ML Raport

AutoPrep

January 12, 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	3.11.10
Physical Cores	8
Logical Cores	8
CPU Frequency (MHz)	4056
Total RAM (GB)	16.00
Available RAM (GB)	4.59
Total Disk Space (GB)	460.43
Free Disk Space (GB)	243.55

Table 1: System overview.

1.2 Dataset

Task detected for the dataset: binary classfication.
Table 2 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 3

class	**number of observations**	**fraction**
0	665	0.64
1	382	0.36

Table 3: Target class distribution.

Table 4 presents the distribution of missing values in the dataset.

feature	**number of observations**	**fraction**
pclass	0	0.00
name	0	0.00
sex	0	0.00
age	207	0.20
sibsp	0	0.00
parch	0	0.00
ticket	0	0.00
fare	1	0.00
cabin	813	0.78
embarked	1	0.00
boat	672	0.64
body	948	0.91
home___dest	453	0.43

Table 4: Missing values distribution.

Table 5 presents the description of features in the dataset.

feature	**type**	**dtype**	**space usage**
pclass	numerical	int64	16.8 kB
name	categorical	object	96.4 kB
sex	categorical	category	9.7 kB
age	numerical	float64	16.8 kB
sibsp	numerical	int64	16.8 kB
parch	numerical	int64	16.8 kB
ticket	categorical	object	75.1 kB
fare	numerical	float64	16.8 kB
cabin	categorical	object	48.6 kB
embarked	categorical	category	9.7 kB
boat	categorical	object	51.8 kB
body	numerical	float64	16.8 kB
home___dest	categorical	object	68.2 kB

Table 5: Features dtypes description.

Table 6 and Table 7 present the description of numerical and categorical features in the dataset.

feature	count	mean	std	min	25%	50%	75%	max
pclass	1047.00	2.30	0.84	1.00	2.00	3.00	3.00	3.00
age	840.00	29.53	14.27	0.17	21.00	28.00	38.62	80.00
sibsp	1047.00	0.52	1.05	0.00	0.00	0.00	1.00	8.00
parch	1047.00	0.40	0.89	0.00	0.00	0.00	0.00	9.00
fare	1046.00	33.55	51.81	0.00	7.92	14.50	31.27	512.33
body	99.00	160.90	98.35	1.00	73.50	156.00	255.50	328.00

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 1 shows the distribution of the target variable.

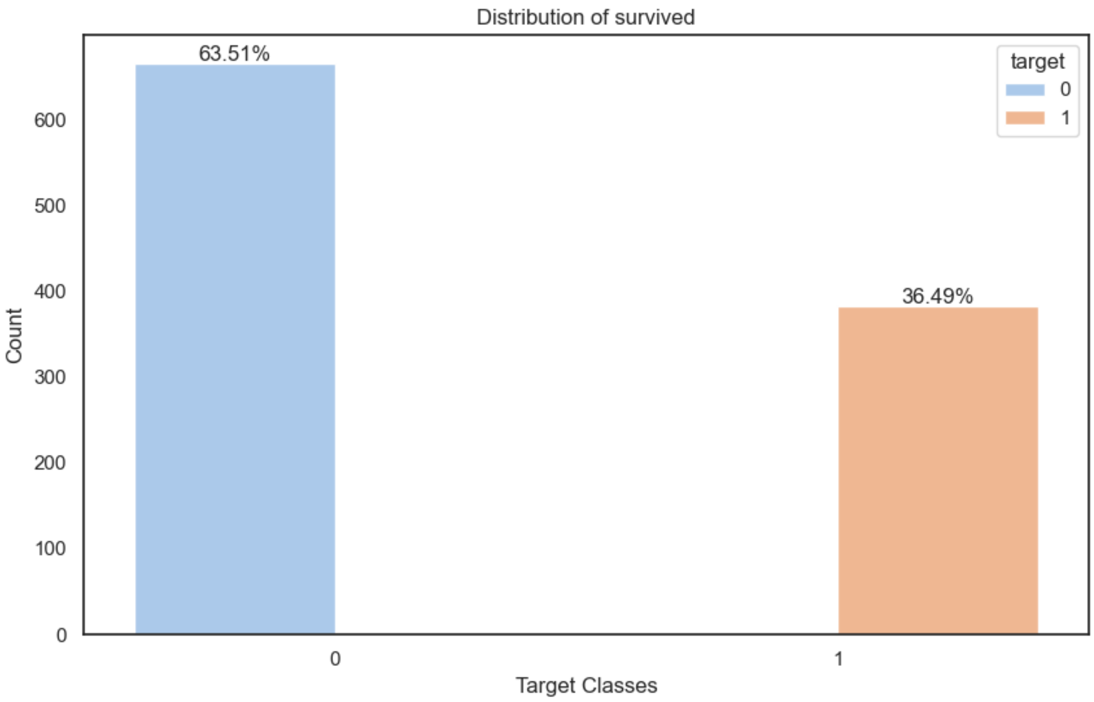

Figure 1: Target distribution.

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

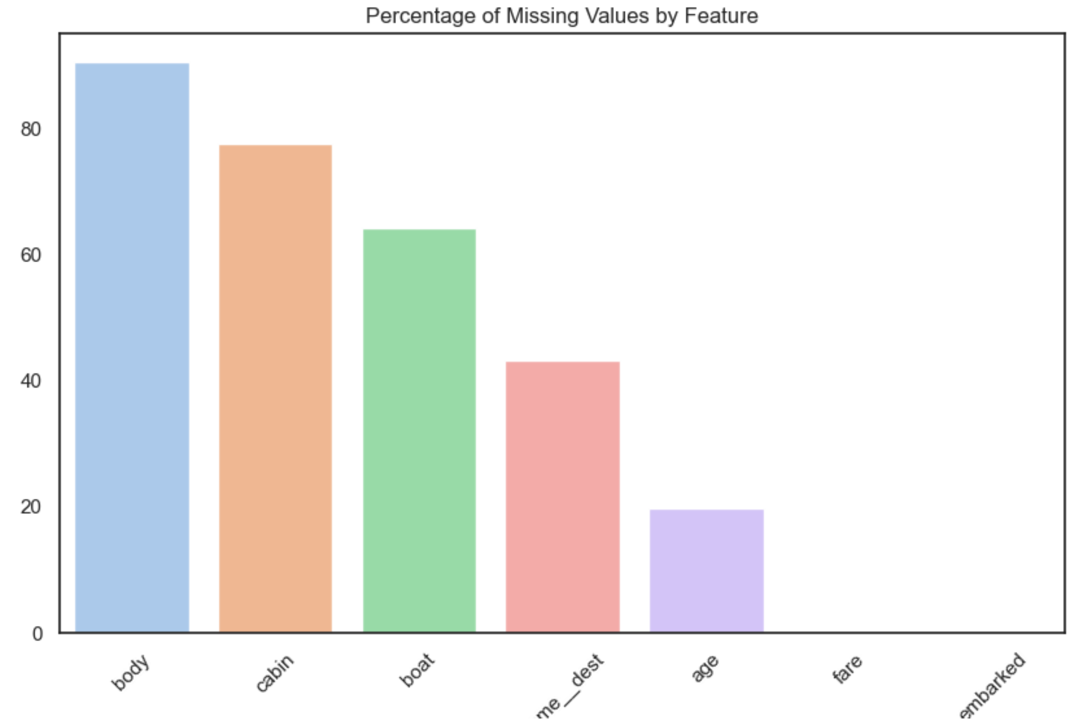

Figure 2: Missing values.

2.2 EDA for categorical features

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

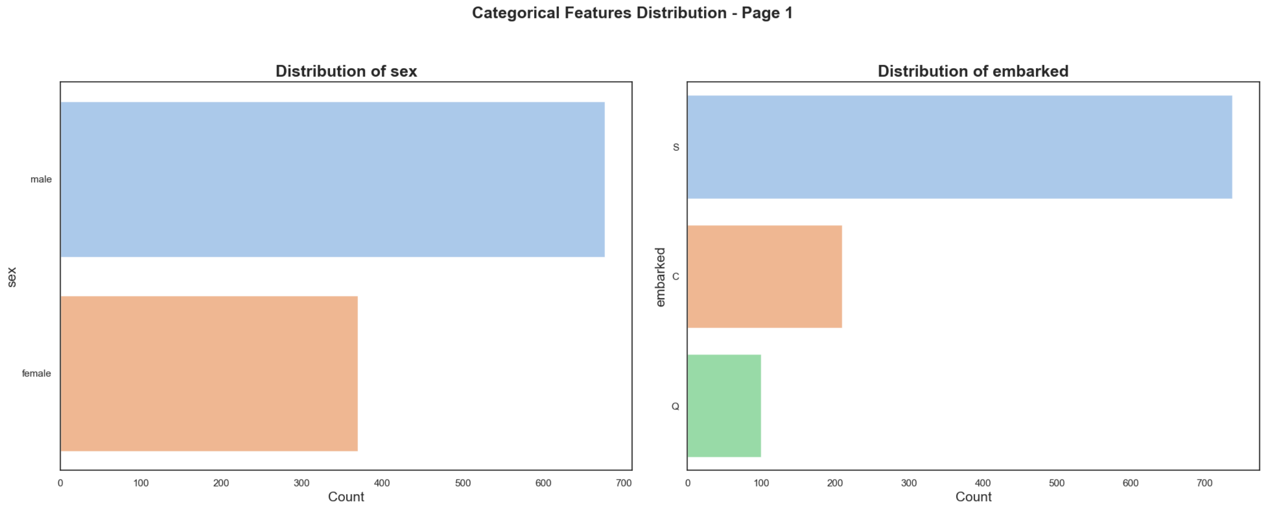

Figure 3: Categorical Features Distribution - Page 1

2.3 EDA for numerical features

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

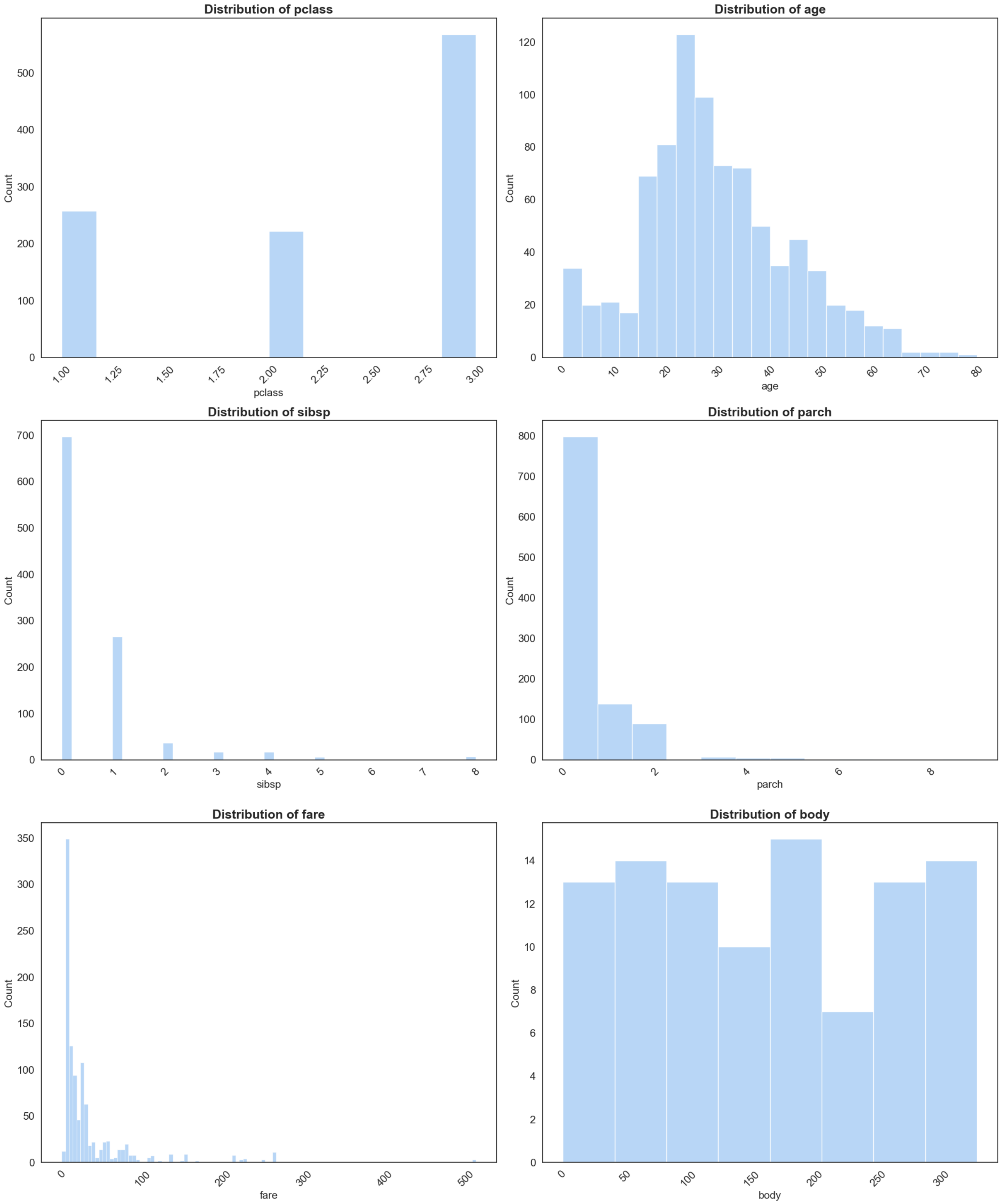

Figure 4: Numerical Features Distribution - Page 1

Figure 5 shows the correlation between features.

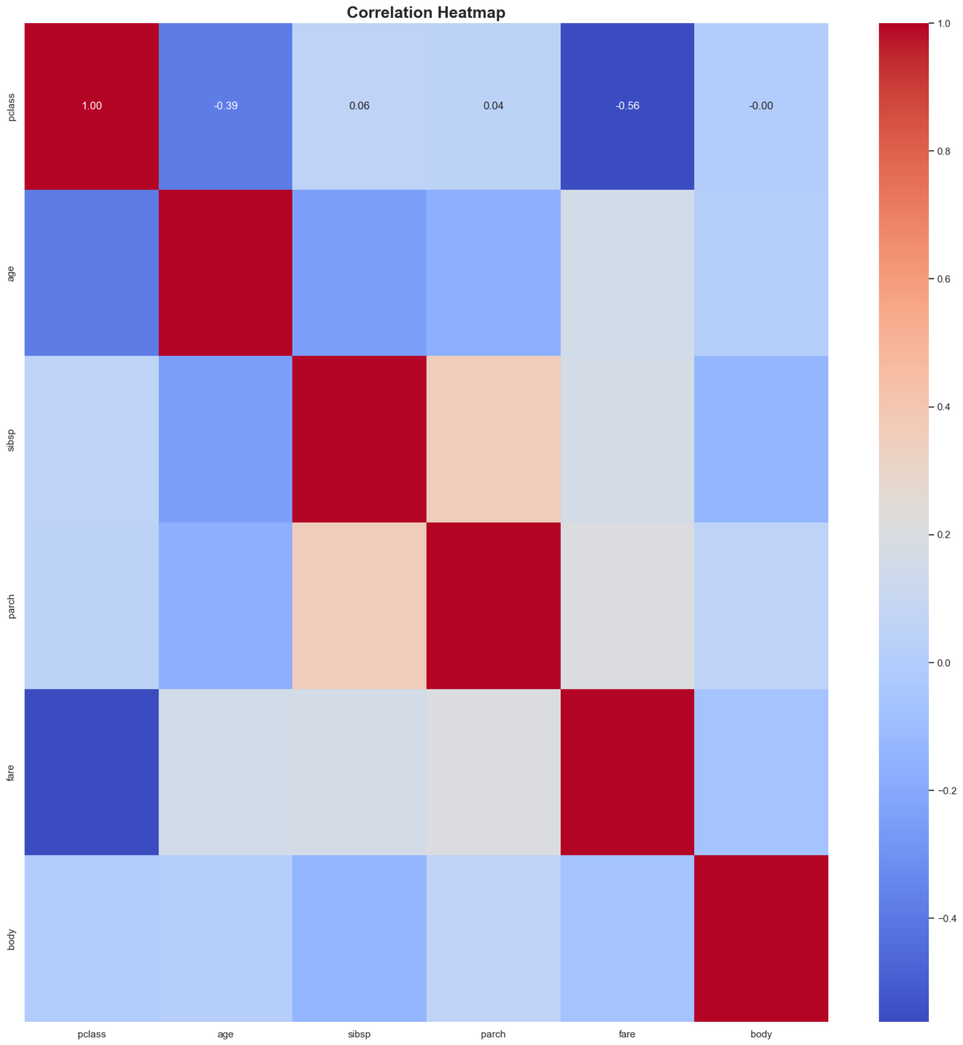

Figure 5: Correlation heatmap.

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

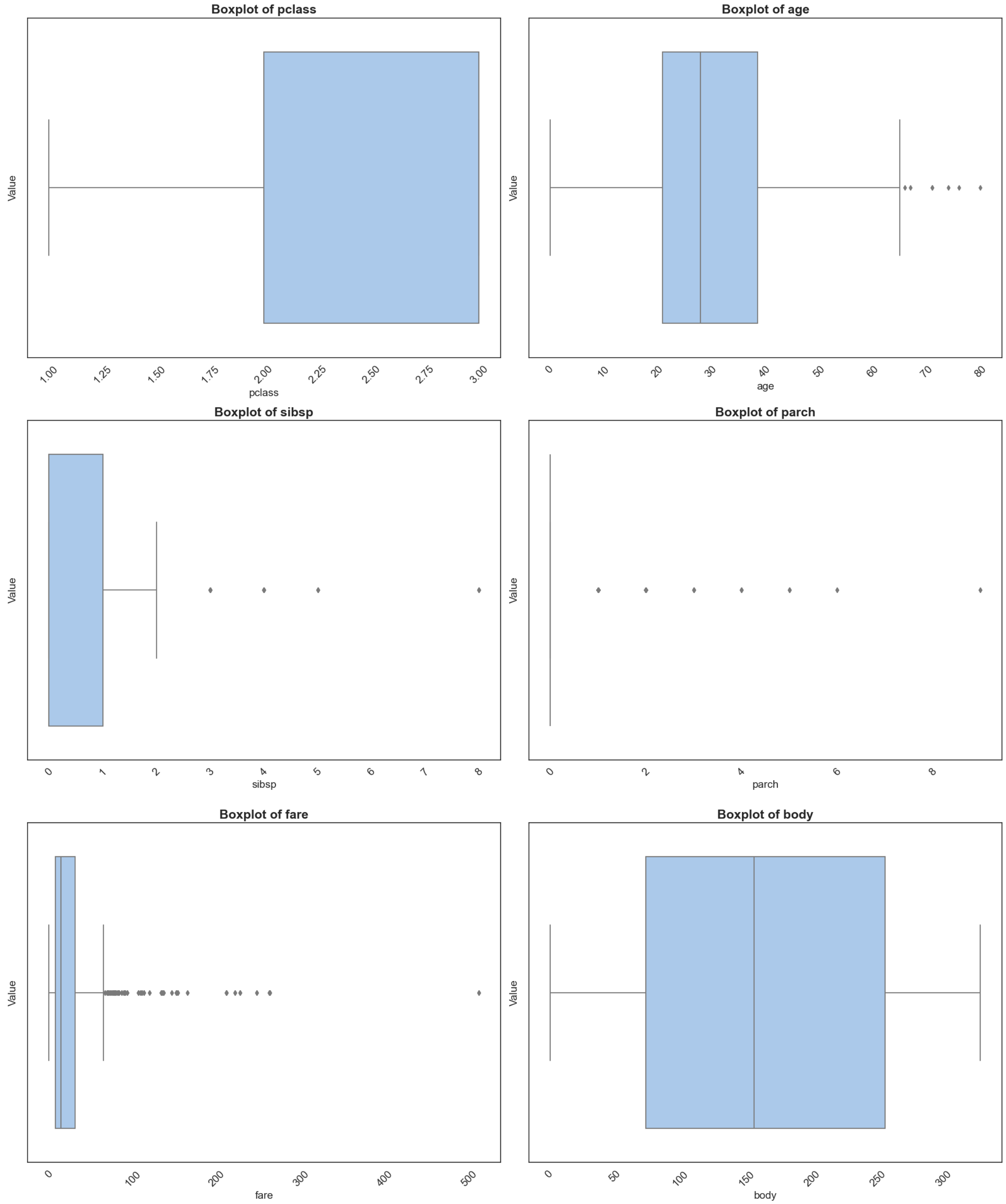

Figure 6: Boxplot page 1

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

Table 8: Pipelines steps overview.

index	file name	score	fit duration	score duration
0	preprocessing_pipeline_0.joblib	0.75	a moment	a moment
1	preprocessing_pipeline_1.joblib	0.75	a moment	a moment
2	preprocessing_pipeline_2.joblib	0.75	a moment	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"}

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

index	**count**	**mean**	**std**	**min**	**25%**	**50%**	**75%**	**max**
pclass	1047.00	0.00	1.00	-1.55	-0.36	0.84	0.84	0.84
name	1047.00	0.00	1.00	-1.73	-0.87	-0.00	0.87	1.73
age	1047.00	-0.00	1.00	-2.27	-0.57	-0.10	0.45	3.97
sibsp	1047.00	-0.00	1.00	-0.50	-0.50	-0.50	0.46	7.13
parch	1047.00	0.00	1.00	-0.44	-0.44	-0.44	-0.44	9.63
ticket	1047.00	-0.00	1.00	-1.68	-0.90	0.00	0.93	1.67
fare	1047.00	0.00	1.00	-0.65	-0.49	-0.37	-0.04	9.25
home__dest	1047.00	0.00	1.00	-2.74	-0.22	0.30	0.30	1.94
sex_female	1047.00	0.00	1.00	-0.74	-0.74	-0.74	1.35	1.35
embarked_C	1047.00	-0.00	1.00	-0.50	-0.50	-0.50	-0.50	2.00
embarked_Q	1047.00	0.00	1.00	-0.32	-0.32	-0.32	-0.32	3.08
embarked_S	1047.00	-0.00	1.00	-1.55	-1.55	0.65	0.65	0.65

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": "minmax"}

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

index	count	mean	std	min	25%	50%	75%	max
pclass	1047.00	0.65	0.42	0.00	0.50	1.00	1.00	1.00
name	1047.00	0.50	0.29	0.00	0.25	0.50	0.75	1.00
age	1047.00	0.36	0.16	0.00	0.27	0.35	0.44	1.00
sibsp	1047.00	0.07	0.13	0.00	0.00	0.00	0.12	1.00
parch	1047.00	0.04	0.10	0.00	0.00	0.00	0.00	1.00
ticket	1047.00	0.50	0.30	0.00	0.23	0.50	0.78	1.00
fare	1047.00	0.07	0.10	0.00	0.02	0.03	0.06	1.00
home___dest	1047.00	0.59	0.21	0.00	0.54	0.65	0.65	1.00
sex_female	1047.00	0.35	0.48	0.00	0.00	0.00	1.00	1.00
embarked_C	1047.00	0.20	0.40	0.00	0.00	0.00	0.00	1.00
embarked_Q	1047.00	0.10	0.29	0.00	0.00	0.00	0.00	1.00
embarked_S	1047.00	0.70	0.46	0.00	0.00	1.00	1.00	1.00

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"}

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

index	**count**	**mean**	**std**	**min**	**25%**	**50%**	**75%**	**max**
pclass	1047.00	-0.70	0.84	-2.00	-1.00	0.00	0.00	0.00
name	1047.00	0.00	0.58	-1.00	-0.50	0.00	0.50	1.00
age	1047.00	0.09	0.98	-2.14	-0.46	0.00	0.54	4.00
sibsp	1047.00	0.52	1.05	0.00	0.00	0.00	1.00	8.00
parch	1047.00	0.40	0.89	0.00	0.00	0.00	0.00	9.00
ticket	1047.00	-0.00	0.55	-0.92	-0.49	0.00	0.51	0.91
fare	1047.00	0.81	2.22	-0.62	-0.28	0.00	0.72	21.32
home___dest	1047.00	-0.57	1.93	-5.86	-1.00	0.00	0.00	3.17
sex_female	1047.00	0.35	0.48	0.00	0.00	0.00	1.00	1.00
embarked_C	1047.00	0.20	0.40	0.00	0.00	0.00	0.00	1.00
embarked_Q	1047.00	0.10	0.29	0.00	0.00	0.00	0.00	1.00
embarked_S	1047.00	-0.30	0.46	-1.00	-1.00	0.00	0.00	0.00

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

Category	**Value**
Unique created pipelines	1
All created pipelines (after exploading each step params)	3
All pipelines fit time	a second
All pipelines score time	a second
scores_count	3.00
scores_mean	0.75
scores_std	0.00
scores_min	0.75
scores_25%	0.75
scores_50%	0.75
scores_75%	0.75
scores_max	0.75
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 22 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	0.73
KNeighborsClassifier	final_pipeline_1.joblib	{"weights": "distance", "p": 2, "n_neighbors": 10, "leaf_size": 40, "algorithm": "auto"}	a moment	0.73
KNeighborsClassifier	final_pipeline_0.joblib	{"weights": "distance", "p": 1, "n_neighbors": 15, "leaf_size": 30, "algorithm": "brute"}	a moment	0.70

Table 22: Best models results

4.3 Interpretability

This section presents SHAP plots for the best model.

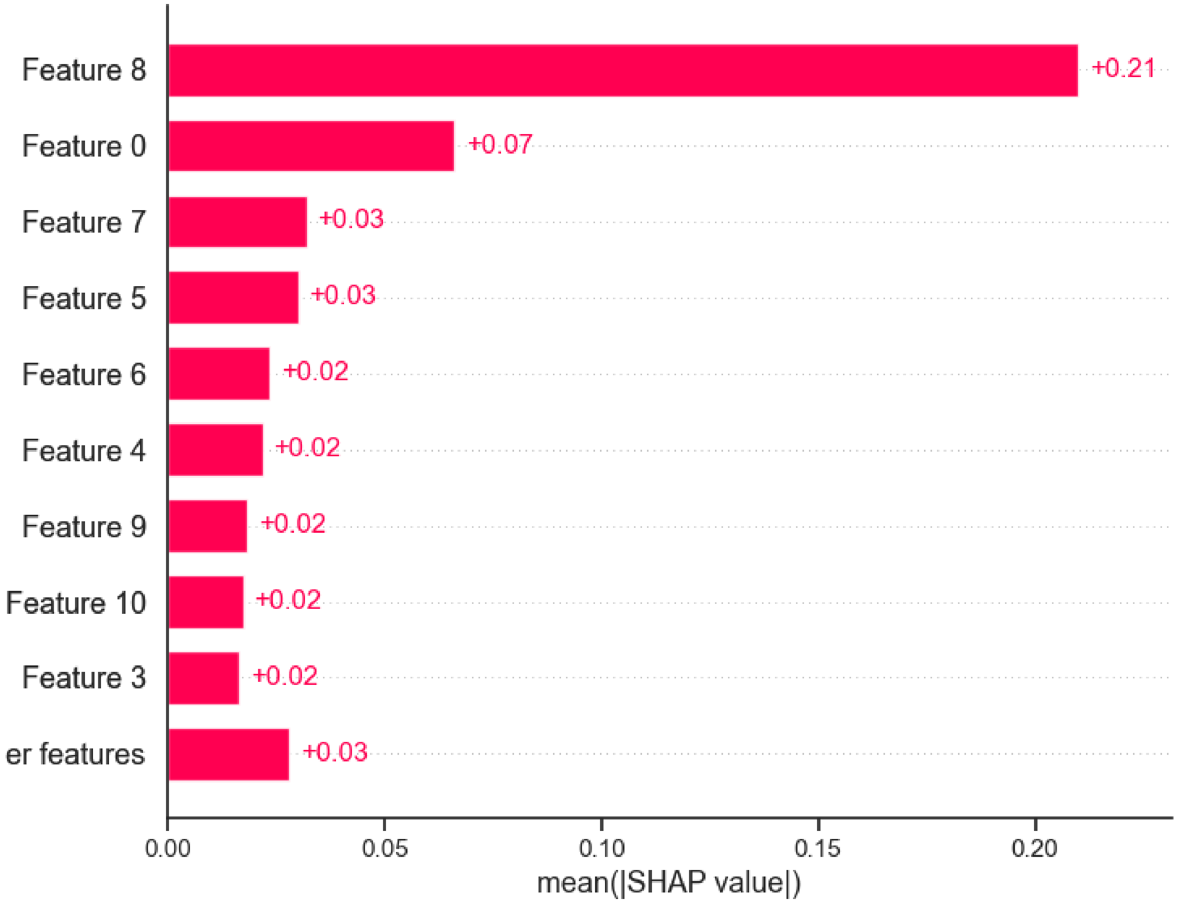

Figure 7: SHAP bar plot for class class.

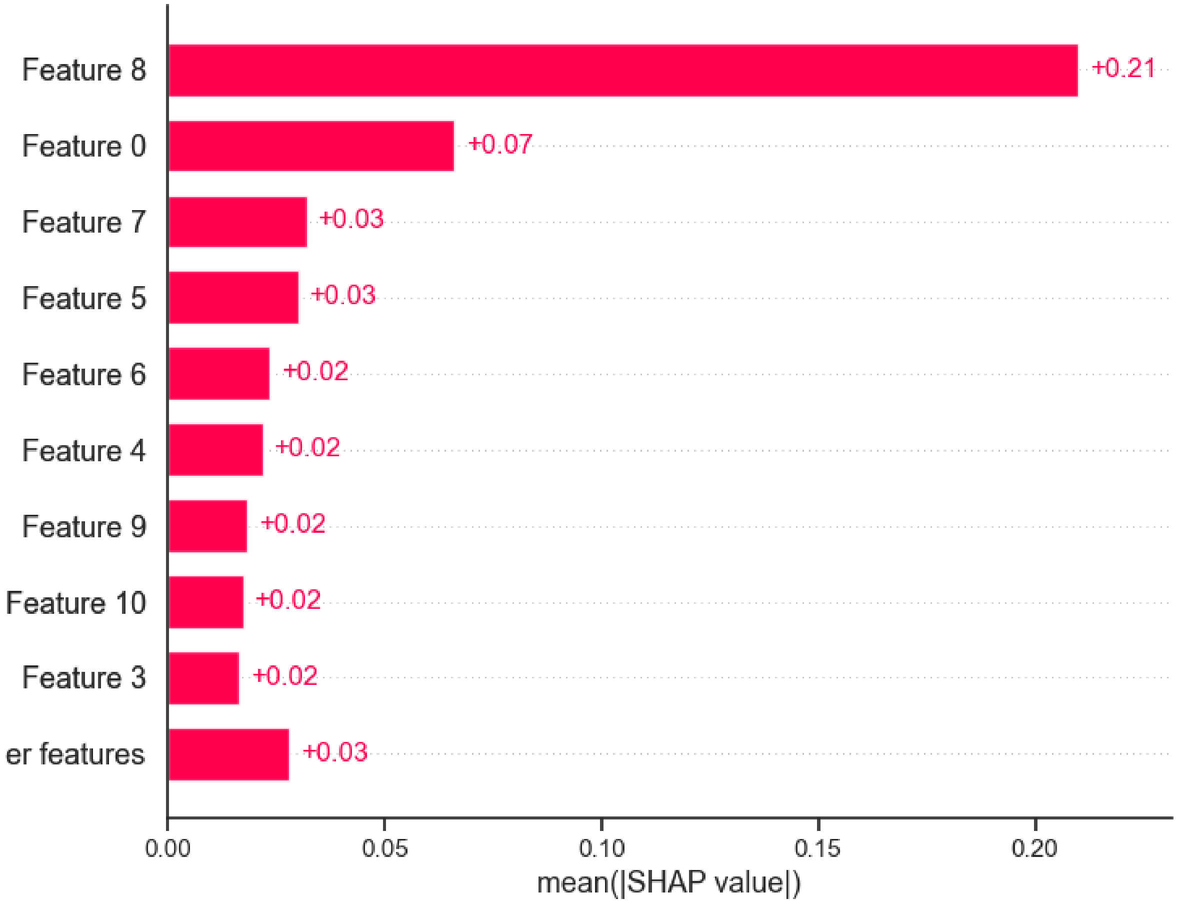

Figure 8: SHAP bar plot for class class.

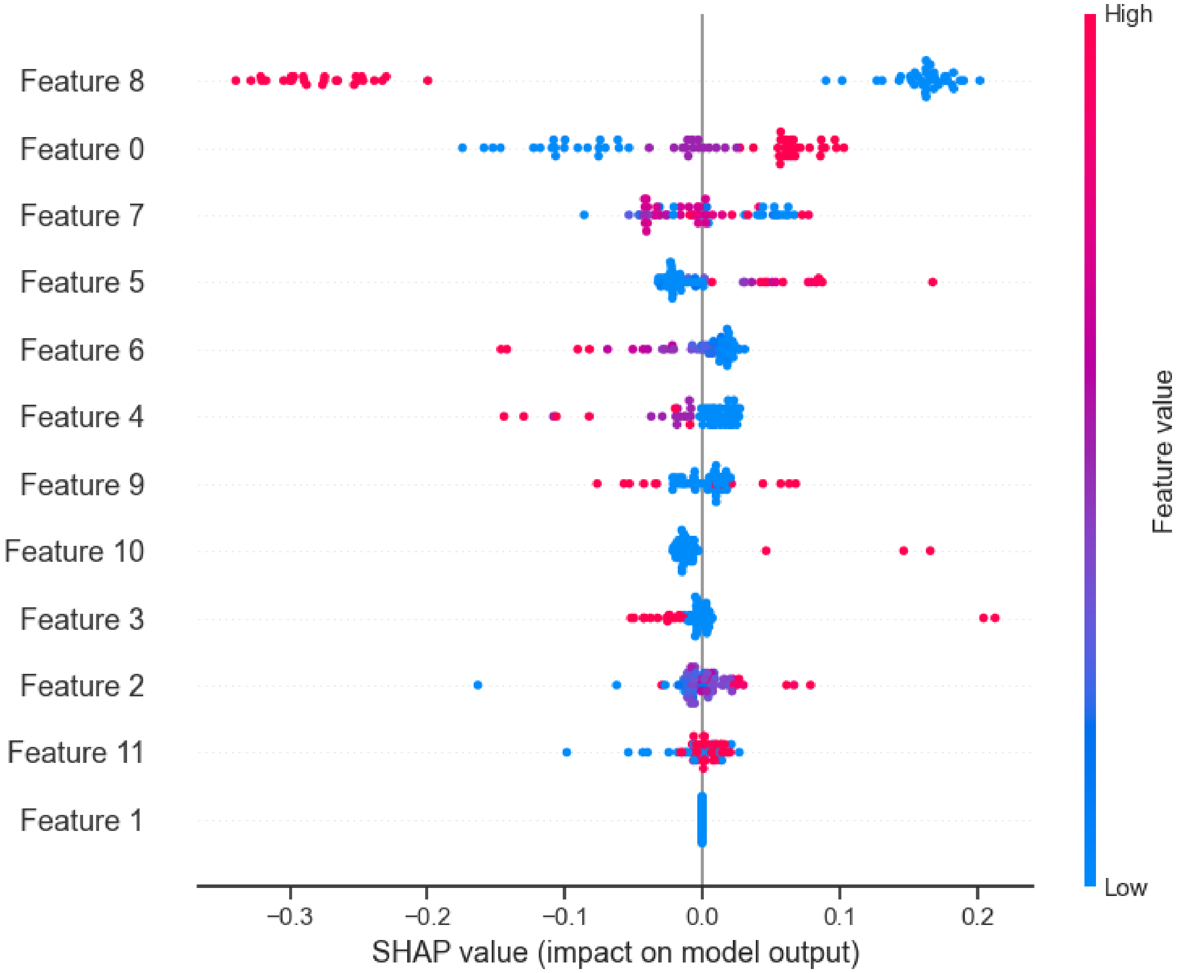

Figure 9: SHAP summary plot for class class.

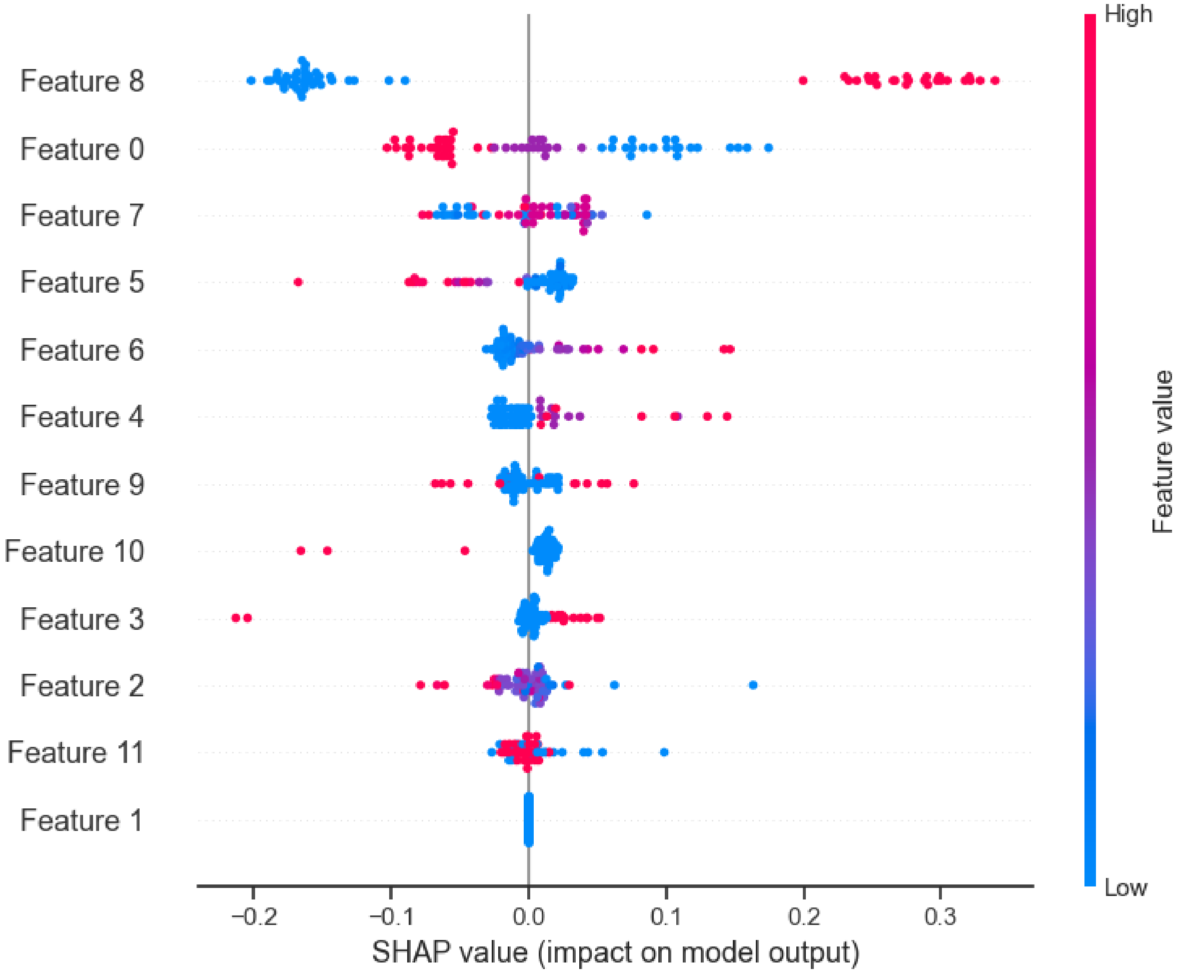

Figure 10: SHAP summary plot for class class.

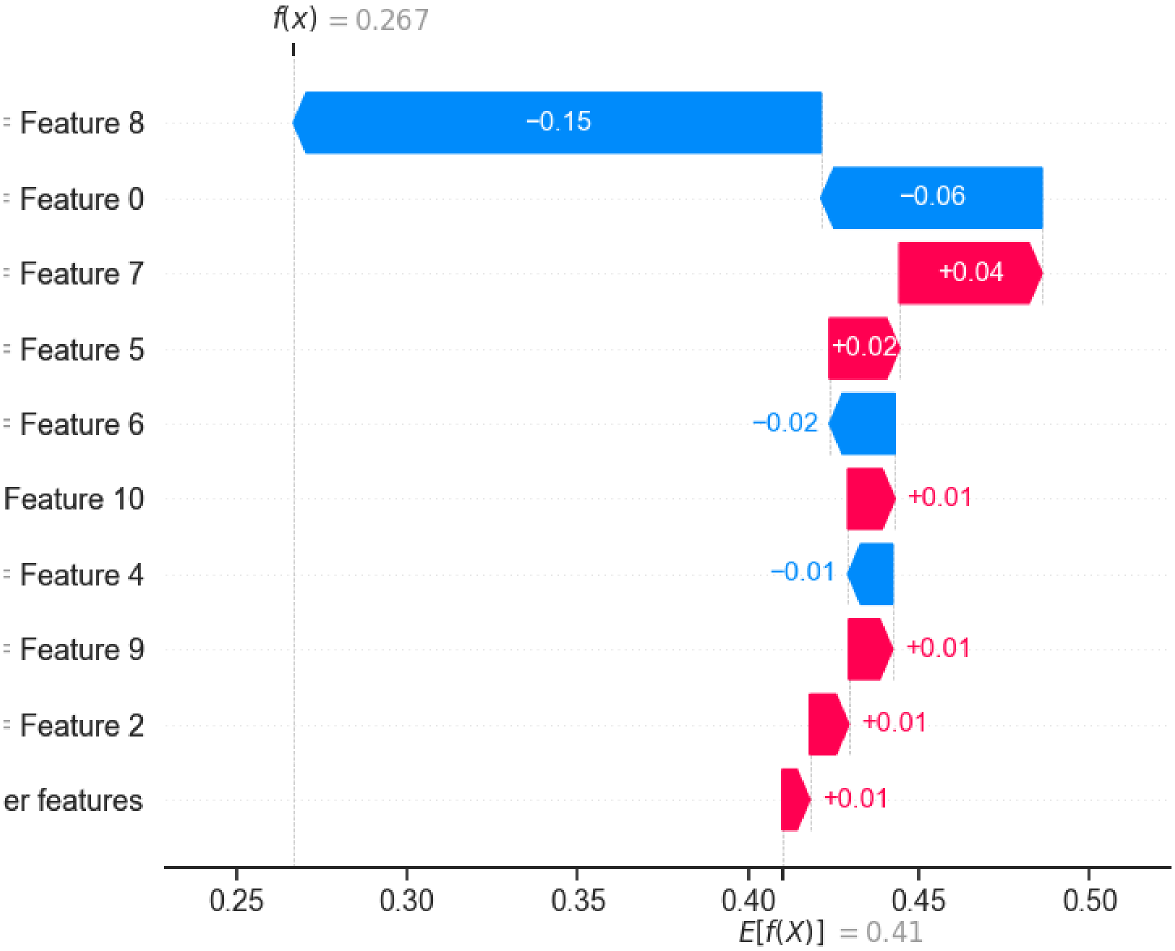

Figure 11: SHAP waterfall plot for class class.

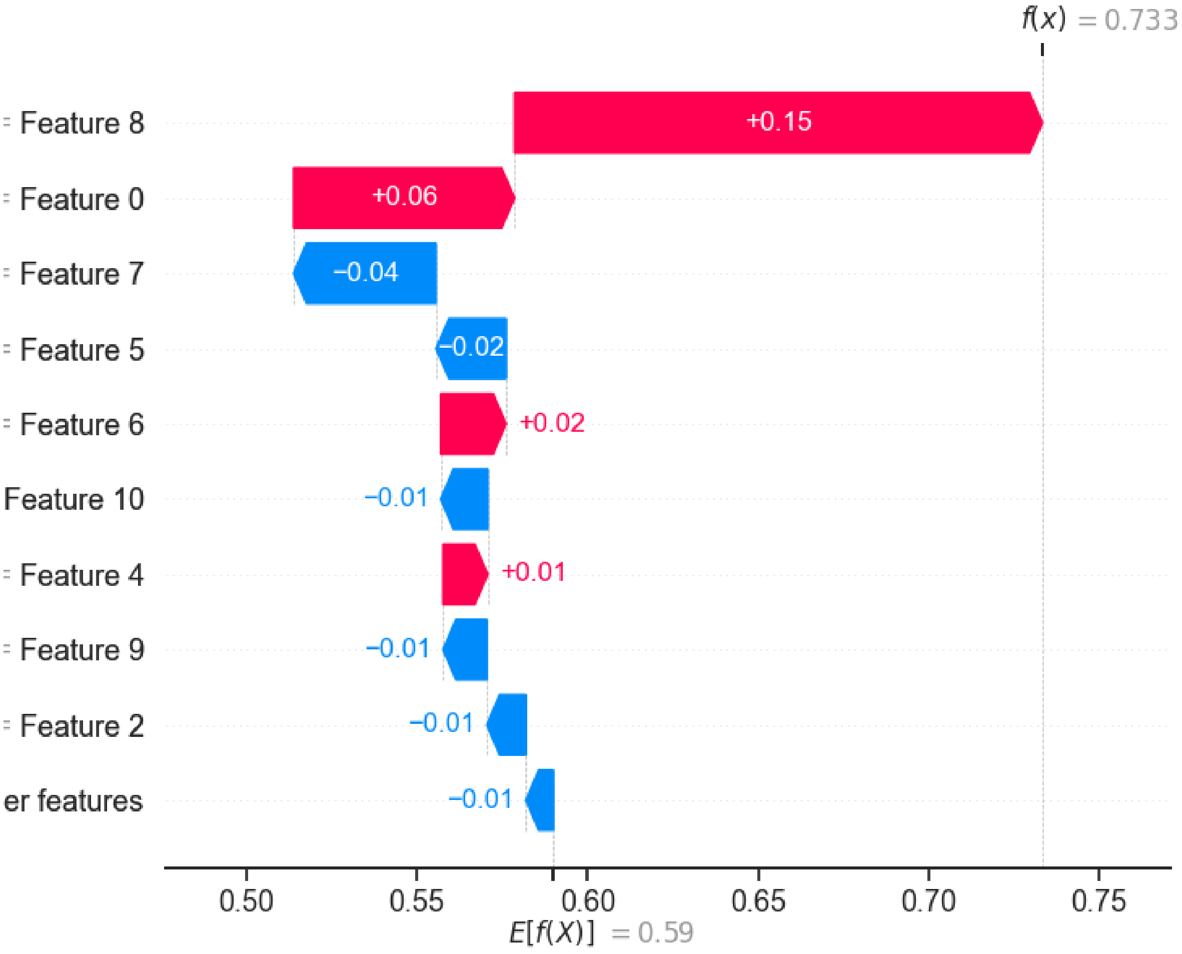

Figure 12: SHAP waterfall plot for class class.