIFDimentionReducer
13	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, CorrelationSelector, VIFDimentionReducer
14	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, FeatureImportanceRegressSelector, VIFDimentionReducer
15	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, FeatureImportanceClassSelector, VIFDimentionReducer

Table 8: Pipelines steps overview.

index	file name	score	fit duration	score duration
0	preprocessing_pipeline_0.joblib	0.7680	a moment	a moment
1	preprocessing_pipeline_1.joblib	0.7595	4 seconds	a moment
2	preprocessing_pipeline_2.joblib	0.7595	4 seconds	a moment

Table 9: Best preprocessing pipelines.

step	name	description	params
0	NAImputer	Imputes missing data.	{"numeric_imputer": "median", "categorical_imputer": "most_frequent"}
1	UniqueFilter	Removes categorical columns with 100% unique values. Dropped columns: []	{}
2	ColumnEncoder	Encodes categorical columns using OneHotEncoder (for columns with <5 unique values) or TolerantLabelEncoder (for columns with >=5 unique values). Encodes target variable using LabelEncoder if provided.	{}
3	VarianceFilter	Removes columns with zero variance. Dropped columns: []	{}
4	CorrelationFilter	Removes one column from pairs of columns correlated above correlation threshold: 0.8.	{}
5	ColumnScaler	Scales numerical columns using one of 3 scaling methods.	{"method": "standard"}
6	CorrelationSelector	Selects the top 70.0% (rounded to whole number) of features most correlated with the target variable. Number of features that were selected: 0	{"correlation_percent": 0.7}
7	PCADimentionReducer	Combines PCA with automatic selection of the number of components to preserve 95% of the variance.	{"n_components": null}

Table 10: Best pipeline No. 0: steps overview.

index	count	mean	std	min	25%	50%	75%	max
pclass	1047.0000	0.0000	1.0005	-1.5506	-0.3551	0.8404	0.8404	0.8404
name	1047.0000	0.0000	1.0005	-1.7293	-0.8667	-0.0007	0.8653	1.7313
age	1047.0000	-0.0000	1.0005	-2.2732	-0.5655	-0.0962	0.4513	3.9711
sibsp	1047.0000	-0.0000	1.0005	-0.4960	-0.4960	-0.4960	0.4568	7.1264
parch	1047.0000	0.0000	1.0005	-0.4424	-0.4424	-0.4424	-0.4424	9.6277
ticket	1047.0000	-0.0000	1.0005	-1.6829	-0.8990	0.0021	0.9336	1.6697
fare	1047.0000	0.0000	1.0005	-0.6477	-0.4946	-0.3676	-0.0435	9.2498
home___dest	1047.0000	-0.0000	1.0005	-2.7245	-0.1840	0.2345	0.3017	2.0128
sex_female	1047.0000	0.0000	1.0005	-0.7393	-0.7393	-0.7393	1.3527	1.3527
embarked_C	1047.0000	-0.0000	1.0005	-0.4994	-0.4994	-0.4994	-0.4994	2.0024
embarked_Q	1047.0000	0.0000	1.0005	-0.3250	-0.3250	-0.3250	-0.3250	3.0773
embarked_S	1047.0000	-0.0000	1.0005	-1.5454	-1.5454	0.6471	0.6471	0.6471

Table 11: Best pipeline No. 0: output overview.

step	name	description	params
0	NAImputer	Imputes missing data.	{"numeric_imputer": "median", "categorical_imputer": "most_frequent"}
1	UniqueFilter	Removes categorical columns with 100% unique values. Dropped columns: []	{}
2	ColumnEncoder	Encodes categorical columns using OneHotEncoder (for columns with <5 unique values) or TolerantLabelEncoder (for columns with >=5 unique values). Encodes target variable using LabelEncoder if provided.	{}
3	VarianceFilter	Removes columns with zero variance. Dropped columns: []	{}
4	CorrelationFilter	Removes one column from pairs of columns correlated above correlation threshold: 0.8.	{}
5	ColumnScaler	Scales numerical columns using one of 3 scaling methods.	{"method": "standard"}
6	FeatureImportanceClassSelector	Selects the top 10.0% (rounded to whole number) of features most important according to Random Forest model for classification. Number of features that were selected: 0	{"k": 10.0}
7	UMAPDimentionReducer	Reduces the dimensionality of the data using UMAP.	{"n_components": null}

Table 12: Best pipeline No. 1: steps overview.

index	count	mean	std	min	25%	50%	75%	max
pclass	1047.0000	0.6485	0.4185	0.0000	0.5000	1.0000	1.0000	1.0000
name	1047.0000	0.4997	0.2891	0.0000	0.2493	0.4995	0.7498	1.0000
age	1047.0000	0.3640	0.1602	0.0000	0.2735	0.3486	0.4363	1.0000
sibsp	1047.0000	0.0651	0.1313	0.0000	0.0000	0.0000	0.1250	1.0000
parch	1047.0000	0.0439	0.0994	0.0000	0.0000	0.0000	0.0000	1.0000
ticket	1047.0000	0.5020	0.2984	0.0000	0.2338	0.5026	0.7804	1.0000
fare	1047.0000	0.0654	0.1011	0.0000	0.0155	0.0283	0.0610	1.0000
home___dest	1047.0000	0.5751	0.2112	0.0000	0.5363	0.6246	0.6388	1.0000
sex_female	1047.0000	0.3534	0.4783	0.0000	0.0000	0.0000	1.0000	1.0000
embarked_C	1047.0000	0.1996	0.3999	0.0000	0.0000	0.0000	0.0000	1.0000
embarked_Q	1047.0000	0.0955	0.2941	0.0000	0.0000	0.0000	0.0000	1.0000
embarked_S	1047.0000	0.7049	0.4563	0.0000	0.0000	1.0000	1.0000	1.0000

Table 13: Best pipeline No. 1: output overview.

step	name	description	params
0	NAImputer	Imputes missing data.	{"numeric_imputer": "median", "categorical_imputer": "most_frequent"}
1	UniqueFilter	Removes categorical columns with 100% unique values. Dropped columns: []	{}
2	ColumnEncoder	Encodes categorical columns using OneHotEncoder (for columns with <5 unique values) or TolerantLabelEncoder (for columns with >=5 unique values). Encodes target variable using LabelEncoder if provided.	{}
3	VarianceFilter	Removes columns with zero variance. Dropped columns: []	{}
4	CorrelationFilter	Removes one column from pairs of columns correlated above correlation threshold: 0.8.	{}
5	ColumnScaler	Scales numerical columns using one of 3 scaling methods.	{"method": "robust"}
6	FeatureImportanceClassSelector	Selects the top 10.0% (rounded to whole number) of features most important according to Random Forest model for classification. Number of features that were selected: 0	{"k": 10.0}
7	UMAPDimentionReducer	Reduces the dimensionality of the data using UMAP.	{"n_components": null}

Table 14: Best pipeline No. 2: steps overview.

index	count	mean	std	min	25%	50%	75%	max
pclass	1047.0000	-0.7030	0.8369	-2.0000	-1.0000	0.0000	0.0000	0.0000
name	1047.0000	0.0004	0.5776	-0.9981	-0.5000	0.0000	0.5000	1.0000
age	1047.0000	0.0946	0.9839	-2.1410	-0.4615	0.0000	0.5385	4.0000
sibsp	1047.0000	0.5205	1.0500	0.0000	0.0000	0.0000	1.0000	8.0000
parch	1047.0000	0.3954	0.8942	0.0000	0.0000	0.0000	0.0000	9.0000
ticket	1047.0000	-0.0011	0.5459	-0.9194	-0.4917	0.0000	0.5083	0.9100
fare	1047.0000	0.8149	2.2179	-0.6210	-0.2816	0.0000	0.7184	21.3203
home___dest	1047.0000	-0.4828	2.0599	-6.0923	-0.8615	0.0000	0.1385	3.6615
sex_female	1047.0000	0.3534	0.4783	0.0000	0.0000	0.0000	1.0000	1.0000
embarked_C	1047.0000	0.1996	0.3999	0.0000	0.0000	0.0000	0.0000	1.0000
embarked_Q	1047.0000	0.0955	0.2941	0.0000	0.0000	0.0000	0.0000	1.0000
embarked_S	1047.0000	-0.2951	0.4563	-1.0000	-1.0000	0.0000	0.0000	0.0000

Table 15: Best pipeline No. 2: output overview.

Category	Value
Unique created pipelines	16
All created pipelines (after exploading each step params)	48
All pipelines fit time	23 seconds
All pipelines score time	20 seconds
scores_count	48.0000
scores_mean	0.7352
scores_std	0.0336
scores_min	0.6239
scores_25%	0.7362
scores_50%	0.7511
scores_75%	0.7511
scores_max	0.7680
Scoring function	function
Scoring model	RandomForestClassifier

Table 16: Preprocessing pipelines runtime statistics.

4 Modeling

4.1 Overview

This part of the report presents the results of the modeling process. There were 5 classification models trained for each of the best preprocessing pipelines.
The following models were used in the modeling process.

- KNeighborsClassifier

- LogisticRegression
- GaussianNB
- SVC
- DecisionTreeClassifier

4.2 Hyperparameter tuning

This section presents the results of hyperparameter tuning for each of the best 3 models using RandomizedSearchCV. Param grids used for each model are presented in the tables below.

Category	**Value**
n_neighbors	[5, 10, 15]
weights	['uniform', 'distance']
algorithm	['auto', 'ball_tree', 'kd_tree', 'brute']
leaf_size	[30, 40, 50]
p	[1, 2]

Table 17: Param grid for model KNeighboursClassifier.

Category	**Value**
0	{"penalty": ["l1"], "C": [0.01, 0.1, 1, 10], "solver": ["liblinear", "saga"]}
1	{"penalty": ["l2"], "C": [0.01, 0.1, 1, 10], "solver": ["lbfgs", "liblinear", "saga", "newton-cg"]}
2	{"penalty": ["elasticnet"], "C": [0.01, 0.1, 1, 10], "solver": ["saga"], "l1_ratio": [0.5, 0.7]}

Table 18: Param grid for model LogisticRegression.

Category	**Value**
priors	[None]
var_smoothing	[1e-09, 1e-07, 1e-05]

Table 19: Param grid for model GaussianNaiveClassifier.

Category	**Value**
C	[0.1, 1, 10, 100, 1000]
kernel	['linear', 'poly', 'rbf', 'sigmoid']
degree	[3, 4, 5]
gamma	['scale', 'auto']
random_state	[42]

Table 20: Param grid for model SVC.

Category	**Value**
criterion	['gini', 'entropy']
splitter	['best', 'random']
max_depth	[None, 5, 10, 15, 20]
min_samples_split	[2, 5, 10]
min_samples_leaf	[1, 2, 4]
random_state	[42]

Table 21: Param grid for model DecisionTreeClassifier.

Table 65 presents the best models and pipelines along with their hyperparameters, mean fit time, and test score.

Model	**Pipeline**	**Best params**	**Mean fit time**	**Test score**
KNeighborsClassifier	final_pipeline_2.joblib	{"weights": "uniform", "p": 2, "n_neighbors": 15, "leaf_size": 30, "algorithm": "kd_tree"}	a moment	0.7611
KNeighborsClassifier	final_pipeline_1.joblib	{"weights": "distance", "p": 2, "n_neighbors": 10, "leaf_size": 40, "algorithm": "auto"}	a moment	0.7356
KNeighborsClassifier	final_pipeline_0.joblib	{"weights": "distance", "p": 1, "n_neighbors": 15, "leaf_size": 30, "algorithm": "brute"}	a moment	0.7341

Table 22: Best models results

4.3 Interpretability

This section presents SHAP plots for the best model.

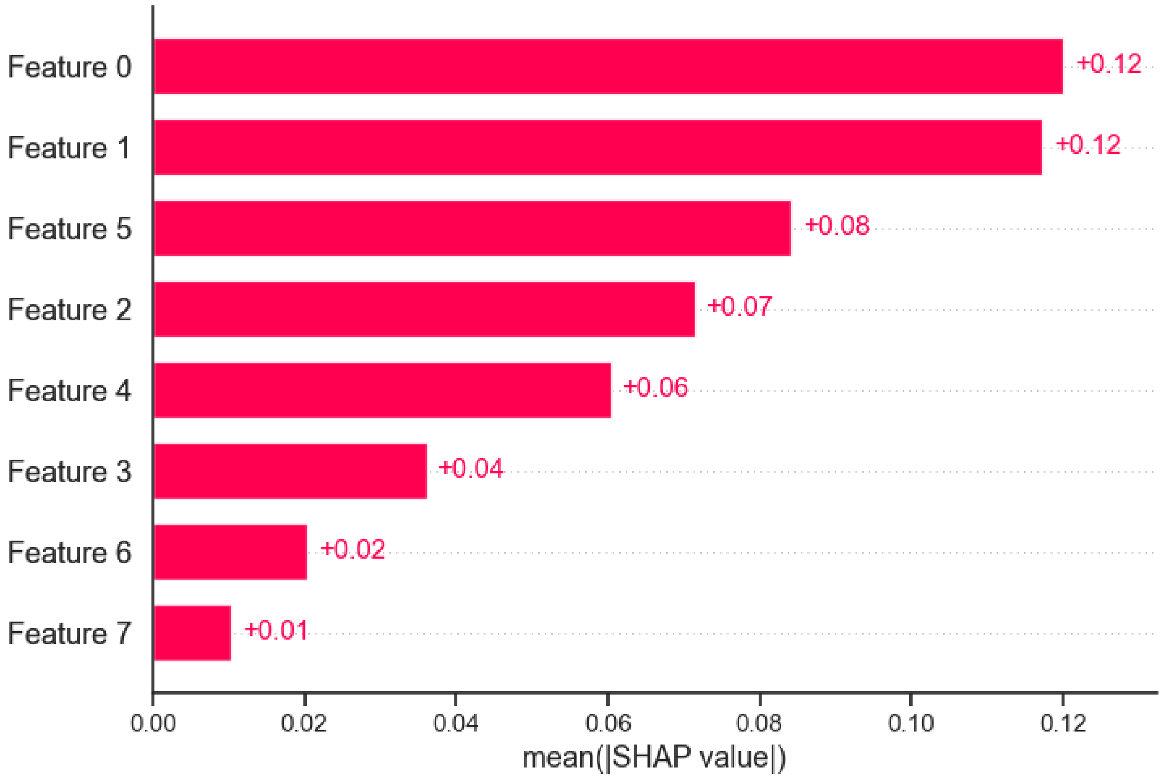

Figure 7: SHAP bar plot for class bar.

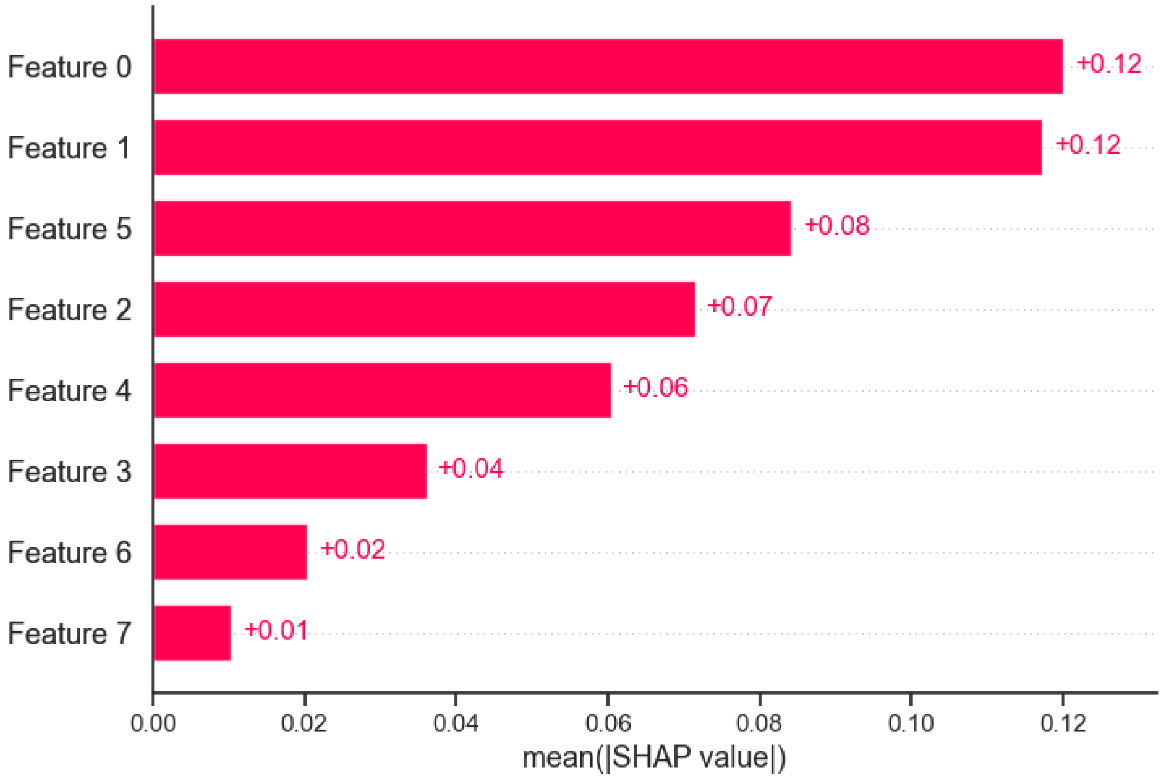

Figure 8: SHAP bar plot for class bar.

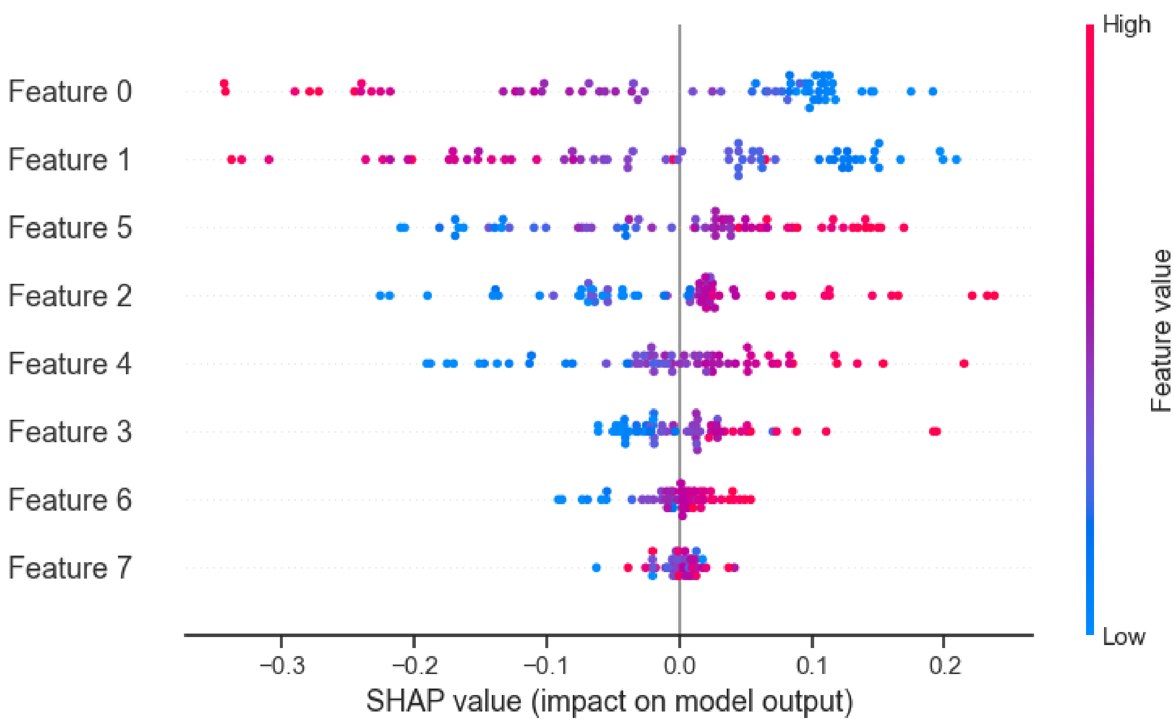

Figure 9: SHAP summary plot for class summary.

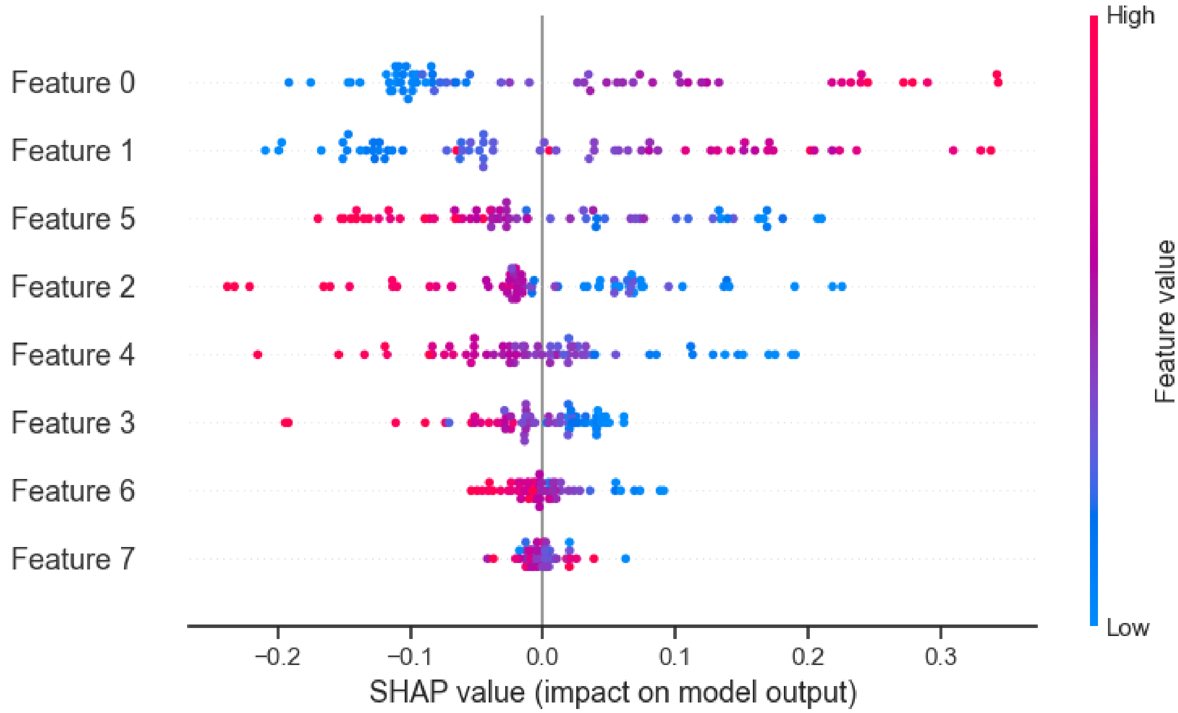

Figure 10: SHAP summary plot for class summary.

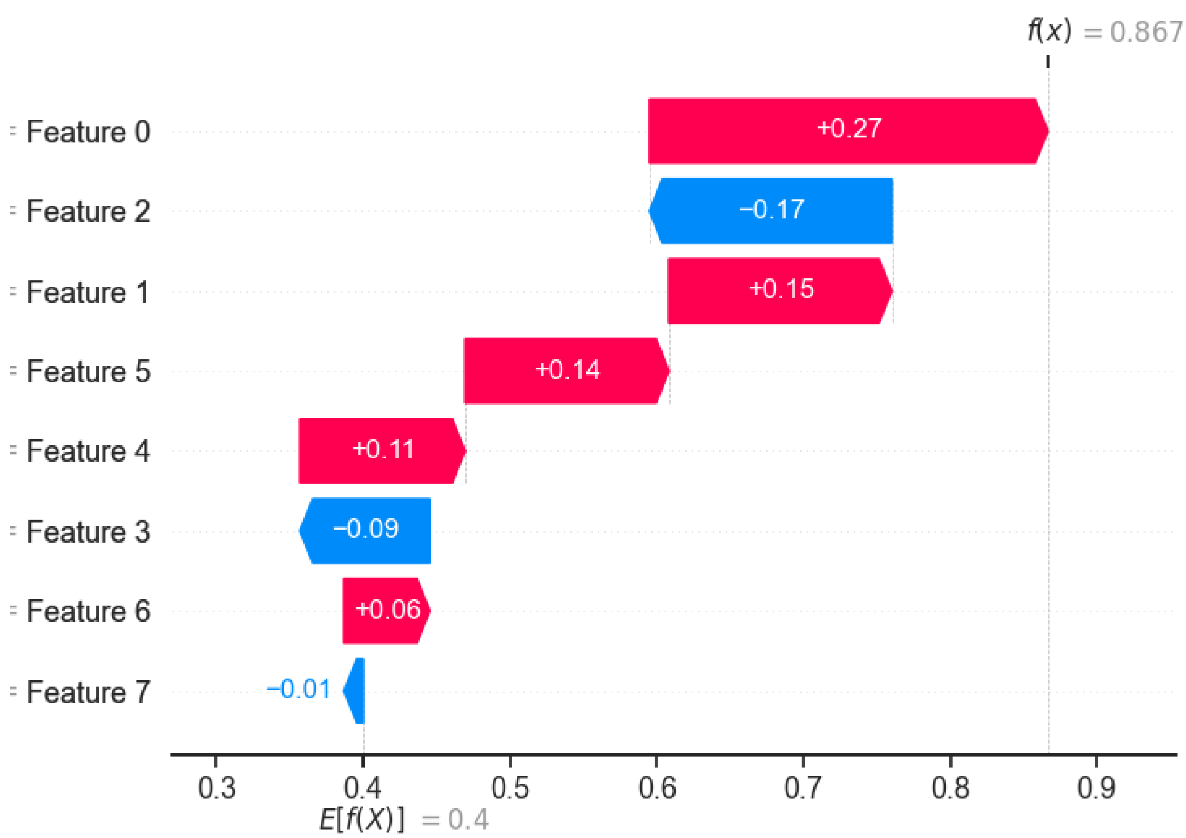

Figure 11: SHAP waterfall plot for class waterfall.

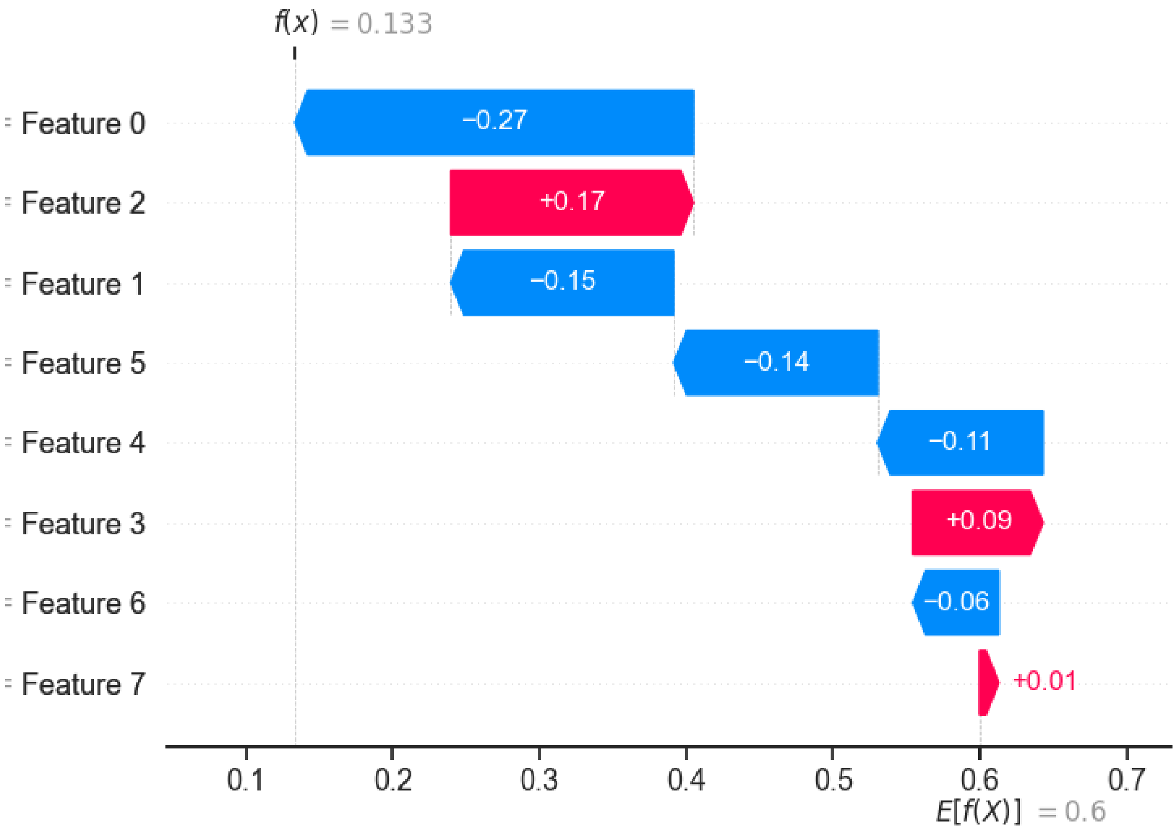

Figure 12: SHAP waterfall plot for class waterfall.

Abstract

This raport has been generated with AutoPrep.

Contents

5 Overview

5.1 System

System	Darwin
Machine	arm64
Processor	arm
Architecture	64bit
Python Version	3.10.5
Physical Cores	8
Logical Cores	8
CPU Frequency (MHz)	3204
Total RAM (GB)	16.0000
Available RAM (GB)	5.2200
Total Disk Space (GB)	228.2700
Free Disk Space (GB)	13.0700

Table 23: System overview.

5.2 Dataset

Task detected for the dataset: multiclass classfication.
Table 46 presents an overview of the dataset including the number of samples, features, and their types.

Number of samples	124
Number of features	8
Number of numerical features	2
Number of categorical features	6

Table 24: Dataset Summary.

Distribution of the target classes in terms of the number of observations and their percentages is presented in Table 25

class	**number of observations**	**fraction**
low	40	0.3226
high	34	0.2742
average	32	0.2581
veryhigh	18	0.1452

Table 25: Target class distribution.

Table 47 presents the distribution of missing values in the dataset.

feature	number of observations	fraction
year_zone	0	0.0000
year	0	0.0000
strip	0	0.0000
pdk	0	0.0000
damage_rankRJT	0	0.0000
damage_rankALL	0	0.0000
dry_or_irr	0	0.0000
zone	0	0.0000

Table 26: Missing values distribution.

Table 48 presents the description of features in the dataset.

feature	type	dtype	space usage
year_zone	categorical	category	2.9 kB
year	categorical	category	1.8 kB
strip	numerical	uint8	1.1 kB
pdk	numerical	uint8	1.1 kB
damage_rankRJT	categorical	category	1.6 kB
damage_rankALL	categorical	category	1.6 kB
dry_or_irr	categorical	category	1.4 kB
zone	categorical	category	1.4 kB

Table 27: Features dtypes description.

Table 49 and Table 29 present the description of numerical and categorical features in the dataset.

feature	count	mean	std	min	25%	50%	75%	max
strip	124.0000	5.2419	3.1632	1.0000	3.0000	5.0000	9.0000	10.0000
pdk	124.0000	2.2258	1.0580	0.0000	1.0000	2.0000	3.0000	5.0000

Table 28: Numerical features description.

index	count	unique	top	freq
year_zone	124	21	9f	11
year	124	7	91	22
damage_rankRJT	124	6	1	31
damage_rankALL	124	6	1	36
dry_or_irr	124	3	D	102
zone	124	3	F	61

Table 29: Categorical features description.

6 Eda

This part of the report provides basic insides to the data and the informations it holds..

6.1 Target variable and missing values

Figure 30 shows the distribution of the target variable.

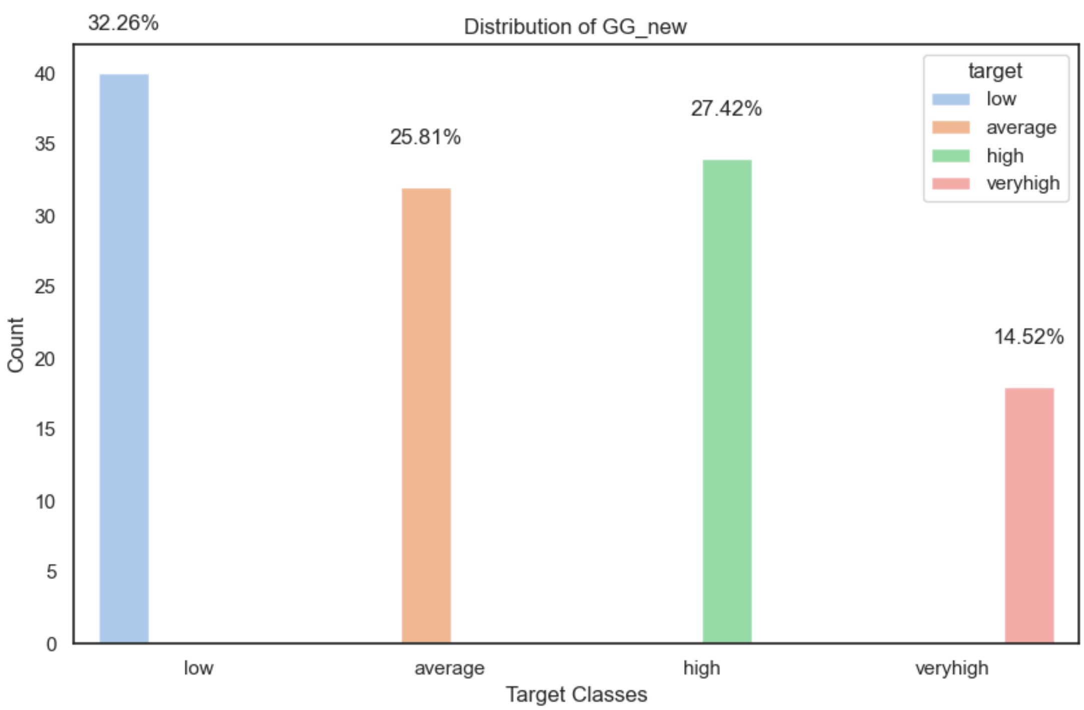

Figure 13: Target distribution.

6.2 EDA for categorical features

The distribution of categorical features is presented on barplot(s) below.

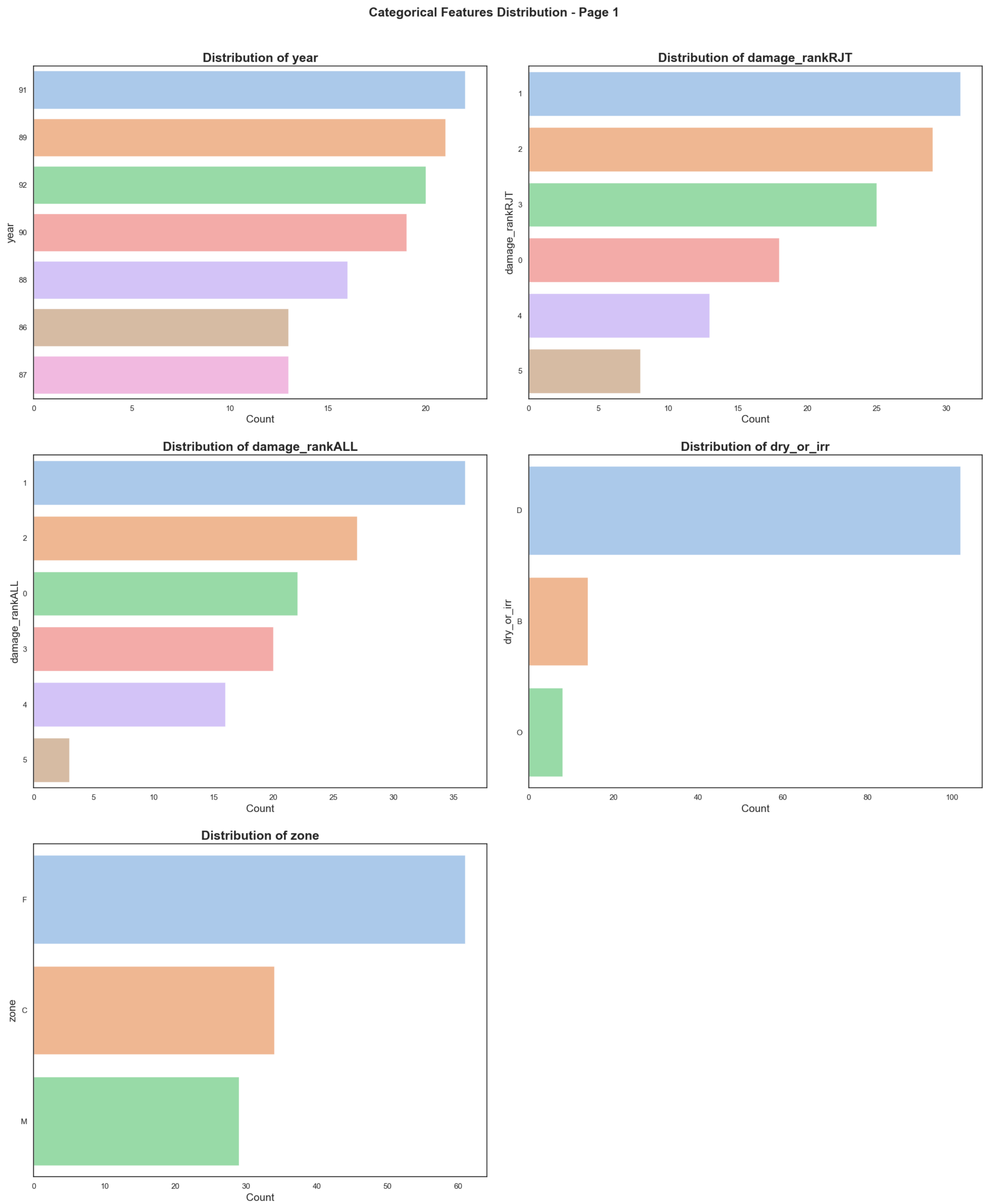

Figure 14: Categorical Features Distribution - Page 1

6.3 EDA for numerical features

The distribution of numerical features is presented on histogram(s) below.

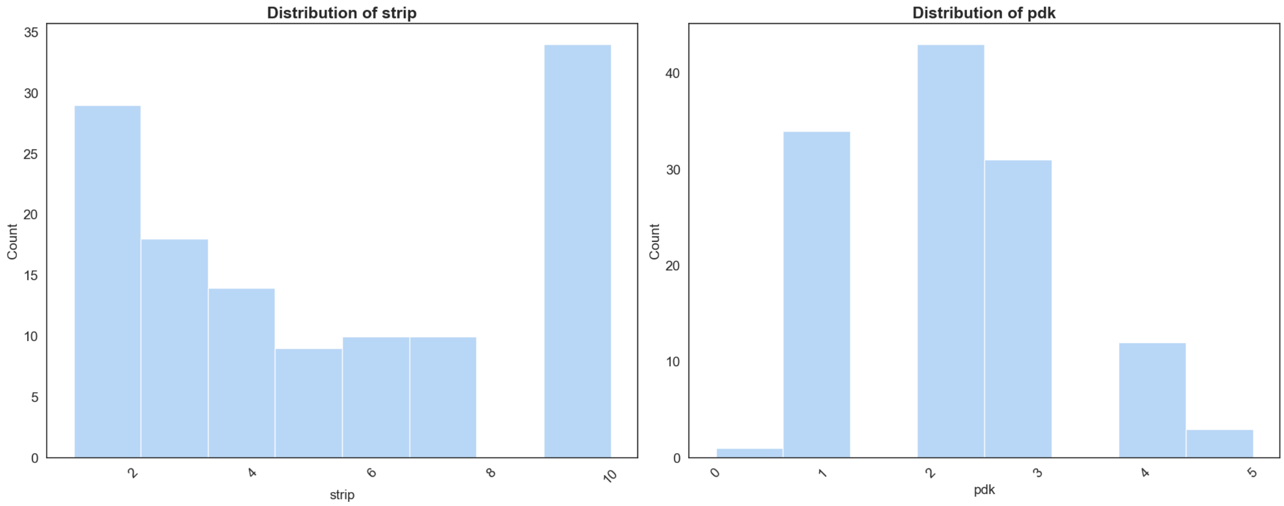

Figure 15: Numerical Features Distribution - Page 1

Figure 33 shows the correlation between features.

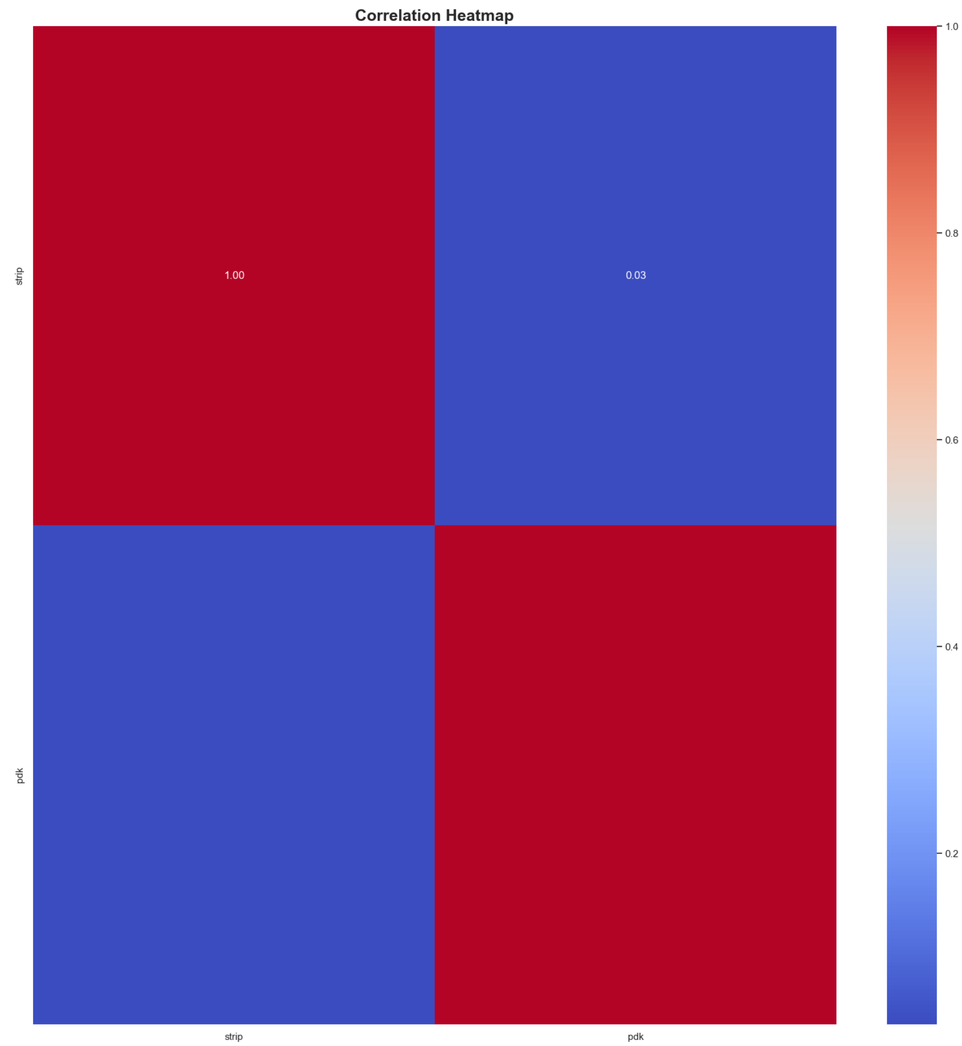

Figure 16: Correlation heatmap.

The boxplot of numerical features is presented on chart(s) below.

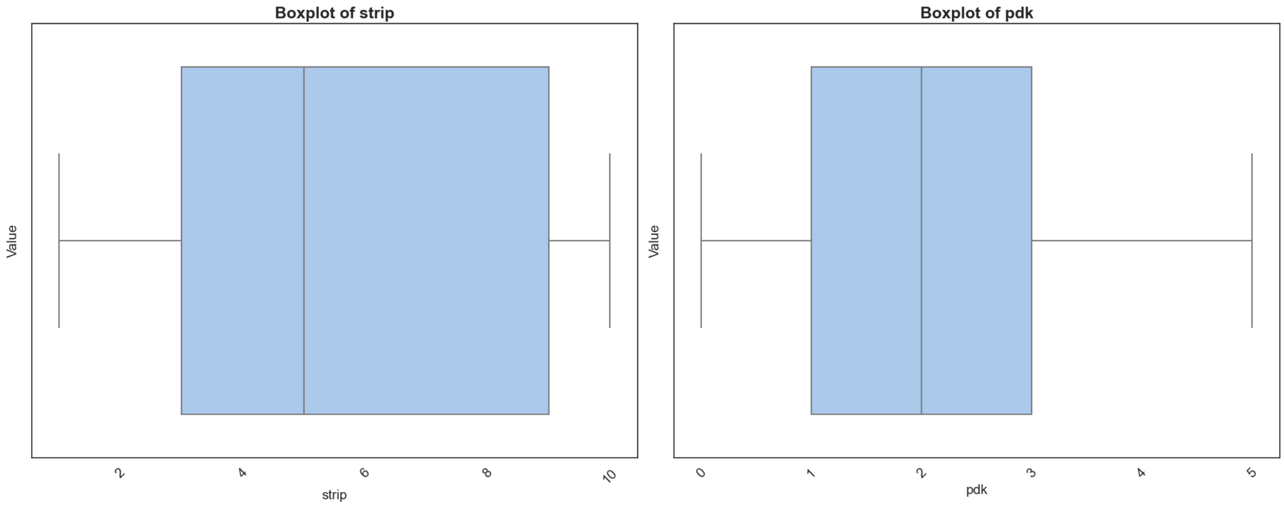

Figure 17: Boxplot page 1

7 Preprocessing

This part of the report presents the results of the preprocessing process. It contains required, as well as non required, steps listed below.

Required preprocessing steps:

- Missing data imputation
- Removing columns with 100% unique categorical values
- Categorical features encoding
- Scaling
- Removing columns with 0 variance
- Detecting highly correlatd features

Additional preprocessing steps:

- Feature selection methods : Correlation with the target or Random Forest feature importance
- Dimention reduction techniques: PCA, VIF, UMAP

Preprocessing process was configured to select up to 3 best unique preprocessing pipelines. Pipelines were scored based on a simple model. Tables below show detailed description of the best pipelines as well as all step combinations that were examined.

index	steps
0	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler
1	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, CorrelationSelector
2	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, FeatureImportanceRegressSelector
3	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, FeatureImportanceClassSelector
4	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, PCADimentionReducer
5	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, CorrelationSelector, PCADimentionReducer
6	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, FeatureImportanceRegressSelector, PCADimentionReducer
7	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, FeatureImportanceClassSelector, PCADimentionReducer
8	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, UMAPDimentionReducer
9	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, CorrelationSelector, UMAPDimentionReducer
10	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, FeatureImportanceRegressSelector, UMAPDimentionReducer
11	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, FeatureImportanceClassSelector, UMAPDimentionReducer
12	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, VIFDimentionReducer
13	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, CorrelationSelector, VIFDimentionReducer
14	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, FeatureImportanceRegressSelector, VIFDimentionReducer
15	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, FeatureImportanceClassSelector, VIFDimentionReducer

Table 30: Pipelines steps overview.

index	file name	score	fit duration	score duration
0	preprocessing_pipeline_0.joblib	0.7333	a moment	a moment
1	preprocessing_pipeline_1.joblib	0.7333	7 seconds	7 seconds
2	preprocessing_pipeline_2.joblib	0.6667	a moment	a moment

Table 31: Best preprocessing pipelines.

step	name	description	params
0	NAImputer	Imputes missing data.	{"numeric_imputer": "median", "categorical_imputer": "most_frequent"}
1	UniqueFilter	Removes categorical columns with 100% unique values. Dropped columns: []	{}
2	ColumnEncoder	Encodes categorical columns using OneHotEncoder (for columns with <5 unique values) or TolerantLabelEncoder (for columns with >=5 unique values). Encodes target variable using LabelEncoder if provided.	{}
3	VarianceFilter	Removes columns with zero variance. Dropped columns: []	{}
4	CorrelationFilter	Removes one column from pairs of columns correlated above correlation threshold: 0.8.	{}
5	ColumnScaler	Scales numerical columns using one of 3 scaling methods.	{"method": "standard"}
6	PCADimentionReducer	Combines PCA with automatic selection of the number of components to preserve 95% of the variance.	{"n_components": null}

Table 32: Best pipeline No. 0: steps overview.

index	count	mean	std	min	25%	50%	75%	max
year_zone	124.0000	0.0000	1.0041	-1.5281	-0.8943	-0.1022	0.8880	1.6405
year	124.0000	-0.0000	1.0041	-1.7377	-0.6967	-0.1763	0.8646	1.3851
strip	124.0000	-0.0000	1.0041	-1.3465	-0.7116	-0.0768	1.1929	1.5103
pdk	124.0000	0.0000	1.0041	-2.1124	-1.1633	-0.2143	0.7347	2.6328
damage_rankRJT	124.0000	-0.0000	1.0041	-1.4497	-0.7475	-0.0453	0.6569	2.0613
damage_rankALL	124.0000	0.0000	1.0041	-1.3499	-0.6189	0.1120	0.8429	2.3048
dry_or_irr_B	124.0000	0.0000	1.0041	-0.3568	-0.3568	-0.3568	-0.3568	2.8031
dry_or_irr_D	124.0000	-0.0000	1.0041	-2.1532	0.4644	0.4644	0.4644	0.4644
dry_or_irr_O	124.0000	-0.0000	1.0041	-0.2626	-0.2626	-0.2626	-0.2626	3.8079
zone_M	124.0000	-0.0000	1.0041	-0.5525	-0.5525	-0.5525	-0.5525	1.8099

Table 33: Best pipeline No. 0: output overview.

step	name	description	params
0	NAImputer	Imputes missing data.	{"numeric_imputer": "median", "categorical_imputer": "most_frequent"}
1	UniqueFilter	Removes categorical columns with 100% unique values. Dropped columns: []	{}
2	ColumnEncoder	Encodes categorical columns using OneHotEncoder (for columns with <5 unique values) or TolerantLabelEncoder (for columns with >=5 unique values). Encodes target variable using LabelEncoder if provided.	{}
3	VarianceFilter	Removes columns with zero variance. Dropped columns: []	{}
4	CorrelationFilter	Removes one column from pairs of columns correlated above correlation threshold: 0.8.	{}
5	ColumnScaler	Scales numerical columns using one of 3 scaling methods.	{"method": "robust"}
6	CorrelationSelector	Selects the top 70.0% (rounded to whole number) of features most correlated with the target variable. Number of features that were selected: 0	{"correlation_percent": 0.7}
7	UMAPDimentionReducer	Reduces the dimensionality of the data using UMAP.	{"n_components": null}

Table 34: Best pipeline No. 1: steps overview.

index	count	mean	std	min	25%	50%	75%	max
year_zone	124.0000	0.4823	0.3169	0.0000	0.2000	0.4500	0.7625	1.0000
year	124.0000	0.5565	0.3215	0.0000	0.3333	0.5000	0.8333	1.0000
strip	124.0000	0.4713	0.3515	0.0000	0.2222	0.4444	0.8889	1.0000
pdk	124.0000	0.4452	0.2116	0.0000	0.2000	0.4000	0.6000	1.0000
damage_rankRJT	124.0000	0.4129	0.2860	0.0000	0.2000	0.4000	0.6000	1.0000
damage_rankALL	124.0000	0.3694	0.2747	0.0000	0.2000	0.4000	0.6000	1.0000
dry_or_irr_B	124.0000	0.1129	0.3178	0.0000	0.0000	0.0000	0.0000	1.0000
dry_or_irr_D	124.0000	0.8226	0.3836	0.0000	1.0000	1.0000	1.0000	1.0000
dry_or_irr_O	124.0000	0.0645	0.2467	0.0000	0.0000	0.0000	0.0000	1.0000
zone_M	124.0000	0.2339	0.4250	0.0000	0.0000	0.0000	0.0000	1.0000

Table 35: Best pipeline No. 1: output overview.

step	name	description	params
0	NAImputer	Imputes missing data.	{"numeric_imputer": "median", "categorical_imputer": "most_frequent"}
1	UniqueFilter	Removes categorical columns with 100% unique values. Dropped columns: []	{}
2	ColumnEncoder	Encodes categorical columns using OneHotEncoder (for columns with <5 unique values) or TolerantLabelEncoder (for columns with >=5 unique values). Encodes target variable using LabelEncoder if provided.	{}
3	VarianceFilter	Removes columns with zero variance. Dropped columns: []	{}
4	CorrelationFilter	Removes one column from pairs of columns correlated above correlation threshold: 0.8.	{}
5	ColumnScaler	Scales numerical columns using one of 3 scaling methods.	{"method": "standard"}

Table 36: Best pipeline No. 2: steps overview.

index	count	mean	std	min	25%	50%	75%	max
year_zone	124.0000	0.0573	0.5633	-0.8000	-0.4444	0.0000	0.5556	0.9778
year	124.0000	0.1129	0.6431	-1.0000	-0.3333	0.0000	0.6667	1.0000
strip	124.0000	0.0403	0.5272	-0.6667	-0.3333	0.0000	0.6667	0.8333
pdk	124.0000	0.1129	0.5290	-1.0000	-0.5000	0.0000	0.5000	1.5000
damage_rankRJT	124.0000	0.0323	0.7149	-1.0000	-0.5000	0.0000	0.5000	1.5000
damage_rankALL	124.0000	-0.0766	0.6868	-1.0000	-0.5000	0.0000	0.5000	1.5000
dry_or_irr_B	124.0000	0.1129	0.3178	0.0000	0.0000	0.0000	0.0000	1.0000
dry_or_irr_D	124.0000	-0.1774	0.3836	-1.0000	0.0000	0.0000	0.0000	0.0000
dry_or_irr_O	124.0000	0.0645	0.2467	0.0000	0.0000	0.0000	0.0000	1.0000
zone_M	124.0000	0.2339	0.4250	0.0000	0.0000	0.0000	0.0000	1.0000

Table 37: Best pipeline No. 2: output overview.

Category	Value
Unique created pipelines	16
All created pipelines (after exploading each step params)	48
All pipelines fit time	18 seconds
All pipelines score time	17 seconds
scores_count	48.0000
scores_mean	0.5653
scores_std	0.0848
scores_min	0.3333
scores_25%	0.5333
scores_50%	0.5333
scores_75%	0.6000
scores_max	0.7333
Scoring function	function
Scoring model	RandomForestClassifier

Table 38: Preprocessing pipelines runtime statistics.

8 Modeling

8.1 Overview

This part of the report presents the results of the modeling process. There were 5 classification models trained for each of the best preprocessing pipelines.
The following models were used in the modeling process.

- KNeighborsClassifier
- LogisticRegression
- GaussianNB
- SVC
- DecisionTreeClassifier

8.2 Hyperparameter tuning

This section presents the results of hyperparameter tuning for each of the best 3 models using RandomizedSearchCV. Param grids used for each model are presented in the tables below.

Category	Value
n_neighbors	[5, 10, 15]
weights	['uniform', 'distance']
algorithm	['auto', 'ball_tree', 'kd_tree', 'brute']
leaf_size	[30, 40, 50]
p	[1, 2]

Table 39: Param grid for model KNeighboursClassifier.

Category	Value
0	{"penalty": ["l1"], "C": [0.01, 0.1, 1, 10], "solver": ["liblinear", "saga"]}
1	{"penalty": ["l2"], "C": [0.01, 0.1, 1, 10], "solver": ["lbfgs", "liblinear", "saga", "newton-cg"]}
2	{"penalty": ["elasticnet"], "C": [0.01, 0.1, 1, 10], "solver": ["saga"], "l1_ratio": [0.5, 0.7]}

Table 40: Param grid for model LogisticRegression.

Category	Value
priors	[None]
var_smoothing	[1e-09, 1e-07, 1e-05]

Table 41: Param grid for model GaussianNaiveClassifier.

Category	Value
C	[0.1, 1, 10, 100, 1000]
kernel	['linear', 'poly', 'rbf', 'sigmoid']
degree	[3, 4, 5]
gamma	['scale', 'auto']
random_state	[42]

Table 42: Param grid for model SVC.

Category	Value
criterion	['gini', 'entropy']
splitter	['best', 'random']
max_depth	[None, 5, 10, 15, 20]
min_samples_split	[2, 5, 10]
min_samples_leaf	[1, 2, 4]
random_state	[42]

Table 43: Param grid for model DecisionTreeClassifier.

Table 65 presents the best models and pipelines along with their hyperparameters, mean fit time, and test score.

Model	Pipeline	Best params	Mean fit time	Test score
KNeighborsClassifier	final_pipeline_0.joblib	{"weights": "distance", "p": 1, "n_neighbors": 15, "leaf_size": 30, "algorithm": "brute"}	a moment	0.0000
KNeighborsClassifier	final_pipeline_1.joblib	{"weights": "distance", "p": 2, "n_neighbors": 10, "leaf_size": 40, "algorithm": "auto"}	a moment	0.0000
KNeighborsClassifier	final_pipeline_2.joblib	{"weights": "uniform", "p": 2, "n_neighbors": 15, "leaf_size": 30, "algorithm": "kd_tree"}	a moment	0.0000

Table 44: Best models results

8.3 Interpretability

This section presents SHAP plots for the best model.

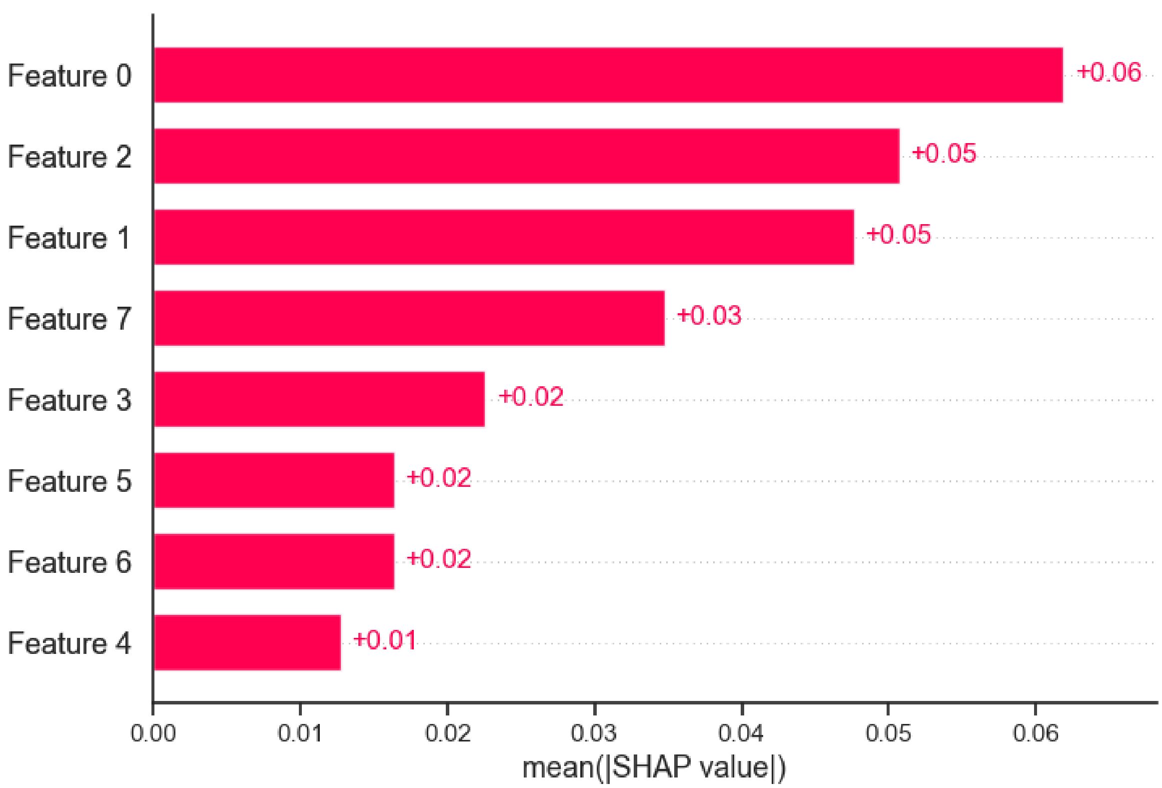

Figure 18: SHAP bar plot for class bar.

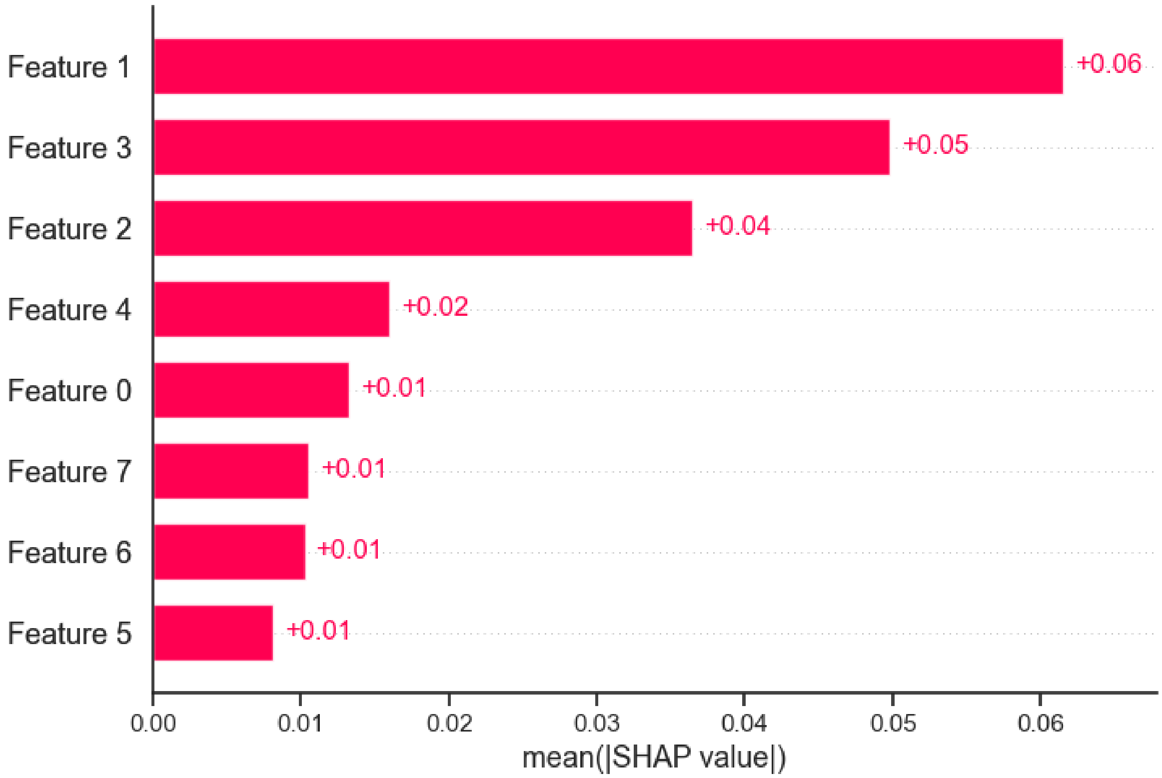

Figure 19: SHAP bar plot for class bar.

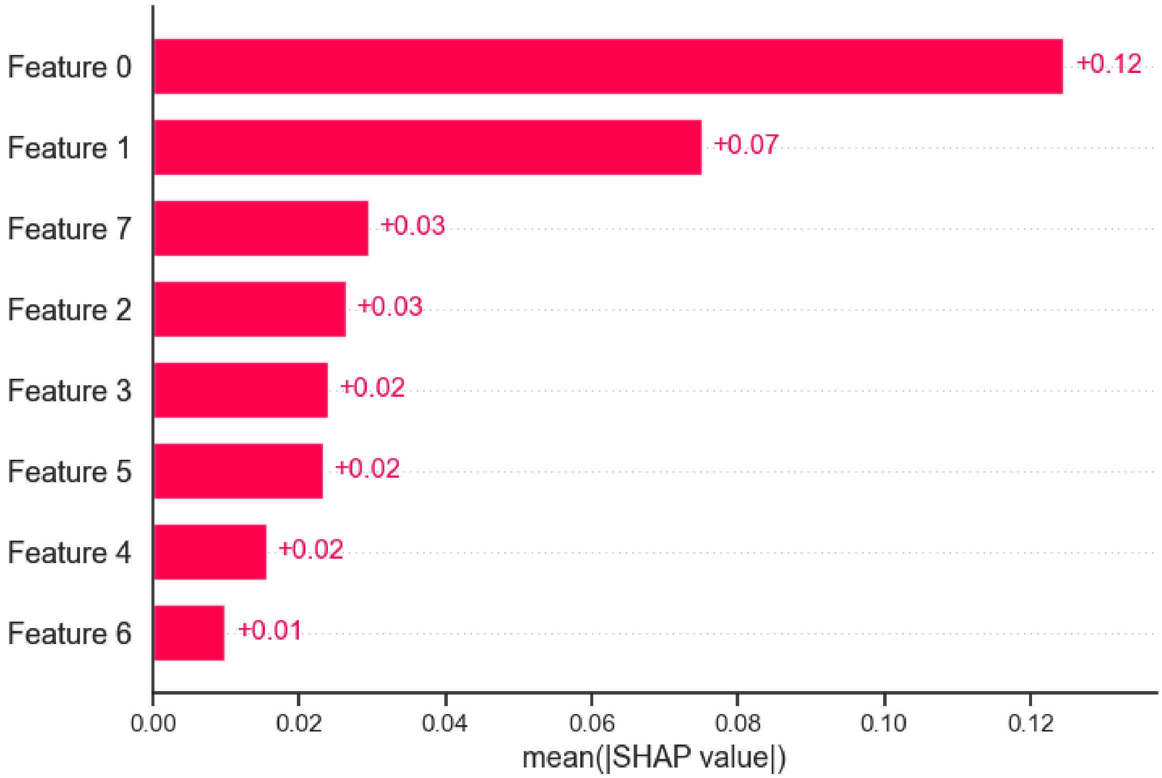

Figure 20: SHAP bar plot for class bar.

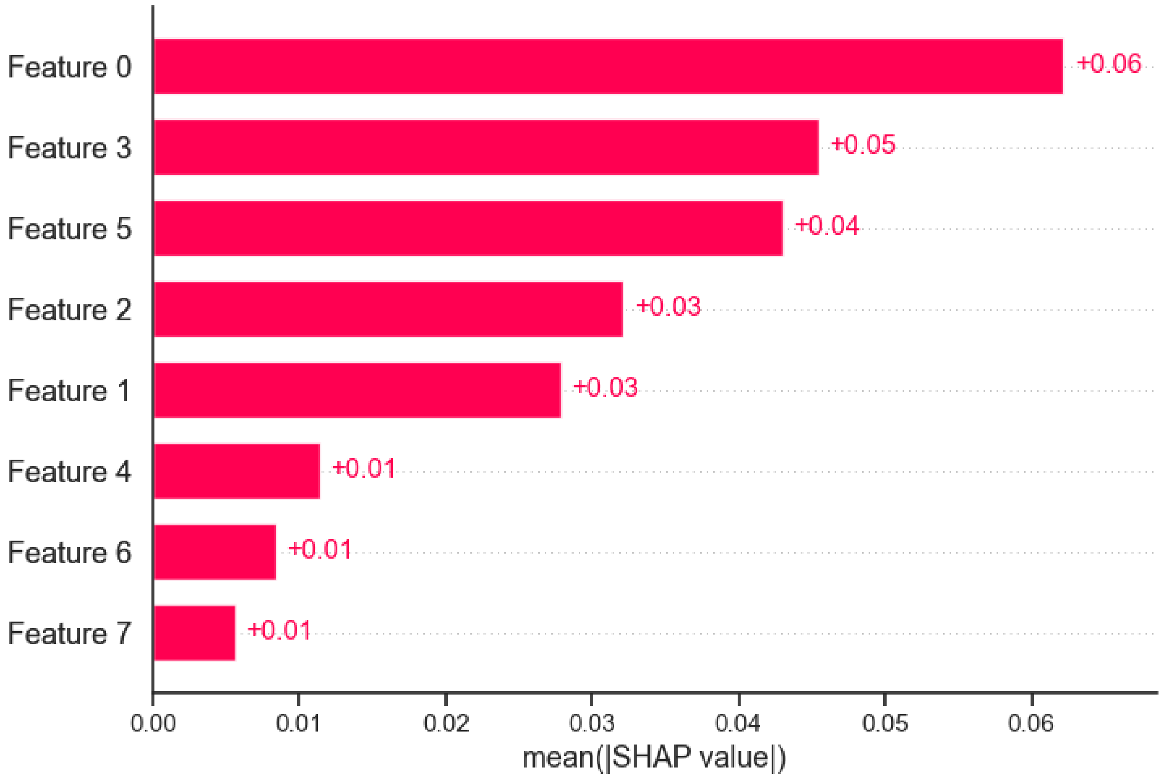

Figure 21: SHAP bar plot for class bar.

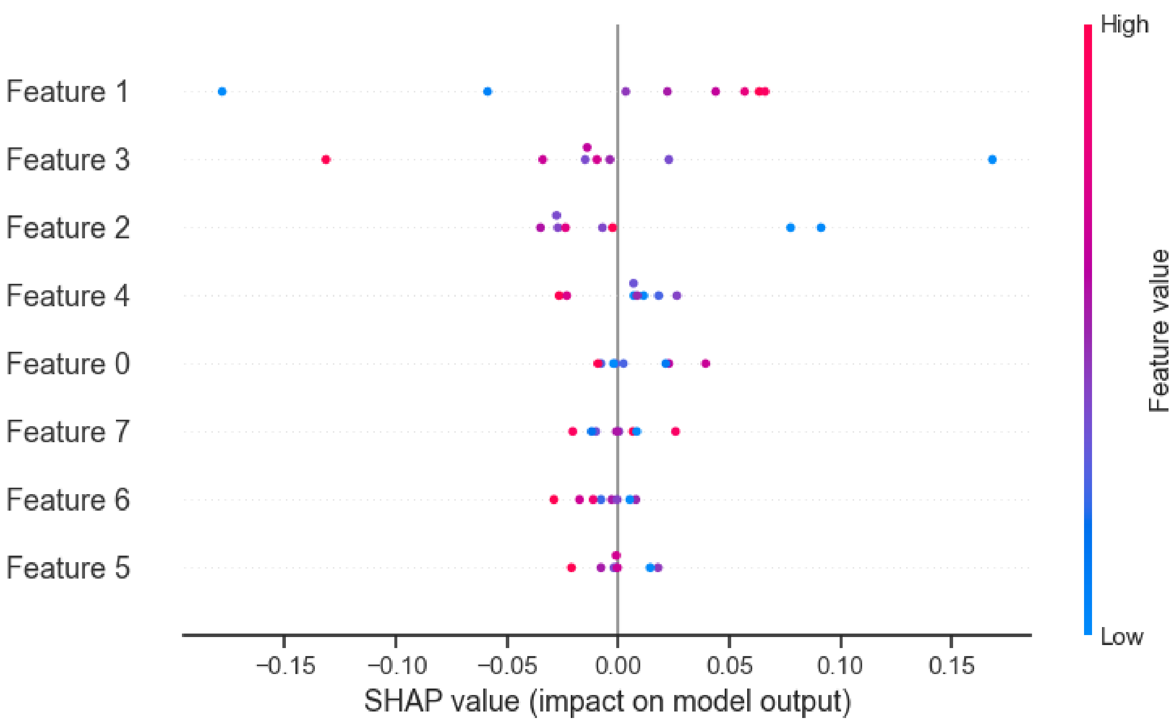

Figure 22: SHAP summary plot for class summary.

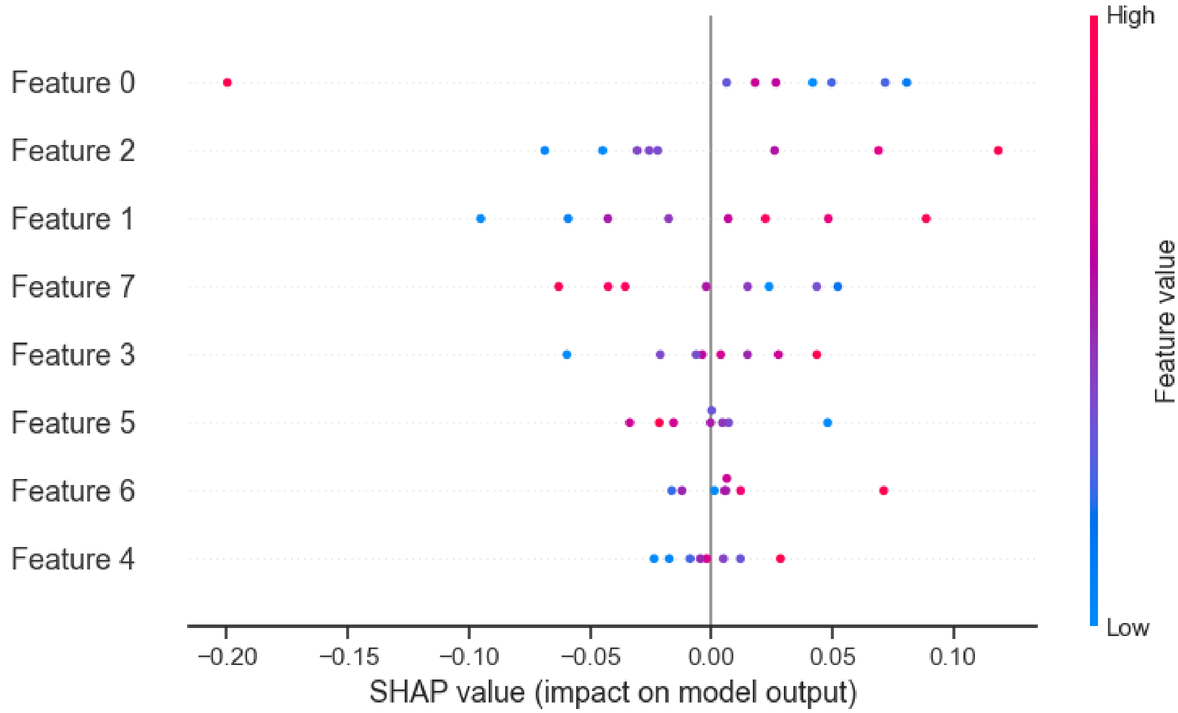

Figure 23: SHAP summary plot for class summary.

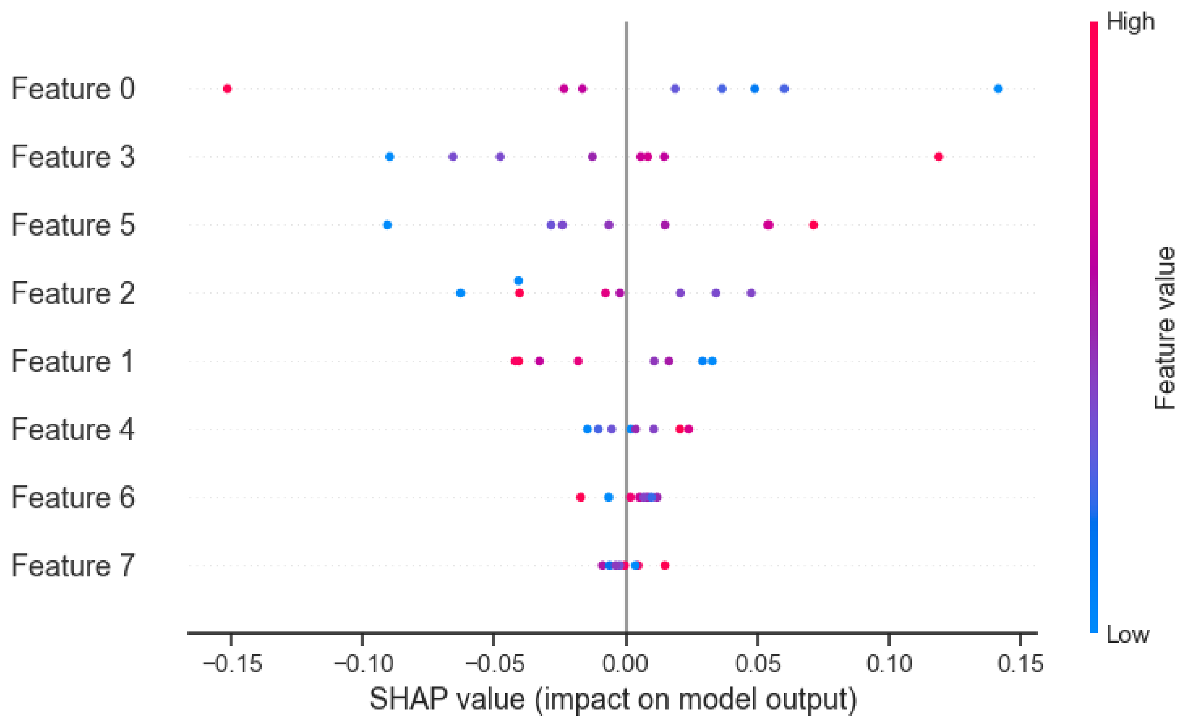

Figure 24: SHAP summary plot for class summary.

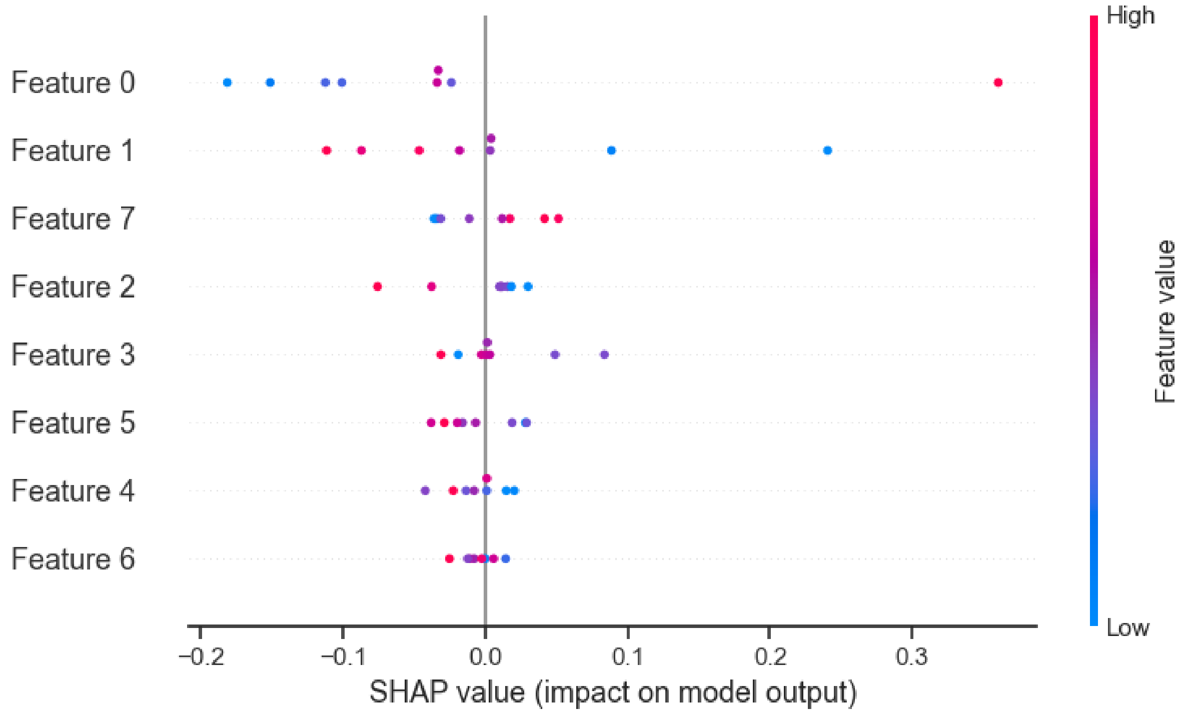

Figure 25: SHAP summary plot for class summary.

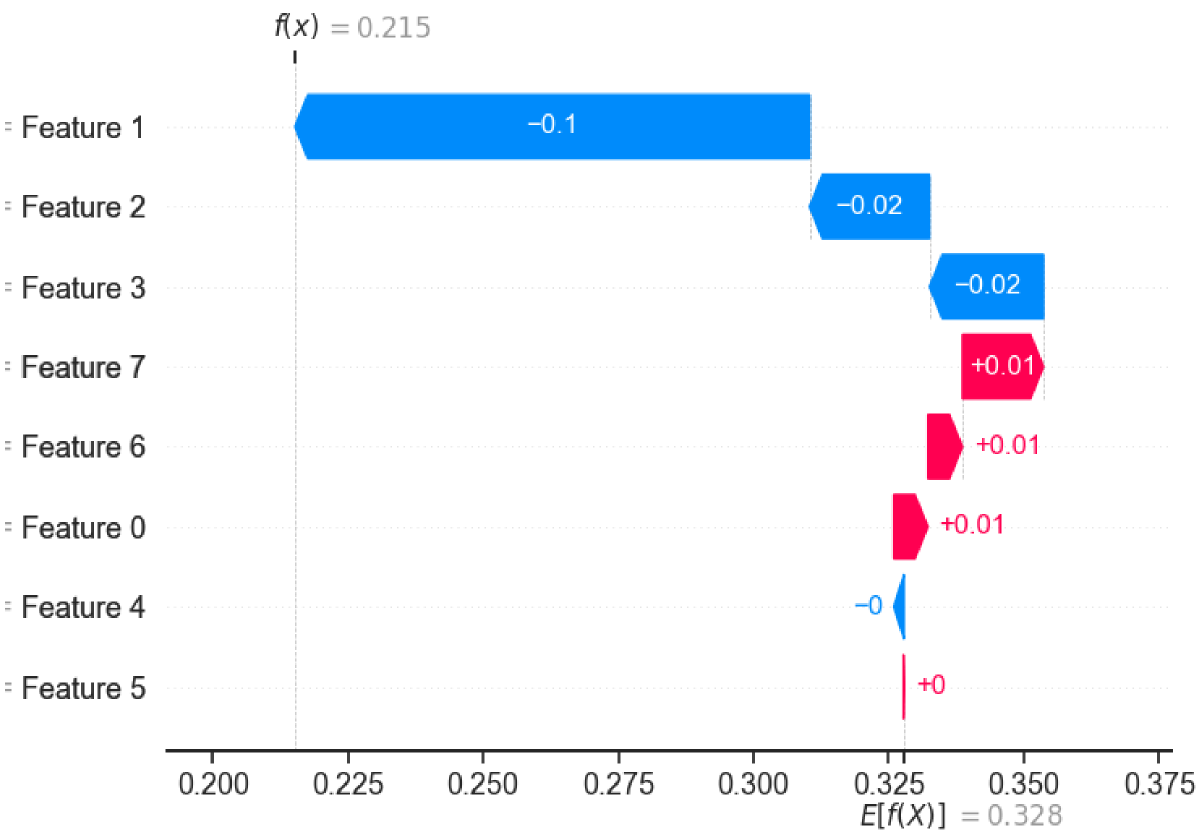

Figure 26: SHAP waterfall plot for class waterfall.

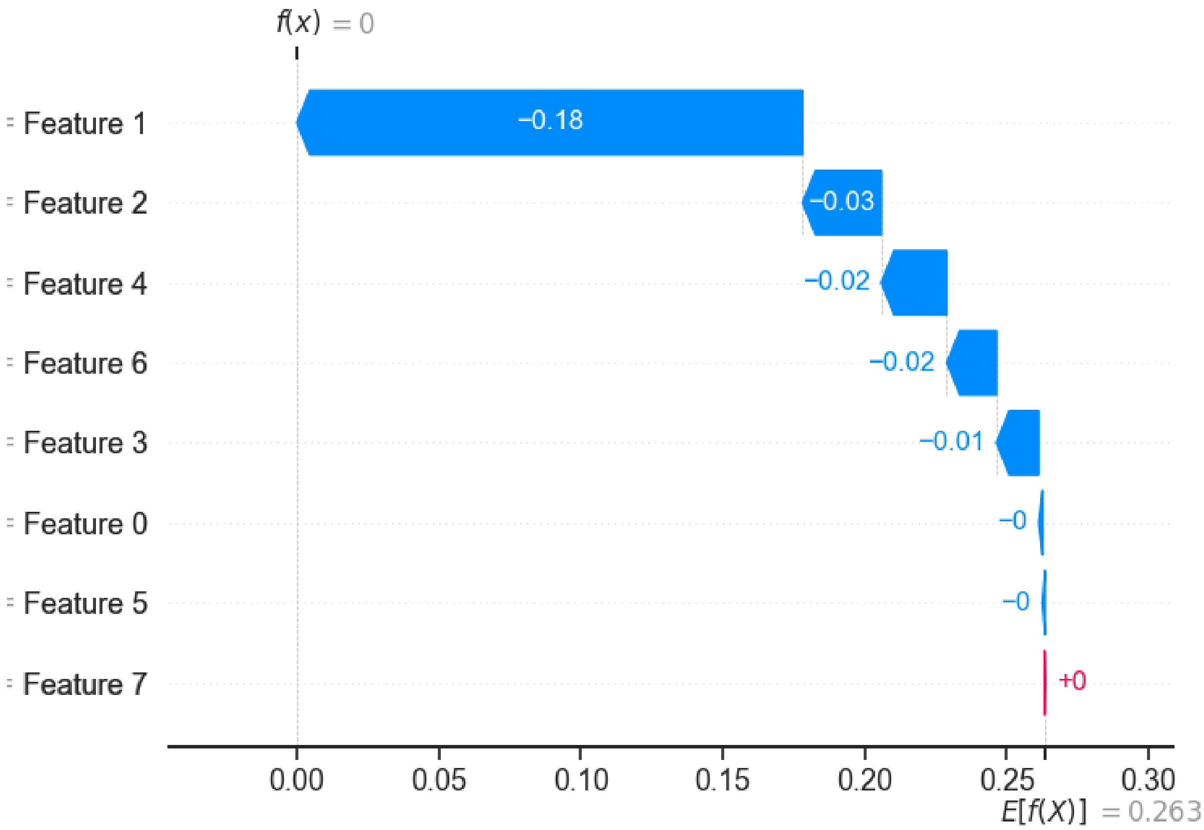

Figure 27: SHAP waterfall plot for class waterfall.

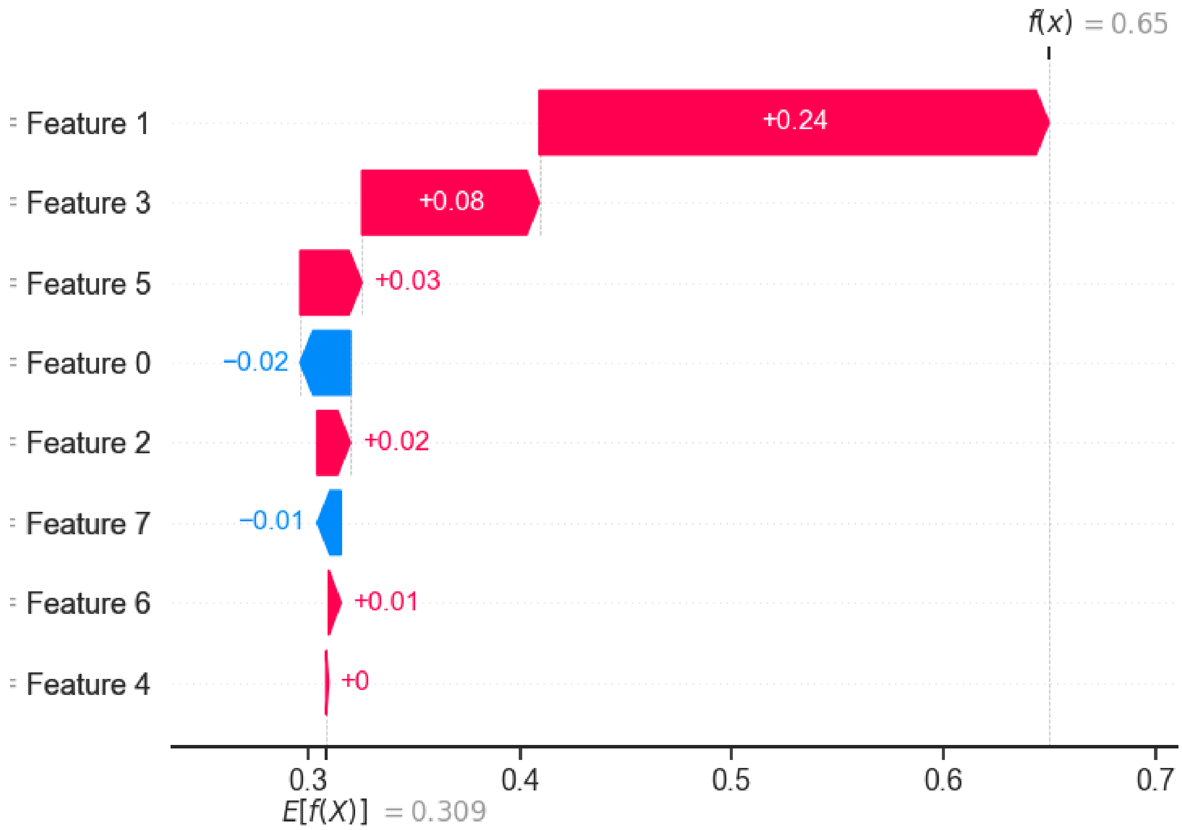

Figure 28: SHAP waterfall plot for class waterfall.

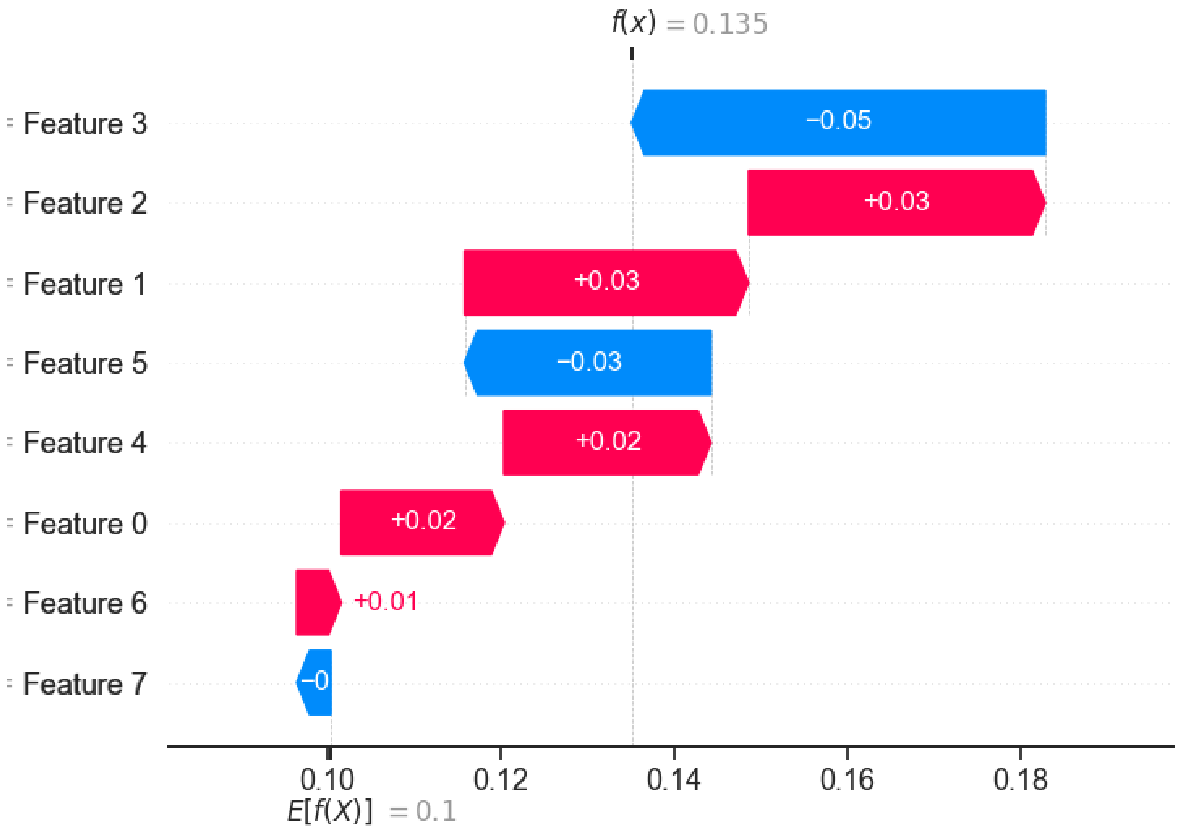

Figure 29: SHAP waterfall plot for class waterfall.

Abstract

This raport has been generated with AutoPrep.

Contents

9 Overview

9.1 System

System	Darwin
Machine	arm64
Processor	arm
Architecture	64bit
Python Version	3.10.5
Physical Cores	8
Logical Cores	8
CPU Frequency (MHz)	3204
Total RAM (GB)	16.0000
Available RAM (GB)	5.5300
Total Disk Space (GB)	228.2700
Free Disk Space (GB)	13.0700

Table 45: System overview.

9.2 Dataset

Task detected for the dataset: regression.
Table 46 presents an overview of the dataset including the number of samples, features, and their types.

Number of samples	227
Number of features	9
Number of numerical features	9
Number of categorical features	0

Table 46: Dataset Summary.

Table 47 presents the distribution of missing values in the dataset.

feature	number of observations	fraction
P85	0	0.0000
P75	0	0.0000
RMT85	0	0.0000
CS82	0	0.0000
SS82	0	0.0000
S82	0	0.0000
ME84	0	0.0000
REV84	0	0.0000
REG	0	0.0000

Table 47: Missing values distribution.

Table 48 presents the description of features in the dataset.

feature	type	dtype	space usage
P85	numerical	int64	3.6 kB
P75	numerical	int64	3.6 kB
RMT85	numerical	int64	3.6 kB
CS82	numerical	uint8	2.0 kB
SS82	numerical	uint8	2.0 kB
S82	numerical	uint8	2.0 kB
ME84	numerical	int64	3.6 kB
REV84	numerical	int64	3.6 kB
REG	numerical	uint8	2.0 kB

Table 48: Features dtypes description.

Table 49 presents the description of numerical features in the dataset.

index	count	mean	std	min	25%	50%	75%	max
P85	227.0000	29.9912	56.1690	3.0000	10.0000	16.0000	30.0000	653.0000
P75	227.0000	29.5242	57.7682	4.0000	10.0000	15.0000	28.0000	671.0000
RMT85	227.0000	254.5066	657.6030	21.0000	66.5000	118.0000	229.5000	6720.0000
CS82	227.0000	9.1762	4.9836	1.0000	6.0000	8.0000	11.0000	34.0000
SS82	227.0000	21.9515	7.2284	8.0000	17.0000	21.0000	27.0000	46.0000
S82	227.0000	47.1498	10.5694	31.0000	41.0000	45.0000	49.0000	101.0000
ME84	227.0000	1842.4141	4685.0646	173.0000	480.5000	839.0000	1580.5000	47074.0000
REV84	227.0000	3048.3084	5125.1721	347.0000	1134.5000	1828.0000	3174.0000	59877.0000
REG	227.0000	4.3304	2.0805	1.0000	2.0000	4.0000	6.0000	8.0000

Table 49: Numerical features description.

10 Eda

This part of the report provides basic insides to the data and the informations it holds..

10.1 Target variable and missing values

Figure 30 shows the distribution of the target variable.

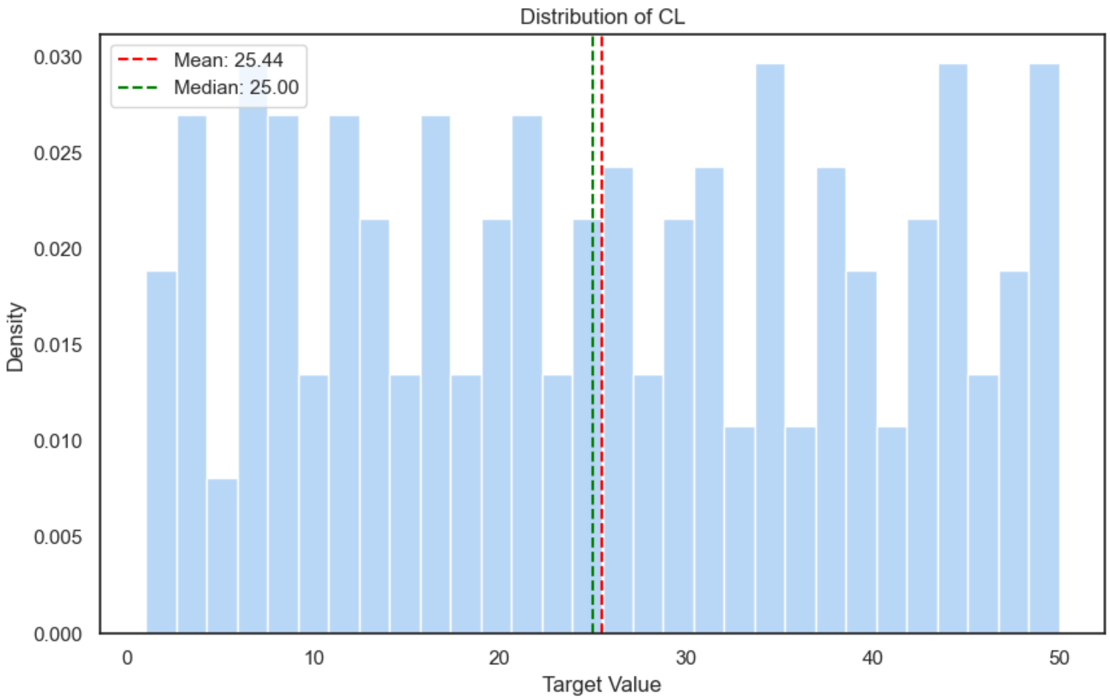

Figure 30: Target distribution.

10.2 EDA for numerical features

The distribution of numerical features is presented on histogram(s) below.

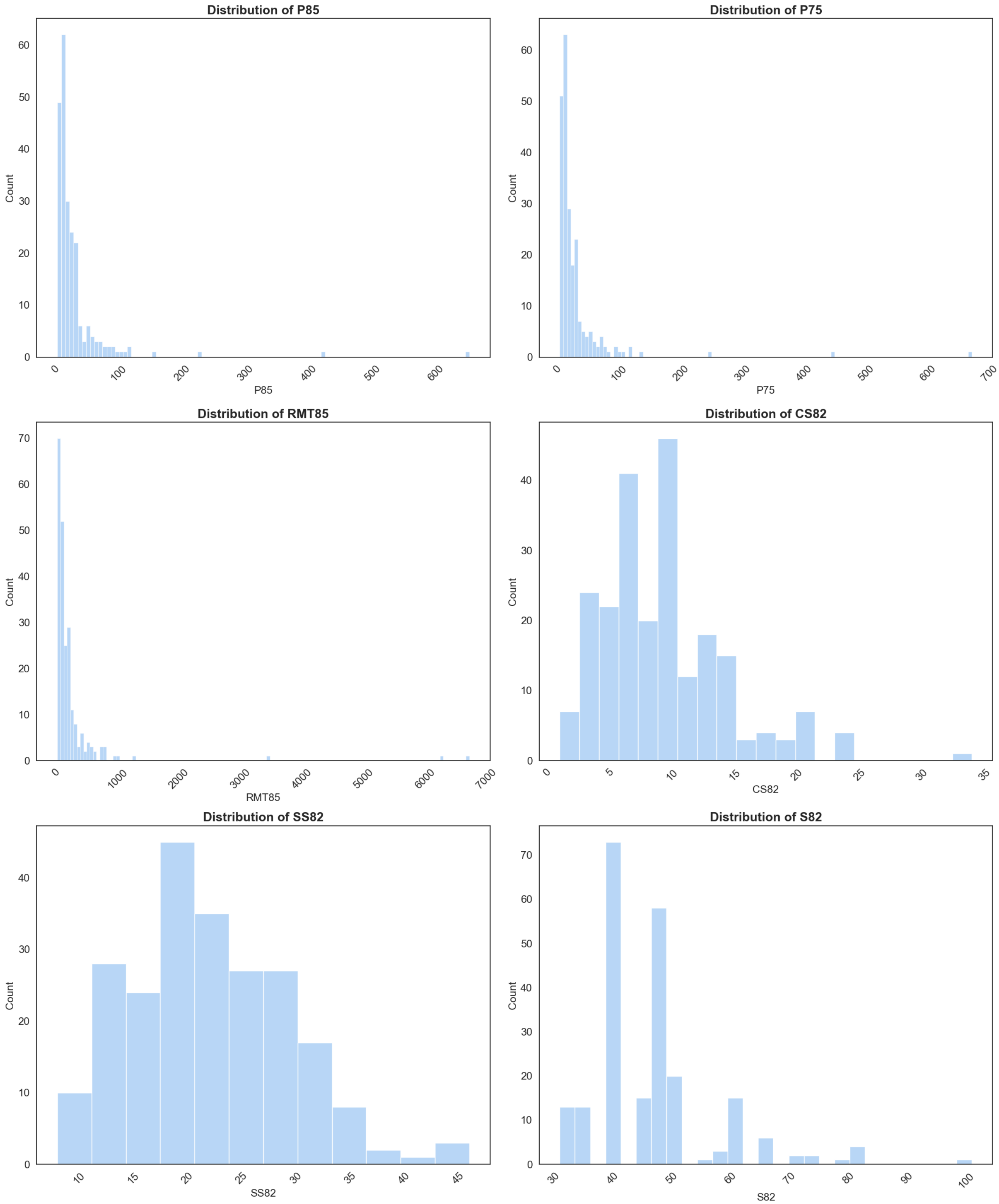

Figure 31: Numerical Features Distribution - Page 1

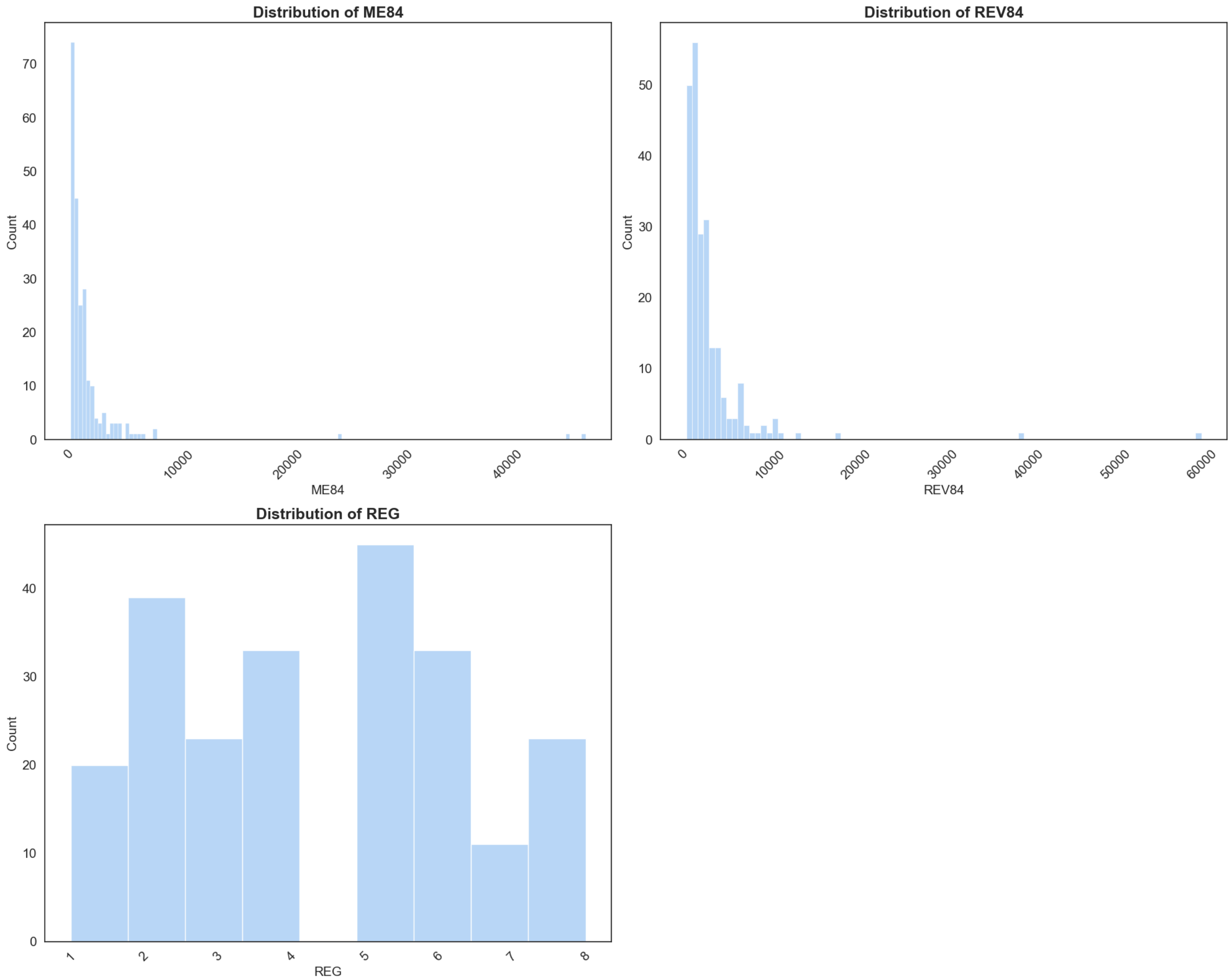

Figure 32: Numerical Features Distribution - Page 2

Figure 33 shows the correlation between features.

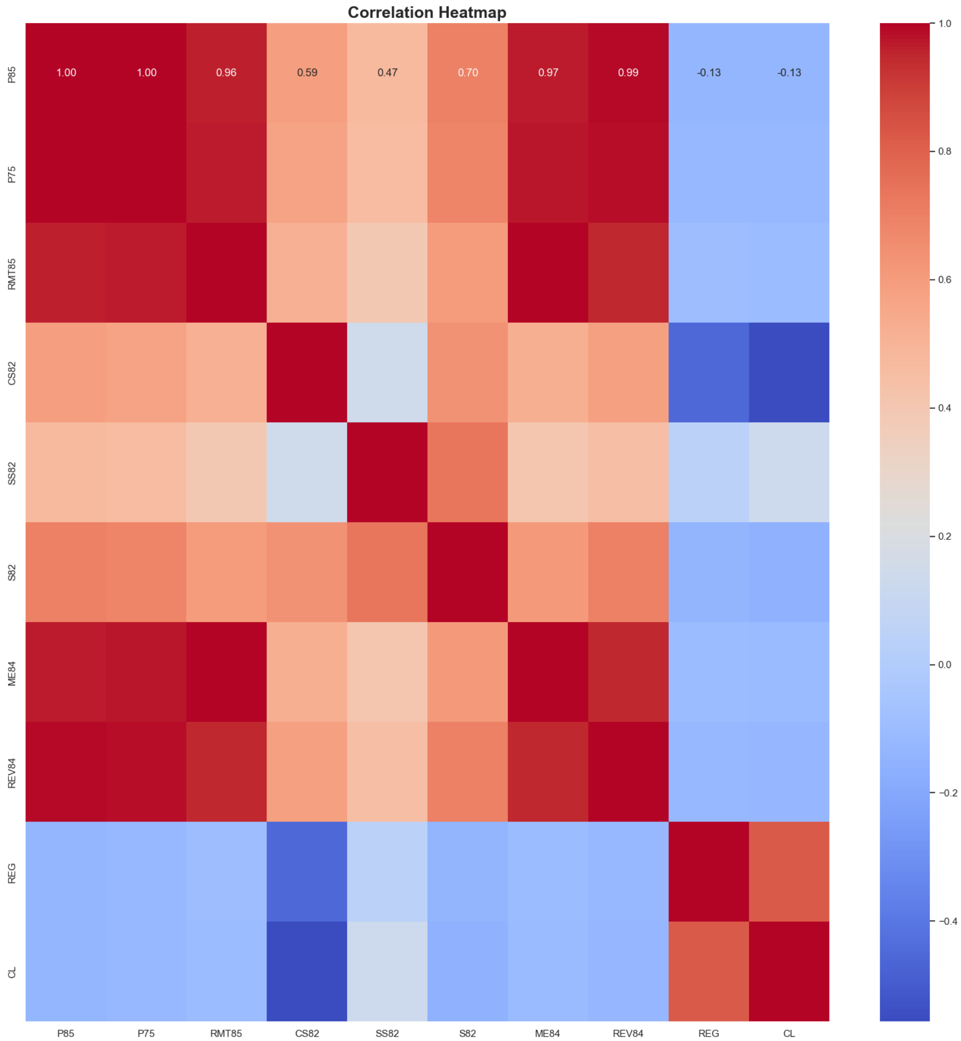

Figure 33: Correlation heatmap.

The boxplot of numerical features is presented on chart(s) below.

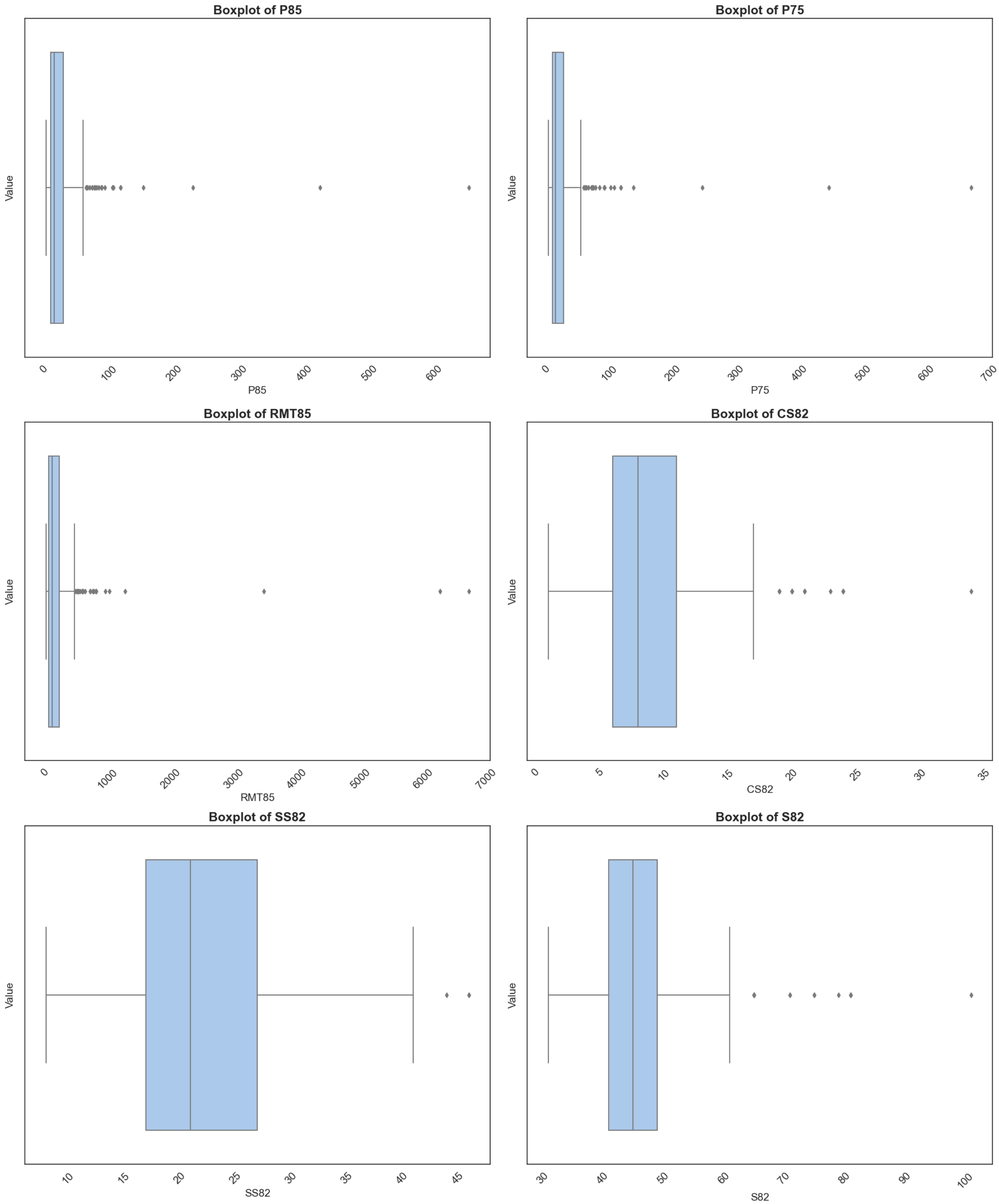

Figure 34: Boxplot page 1

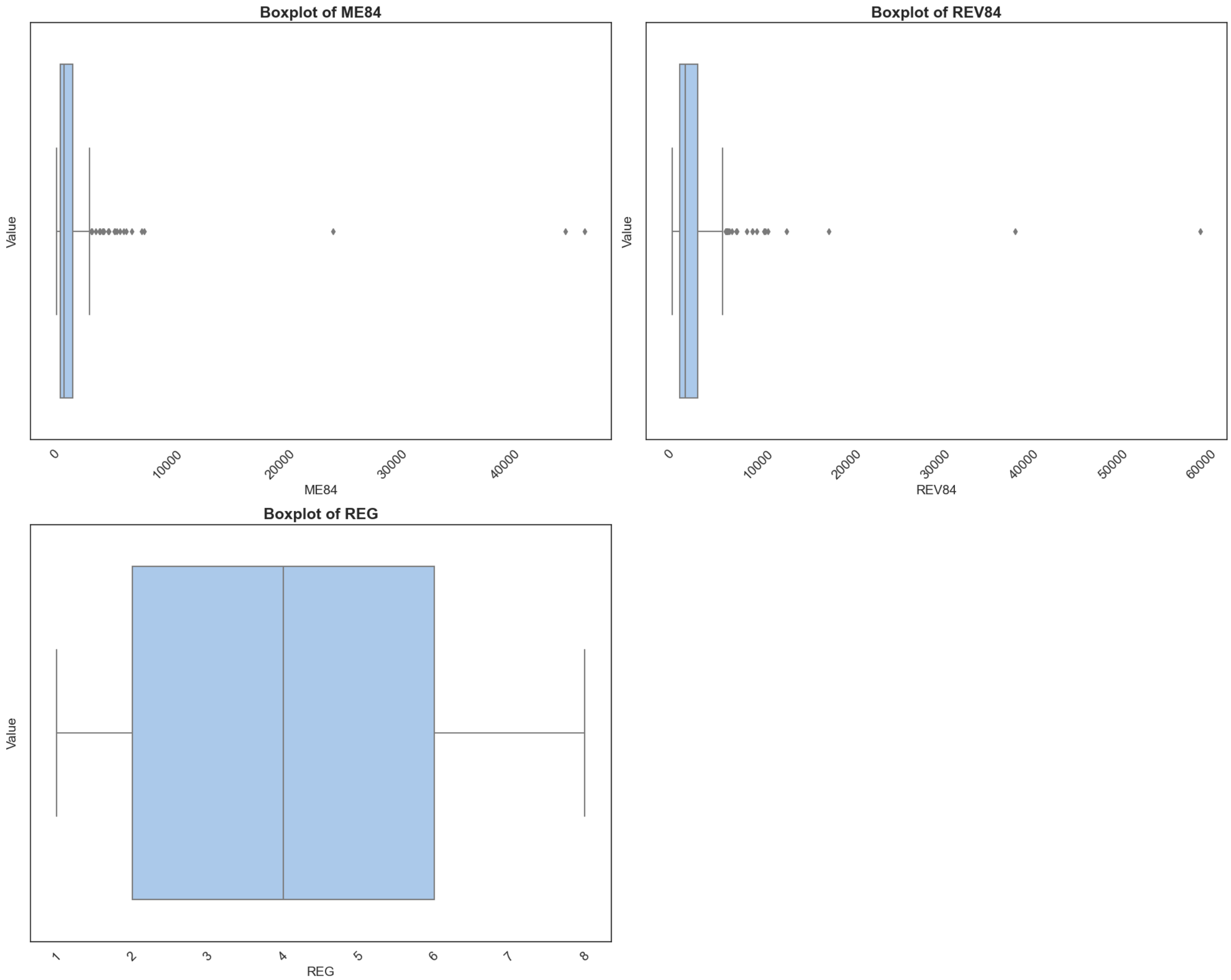

Figure 35: Boxplot page 2

11 Preprocessing

This part of the report presents the results of the preprocessing process. It contains required, as well as non required, steps listed below.

Required preprocessing steps:

- Missing data imputation
- Removing columns with 100% unique categorical values
- Categorical features encoding
- Scaling
- Removing columns with 0 variance
- Detecting highly correlatd features

Additional preprocessing steps:

- Feature selection methods : Correlation with the target or Random Forest feature importance
- Dimention reduction techniques: PCA, VIF, UMAP

Preprocessing process was configured to select up to 3 best unique preprocessing pipelines. Pipelines were scored based on a simple model. Tables below show detailed description of the best pipelines as well as all step combinations that were examined.

index	steps
0	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler
1	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, CorrelationSelector
2	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, FeatureImportanceRegressSelector
3	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, FeatureImportanceClassSelector
4	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, PCADimentionReducer
5	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, CorrelationSelector, PCADimentionReducer
6	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, FeatureImportanceRegressSelector, PCADimentionReducer
7	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, FeatureImportanceClassSelector, PCADimentionReducer
8	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, UMAPDimentionReducer
9	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, CorrelationSelector, UMAPDimentionReducer
10	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, FeatureImportanceRegressSelector, UMAPDimentionReducer
11	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, FeatureImportanceClassSelector, UMAPDimentionReducer
12	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, VIFDimentionReducer
13	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, CorrelationSelector, VIFDimentionReducer
14	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, FeatureImportanceRegressSelector, VIFDimentionReducer
15	NAImputer, UniqueFilter, ColumnEncoder, VarianceFilter, CorrelationFilter, ColumnScaler, FeatureImportanceClassSelector, VIFDimentionReducer

Table 50: Pipelines steps overview.

index	file name	score	fit duration	score duration
0	preprocessing_pipeline_0.joblib	192.7983	a moment	a moment
1	preprocessing_pipeline_1.joblib	192.7983	a moment	a moment
2	preprocessing_pipeline_2.joblib	189.4548	a second	a moment

Table 51: Best preprocessing pipelines.

step	name	description	params
0	NAImputer	Imputes missing data.	{"numeric_imputer": "median", "categorical_imputer": "most_frequent"}
1	UniqueFilter	Removes categorical columns with 100% unique values. Dropped columns: []	{}
2	ColumnEncoder	Encodes categorical columns using OneHotEncoder (for columns with <5 unique values) or TolerantLabelEncoder (for columns with >=5 unique values). Encodes target variable using LabelEncoder if provided.	{}
3	VarianceFilter	Removes columns with zero variance. Dropped columns: []	{}
4	CorrelationFilter	Removes one column from pairs of columns correlated above correlation threshold: 0.8.	{}
5	ColumnScaler	Scales numerical columns using one of 3 scaling methods.	{"method": "minmax"}
6	FeatureImportanceRegressSelector	Selects the top 10.0% (rounded to whole number) of features most important according to Random Forest model for regression. Number of features that were selected: 0	{"k": 10.0}
7	PCADimentionReducer	Combines PCA with automatic selection of the number of components to preserve 95% of the variance.	{"n_components": null}

Table 52: Best pipeline No. 0: steps overview.

index	count	mean	std	min	25%	50%	75%	max
P85	227.0000	-0.0000	1.0022	-0.4816	-0.3567	-0.2496	0.0002	11.1162
CS82	227.0000	0.0000	1.0022	-1.6443	-0.6387	-0.2365	0.3668	4.9921
SS82	227.0000	0.0000	1.0022	-1.9344	-0.6865	-0.1319	0.7000	3.3343
S82	227.0000	-0.0000	1.0022	-1.5314	-0.5831	-0.2038	0.1754	5.1062
REG	227.0000	-0.0000	1.0022	-1.6043	-1.1226	-0.1592	0.8043	1.7677

Table 53: Best pipeline No. 0: output overview.

step	name	description	params
0	NAImputer	Imputes missing data.	{"numeric_imputer": "median", "categorical_imputer": "most_frequent"}
1	UniqueFilter	Removes categorical columns with 100% unique values. Dropped columns: []	{}
2	ColumnEncoder	Encodes categorical columns using OneHotEncoder (for columns with <5 unique values) or TolerantLabelEncoder (for columns with >=5 unique values). Encodes target variable using LabelEncoder if provided.	{}
3	VarianceFilter	Removes columns with zero variance. Dropped columns: []	{}
4	CorrelationFilter	Removes one column from pairs of columns correlated above correlation threshold: 0.8.	{}
5	ColumnScaler	Scales numerical columns using one of 3 scaling methods.	{"method": "minmax"}
6	FeatureImportanceClassSelector	Selects the top 10.0% (rounded to whole number) of features most important according to Random Forest model for classification. Number of features that were selected: 0	{"k": 10.0}
7	PCADimentionReducer	Combines PCA with automatic selection of the number of components to preserve 95% of the variance.	{"n_components": null}

Table 54: Best pipeline No. 1: steps overview.

index	count	mean	std	min	25%	50%	75%	max
P85	227.0000	0.0415	0.0864	0.0000	0.0108	0.0200	0.0415	1.0000
CS82	227.0000	0.2478	0.1510	0.0000	0.1515	0.2121	0.3030	1.0000
SS82	227.0000	0.3671	0.1902	0.0000	0.2368	0.3421	0.5000	1.0000
S82	227.0000	0.2307	0.1510	0.0000	0.1429	0.2000	0.2571	1.0000
REG	227.0000	0.4758	0.2972	0.0000	0.1429	0.4286	0.7143	1.0000

Table 55: Best pipeline No. 1: output overview.

step	name	description	params
0	NAImputer	Imputes missing data.	{"numeric_imputer": "median", "categorical_imputer": "most_frequent"}
1	UniqueFilter	Removes categorical columns with 100% unique values. Dropped columns: []	{}
2	ColumnEncoder	Encodes categorical columns using OneHotEncoder (for columns with <5 unique values) or TolerantLabelEncoder (for columns with >=5 unique values). Encodes target variable using LabelEncoder if provided.	{}
3	VarianceFilter	Removes columns with zero variance. Dropped columns: []	{}
4	CorrelationFilter	Removes one column from pairs of columns correlated above correlation threshold: 0.8.	{}
5	ColumnScaler	Scales numerical columns using one of 3 scaling methods.	{"method": "robust"}
6	FeatureImportanceRegressSelector	Selects the top 10.0% (rounded to whole number) of features most important according to Random Forest model for regression. Number of features that were selected: 0	{"k": 10.0}
7	UMAPDimentionReducer	Reduces the dimensionality of the data using UMAP.	{"n_components": null}

Table 56: Best pipeline No. 2: steps overview.

index	count	mean	std	min	25%	50%	75%	max
P85	227.0000	0.6996	2.8084	-0.6500	-0.3000	0.0000	0.7000	31.8500
CS82	227.0000	0.2352	0.9967	-1.4000	-0.4000	0.0000	0.6000	5.2000
SS82	227.0000	0.0952	0.7228	-1.3000	-0.4000	0.0000	0.6000	2.5000
S82	227.0000	0.2687	1.3212	-1.7500	-0.5000	0.0000	0.5000	7.0000
REG	227.0000	0.0826	0.5201	-0.7500	-0.5000	0.0000	0.5000	1.0000

Table 57: Best pipeline No. 2: output overview.

Category	Value
Unique created pipelines	16
All created pipelines (after exploading each step params)	48
All pipelines fit time	19 seconds
All pipelines score time	19 seconds
scores_count	48.0000
scores_mean	116.7630
scores_std	71.0902
scores_min	23.8753
scores_25%	33.5481
scores_50%	146.7141
scores_75%	186.5825
scores_max	192.7983
Scoring function	function
Scoring model	RandomForestRegressor

Table 58: Preprocessing pipelines runtime statistics.

12 Modeling

12.1 Overview

This part of the report presents the results of the modeling process. There were 6 regression models trained for each of the best preprocessing pipelines.
The following models were used in the modeling process.

- LinearSVR
- KNeighborsRegressor
- RandomForestRegressor
- BayesianRidge
- GradientBoostingRegressor
- LinearRegression

12.2 Hyperparameter tuning

This section presents the results of hyperparameter tuning for each of the best 3 models using RandomizedSearchCV. Param grids used for each model are presented in the tables below.

Category	Value
epsilon	[0.0, 0.1, 0.2, 0.5, 1.0]
C	[0.1, 1.0, 10.0, 100.0]
loss	['epsilon_insensitive', 'squared_epsilon_insensitive']
fit_intercept	[True, False]

Table 59: Param grid for model LinearSVR.

Category	Value
n_neighbors	[5, 10, 15]
weights	['uniform', 'distance']
algorithm	['auto', 'ball_tree', 'kd_tree', 'brute']
leaf_size	[30, 40, 50]
p	[1, 2]

Table 60: Param grid for model KNeighboursRegressor.

Category	Value
n_estimators	[100, 200, 300]
max_depth	[None, 5, 10, 15, 20]
min_samples_split	[2, 5, 10]
min_samples_leaf	[1, 2, 4]
max_features	['sqrt', 'log2', None]
bootstrap	[True, False]
random_state	[42]

Table 61: Param grid for model RandomForestRegressor.

Category	Value
max_iter	[300, 400, 500]
tol	[0.001, 0.0001, 1e-05]
alpha_1	[1e-06, 1e-07, 1e-08]
alpha_2	[1e-06, 1e-07, 1e-08]
lambda_1	[1e-06, 1e-07, 1e-08]
lambda_2	[1e-06, 1e-07, 1e-08]

Table 62: Param grid for model BayesianRidgeRegressor.

Category	Value
n_estimators	[100, 200, 300]
learning_rate	[0.1, 0.05, 0.02]
max_depth	[4, 6, 8]
min_samples_split	[2, 5, 10]
min_samples_leaf	[1, 2, 4]
subsample	[1.0, 0.5]
random_state	[42]

Table 63: Param grid for model GradientBoostingRegressor.

Category	Value
fit_intercept	[True, False]

Table 64: Param grid for model LinearRegression.

Table 65 presents the best models and pipelines along with their hyperparameters, mean fit time, and test score.

Model	Pipeline	Best params	Mean fit time	Test score
LinearSVR	final_pipeline_1.joblib	{"loss": "epsilon_insensitive", "fit_intercept": true, "epsilon": 0.0, "C": 0.1}	a moment	384.6749
LinearSVR	final_pipeline_0.joblib	{"loss": "epsilon_insensitive", "fit_intercept": false, "epsilon": 0.2, "C": 1.0}	a moment	872.7019
LinearSVR	final_pipeline_2.joblib	{"loss": "epsilon_insensitive", "fit_intercept": false, "epsilon": 0.0, "C": 1.0}	a moment	872.7019

Table 65: Best models results

12.3 Interpretability

This section presents SHAP plots for the best model.

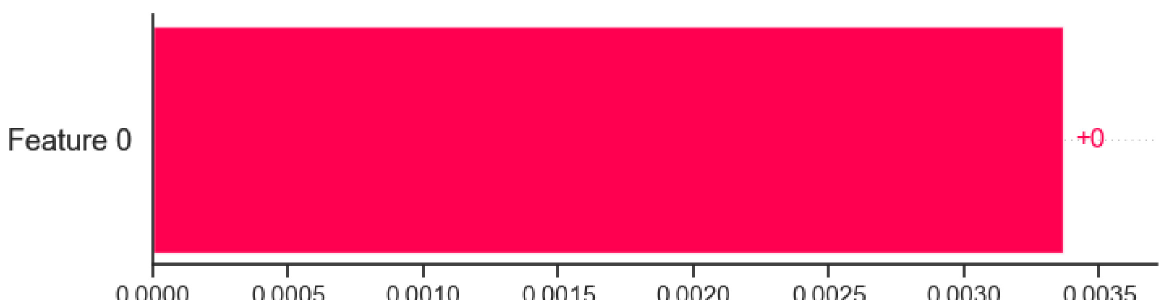

Figure 36: SHAP bar plot.

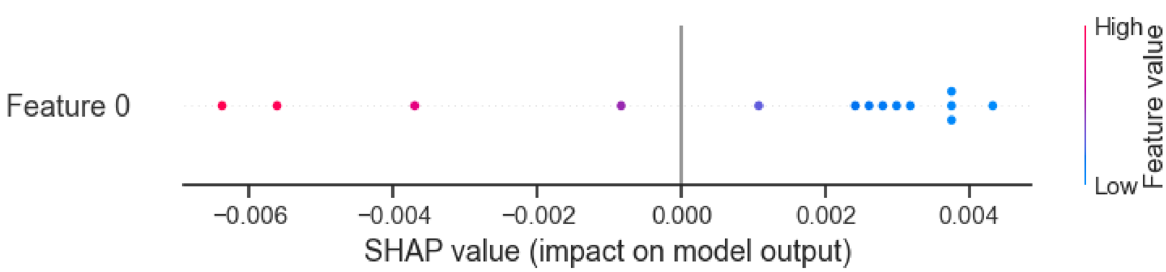

Figure 37: SHAP summary plot.

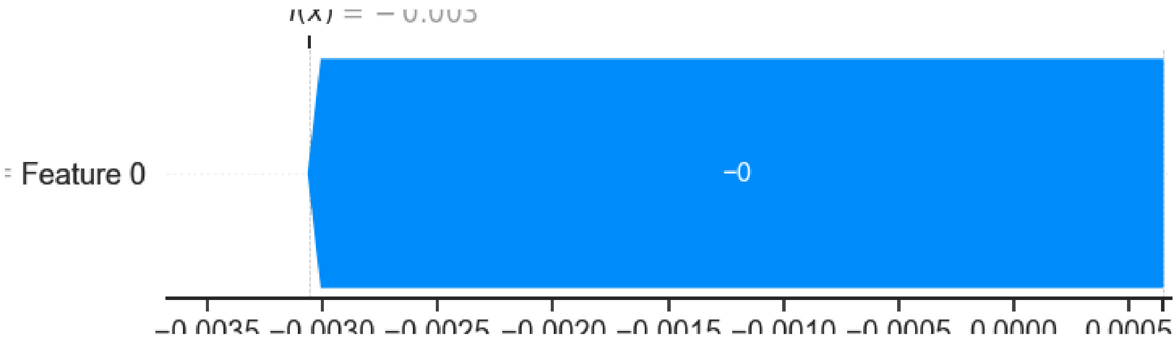

Figure 38: SHAP waterfall plot.