Università di Genova | DIBRIS DIPARTIMENTO DI INFORMATICA, BIOINGEGNERIA, ROBOTICA E INGEGNERIA DEI SISTEMI

Weather Prediction

Machine Learning and Data Analysis

Samuele Crea (4868097) & Marco Zucca (4828628)

Università di Genova
DIBRIS DIPARTIMENTO
DI INFORMATICA, BIOINGEGNERIA,
ROBOTICA E INGEGNERIA DEI SISTEMI

Data Analysis

Exploration, cleaning and analysis of the dataset

Dataset exploration and cleaning

Dataset exploration and cleaning

Slice of the original dataset

	DATE	MONTH	BASEL_cloud_cover	BASEL_humidity	BASEL_pressure	BASEL_global_radiation	BASEL_precipitation	BASEL_sunshine	BASEL_temp_mean	BASEL_temp_min	...	STOCKHOLM_temp_min
0	20000101	1	8	0.89	1.0286	0.20	0.03	0.0	2.9	1.6	...	-9.3
1	20000102	1	8	0.87	1.0318	0.25	0.00	0.0	3.6	2.7	...	0.5
2	20000103	1	5	0.81	1.0314	0.50	0.00	3.7	2.2	0.1	...	-1.0
3	20000104	1	7	0.79	1.0262	0.63	0.35	6.9	3.9	0.5	...	2.5
4	20000105	1	5	0.90	1.0246	0.51	0.07	3.7	6.0	3.8	...	-1.8

- Includes measurements of meteorological data from 18 different European cities
- obtained for each day from 2000 to 2009 included
- Several collected features that allow the study of weather

Dataset exploration and cleaning

Features per city

Feature (type)	Column name	Description	Physical Unit
mean temperature	temp_mean	mean daily temperature	in 1 °C
max temperature	temp_max	max daily temperature	in 1 °C
min temperature	temp_min	min daily temperature	in 1 °C
cloud_cover	cloud_cover	cloud cover	oktas
global_radiation	global_radiation	global radiation	in 100 W/m2
humidity	humidity	humidity	in 1 %
pressure	pressure	pressure	in 1000 hPa
precipitation	precipitation	daily precipitation	in 10 mm
sunshine	sunshine	sunshine hours	in 0.1 hours
wind_gust	wind_gust	wind gust	in 1 m/s
wind_speed	wind_speed	wind speed	in 1 m/s

Dataset exploration and cleaning

Cleaning

```
Missing values per column:
DATE                        0
MONTH                       0
BASEL_cloud_cover           0
BASEL_humidity              0
BASEL_pressure              0
                           ..
TOURS_global_radiation      0
TOURS_precipitation         0
TOURS_temp_mean             0
TOURS_temp_min              0
TOURS_temp_max              0
Length: 165, dtype: int64
```

```
Number of duplicate rows: 0
```

There are no missing values or duplicate rows

We don't need any additional dataset filling actions

Dataset exploration and cleaning

Cleaning

- **We decided to take in account only one city: Munich**
  - By dropping every column not related to that city
  - Renaming the features to not include the city name as prefix
- **Converting the ‘DATE’ feature**
  - from YYYYMMDD to three different features: ‘year’, ‘month’ and ‘day’
  - the ‘DATE’ feature dropped
- **Creating new boolean features (features engineering)**
  - ‘rainy_day’ based on ‘precipitation’
  - ‘sunny_day’ based on ‘sunshine’
  - ‘good_day’ based on ‘precipitation’, ‘sunshine’ and ‘temp_mean’

Dataset exploration and cleaning

Cleaned dataset

	day	month	year	cloud_cover	wind_speed	wind_gust	humidity	pressure	global_radiation	precipitation	sunshine	temp_mean	temp_min	temp_max	rainy_day	sunny_day	good_day
0	1	1	2000	8	2.6	9.4	0.91	1.0273	0.20	0.20	0.0	1.7	-0.5	2.6	1	0	0
1	2	1	2000	6	2.1	8.2	0.90	1.0321	0.66	0.00	6.1	1.9	-0.2	5.8	0	1	0
2	3	1	2000	7	2.1	6.9	0.92	1.0317	0.28	0.00	0.4	-0.4	-3.3	0.9	0	0	0
3	4	1	2000	6	2.7	11.7	0.75	1.0260	0.58	0.04	4.5	3.8	-2.8	6.6	1	0	0
4	5	1	2000	5	3.3	13.2	0.87	1.0248	0.26	0.00	0.2	5.3	4.3	7.3	0	0	0

This will be the dataset used for our studies

Aim of the project

Aim of the project

Our study

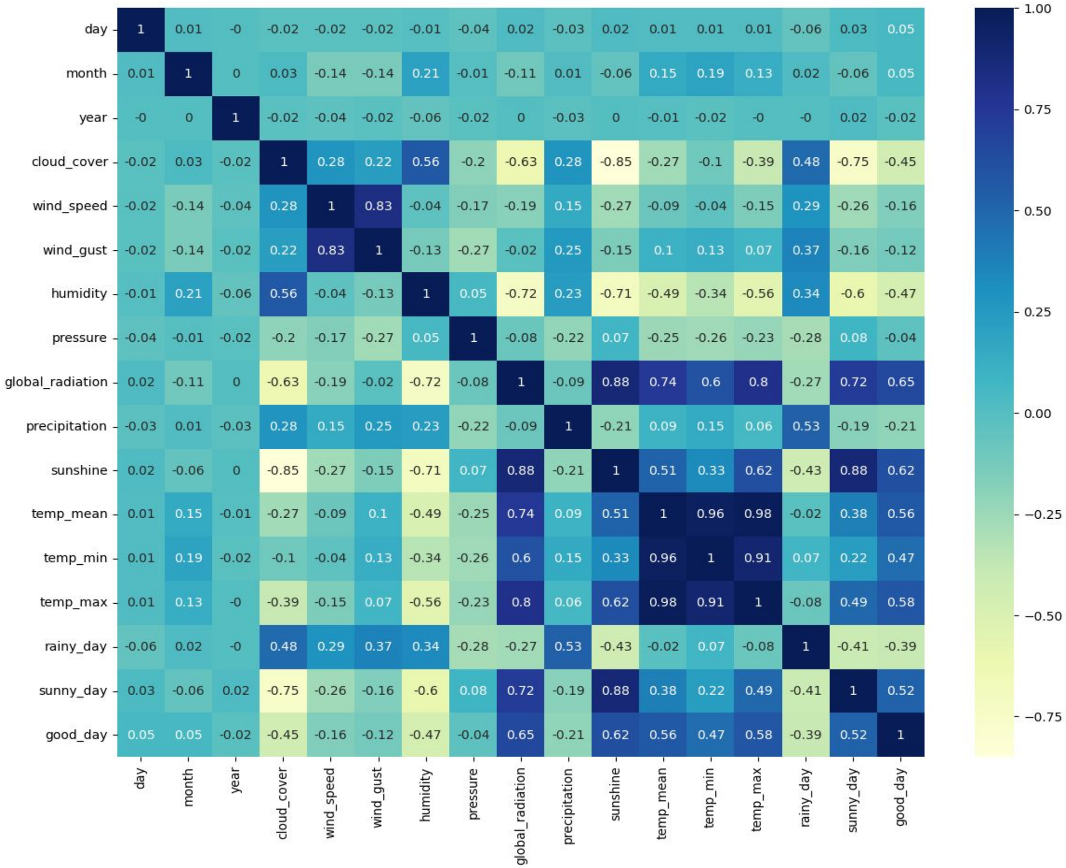
Our study focuses on predicting whether or not it will rain based on the features available to us

Data Visualization

Data Visualization

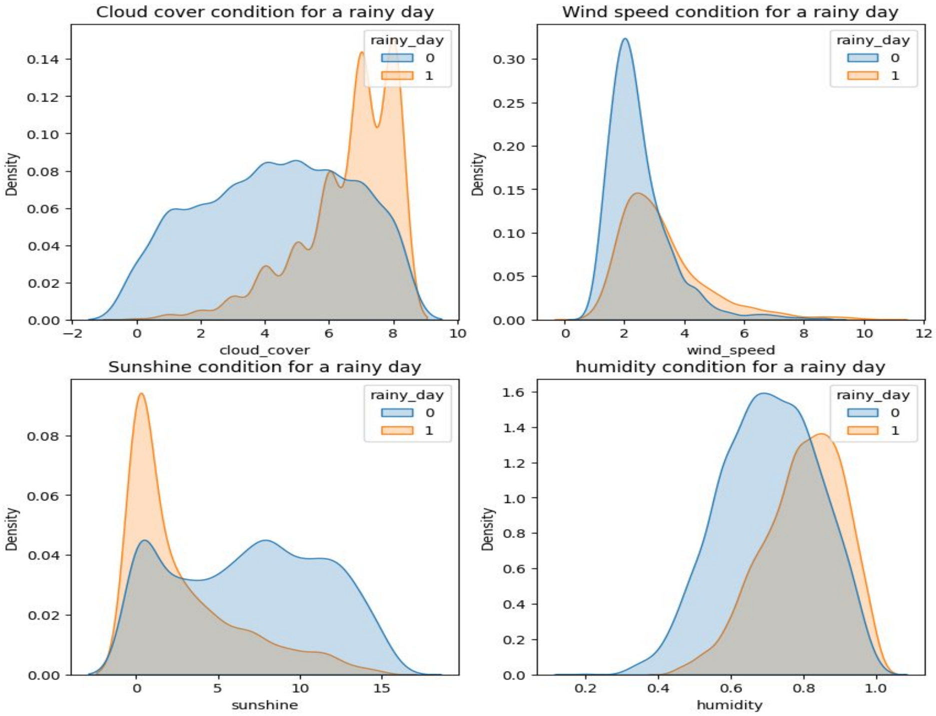
Correlation between the features

- calculated using Pearson coefficient
- Some meaningful correlations:
  - temp_mean/global_radiation
  - cloud_cover/humidity
  - cloud_cover/rainy_day



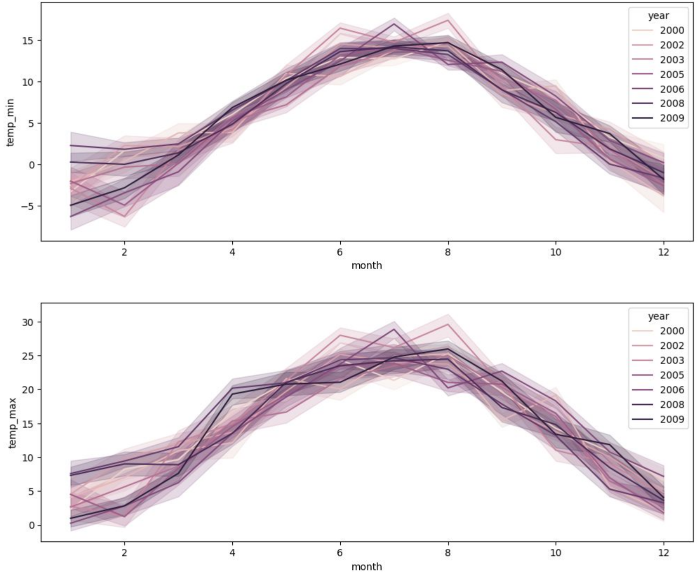
Data Visualization

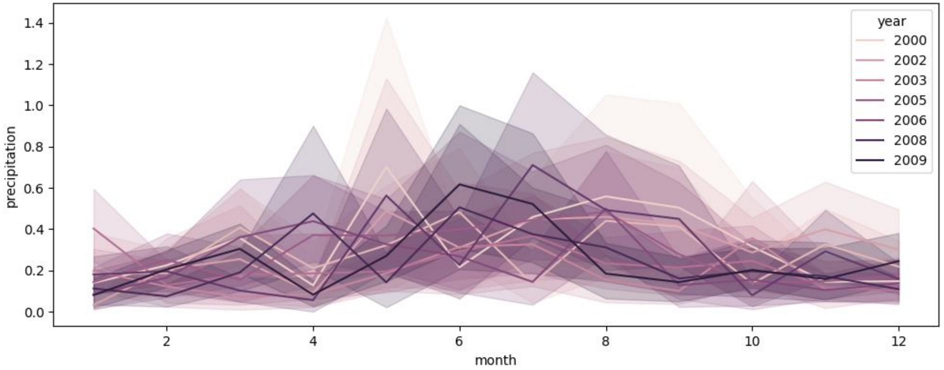
Rainy day based on other features



Data Visualization

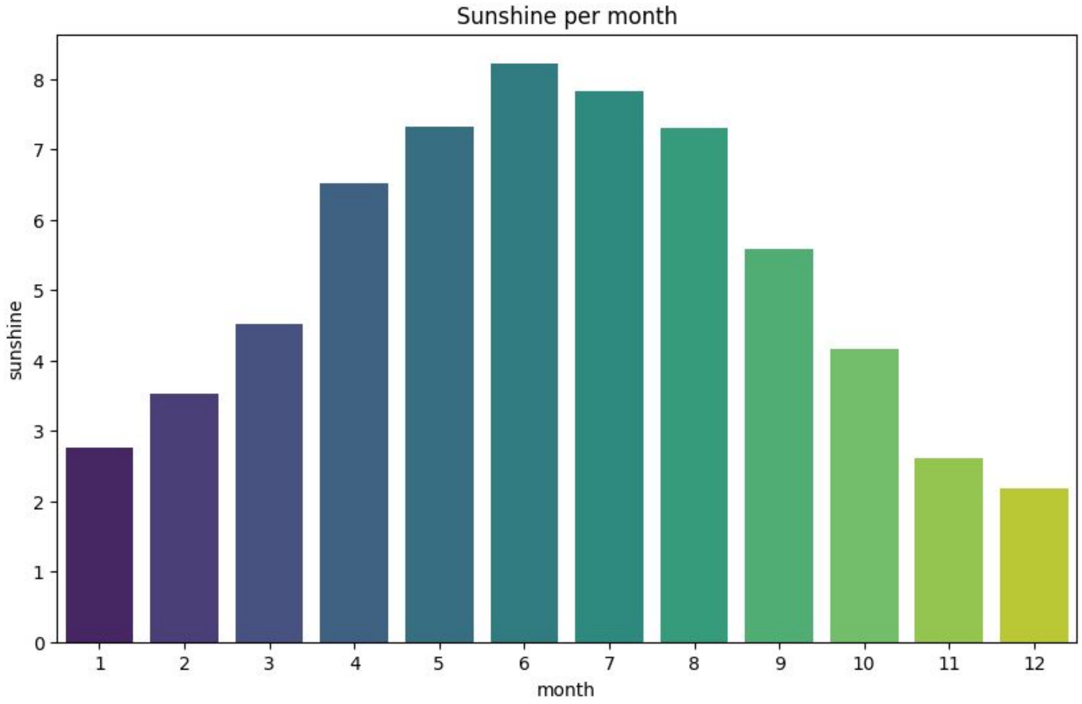
precipitation and temperature curves





Data Visualization

The sunshine during the months



Università di Genova
DIBRIS DIPARTIMENTO
DI INFORMATICA, BIOINGEGNERIA,
ROBOTICA E INGEGNERIA DEI SISTEMI

Machine Learning

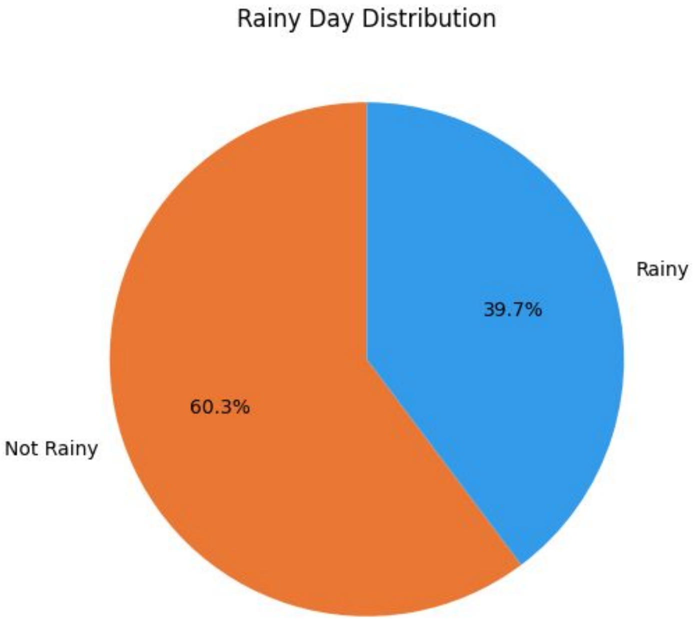
Study on different machine learning methods

Preliminary actions for machine learning

Preliminary actions for machine learning

Data balancing

- the data taken for the training of the dataset are balanced
- not necessary to undersample the data
- Data scaled between the interval [0,1] with MinMaxScaler()



Preliminary actions for machine learning

KFold

- Used for the creation of train and test sets for machine learning
    - x sets with every features except rainy_day and precipitation
    - y sets with only boolean values (0,1) from the rainy_day feature
- Is a cross-validator function that provides train/test indices to split data in train/test sets, Split dataset into k consecutive folds
- Improve hyper-parameters tuning

Preliminary actions for machine learning

The metrics

Accuracy	Measures the proportion of correct predictions to the total number of predictions
Roc-Auc	Used to evaluate the performance of binary classification models. A higher ROC AUC score indicates that the model is better at distinguishing between positive and negative cases.
f1	It is the number of true positive predictions divided by the sum of the true positive predictions and false negative predictions. Recall is an important metric when it is important to minimize the number of false negatives.
recall	The harmonic mean of precision and recall, used when there is a need to balance the trade-off between precision and recall.

The models

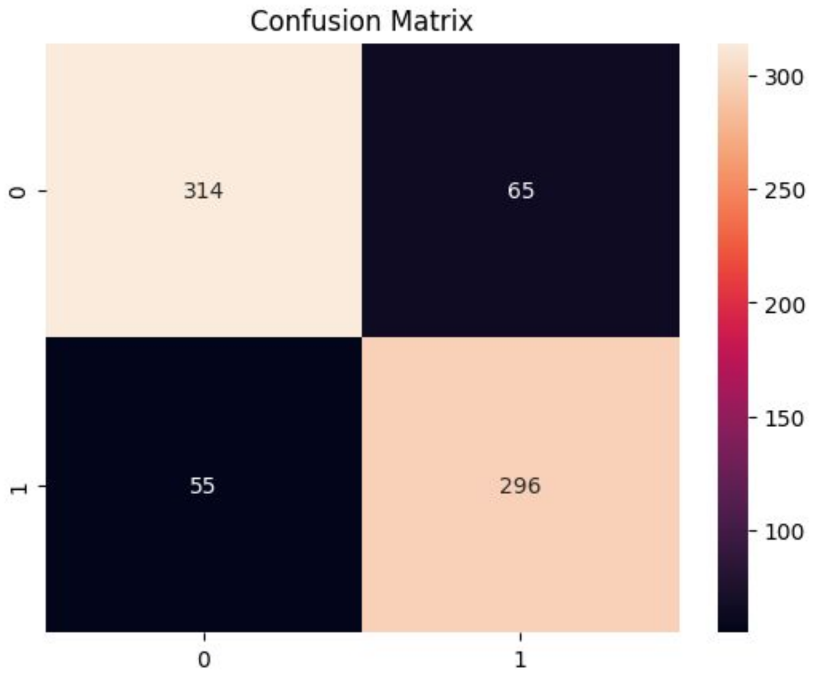
The models

SVC

A support vector machine (SVM) is a supervised machine learning model that uses classification algorithms for two-group classification problems.

```
Results of test set prediction
analyzing:

Accuracy of prediction: 83.56%
roc-auc score: 83.59%
f1 score: 83.15%
recall score: 84.33%
```

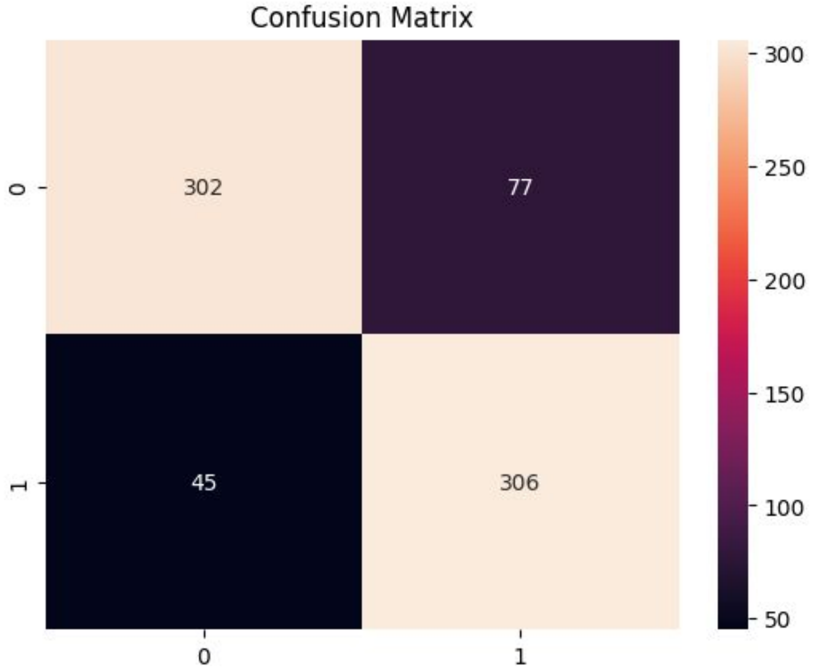


The models

Tuned SVC

```
Results of test set prediction analyzing:

Accuracy of prediction: 83.29%
roc-auc score: 83.33%
f1 score: 82.91%
recall score: 84.33%
```



The models

SVC TUNING

Hyperparameters	Brief description	Values to be selected	Default values	Selected values
C	Regularization parameter	[0.1,0.5,1,5]	1.0	0.5
gamma	Kernel coefficient	['scale','auto']	‘scale’	‘scale’
kernel	the kernel type to be used in the algorithm	['rbf', 'poly', 'sigmoid']	‘rbf’	‘rbf’

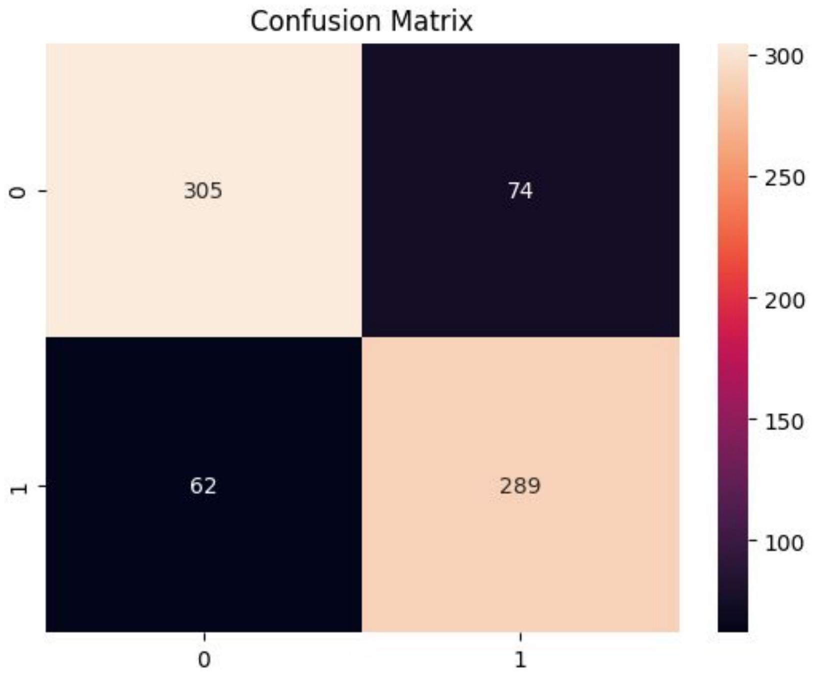
The models

Logistic Regression

Logistic regression is a statistical analysis method to predict a binary outcome based on prior observations of a data set.
It predicts a dependent data variable by analyzing the relationship between one or more existing independent variables.

```
Results of test set prediction
analyzing:

Accuracy of prediction: 81.37%
roc-auc score: 81.41%
f1 score: 80.95%
recall score: 82.34%
```

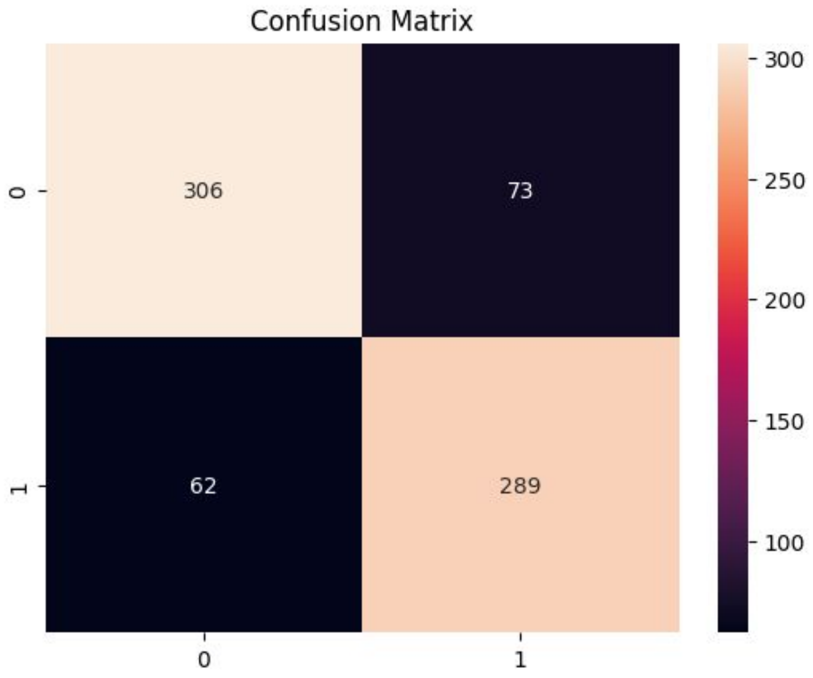


The models

Tuned Logistic Regression

```
Results of test set prediction
analyzing:

Accuracy of prediction: 81.51%
roc-auc score: 81.54%
f1 score: 81.07%
recall score: 82.34%
```



The models

Logistic Regression

Hyperparameters	Brief description	Values to be selected	Default Values	Selected values
C	Regularization intensity	np.logspace(-3, 3, 10)	1.0	0.1
penalty	Penality type	["l1", "l2","elasticnet", None]	‘l2’	l2
solver	Algorithm to use in the optimization problem	["lbfgs", "saga","sag"]	‘lbfgs’	saga

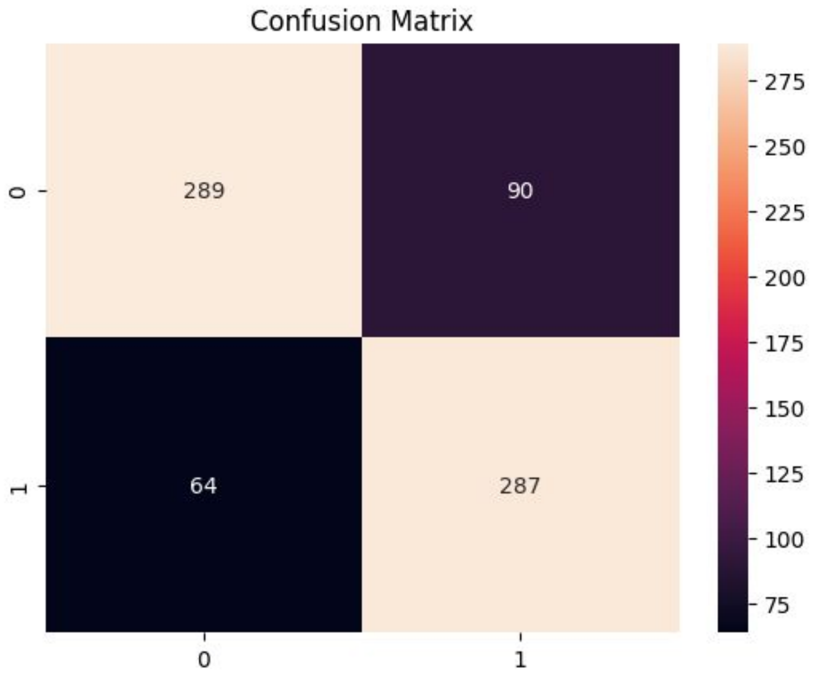
The models

KNN

KNN is an algorithm that can be used to solve both classification and regression problems.
it's used in pattern recognition for the classification of objects based on the characteristics of the objects close to the one considered.

```
Results of test set prediction
analyzing:

Accuracy of prediction: 78.9%
roc-auc score: 79.01%
f1 score: 78.85%
recall score: 81.77%
```

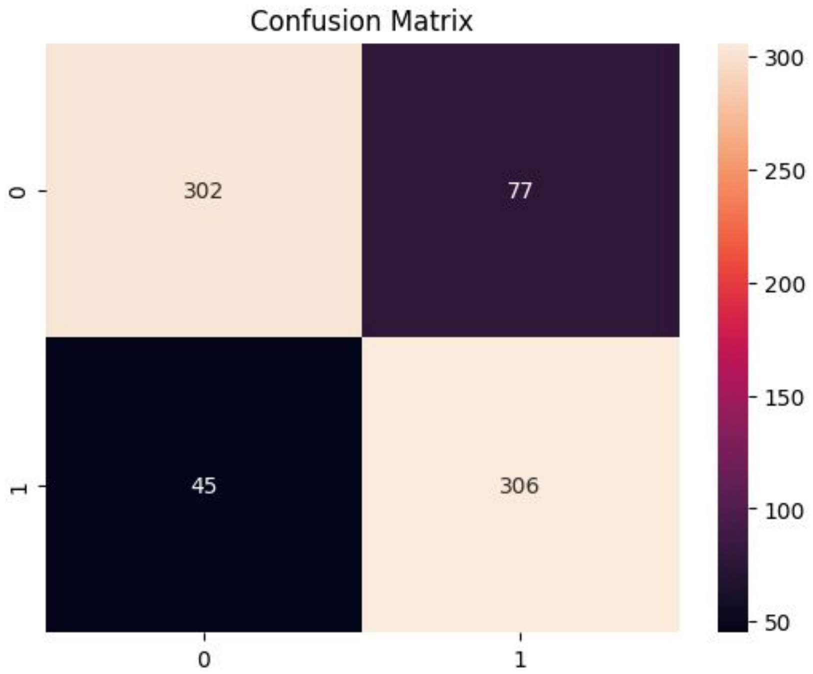


The models

Tuned KNN

```
Results of test set prediction
analyzing:

Accuracy of prediction: 83.29%
roc-auc score: 83.43%
f1 score: 83.38%
recall score: 87.18%
```



The models

KNN TUNING

Hyperparameters	Brief description	Values to be selected	Default Value	Selected values
metric	distance computation metrics	['minkowski','euclidean','chebyshev']	‘minkowski’	euclidean
n_neighbors	Number of neighbors	np.arange(2,25)	5	24
weights	Weight function used in prediction	['uniform', 'distance']	‘uniform’	distance

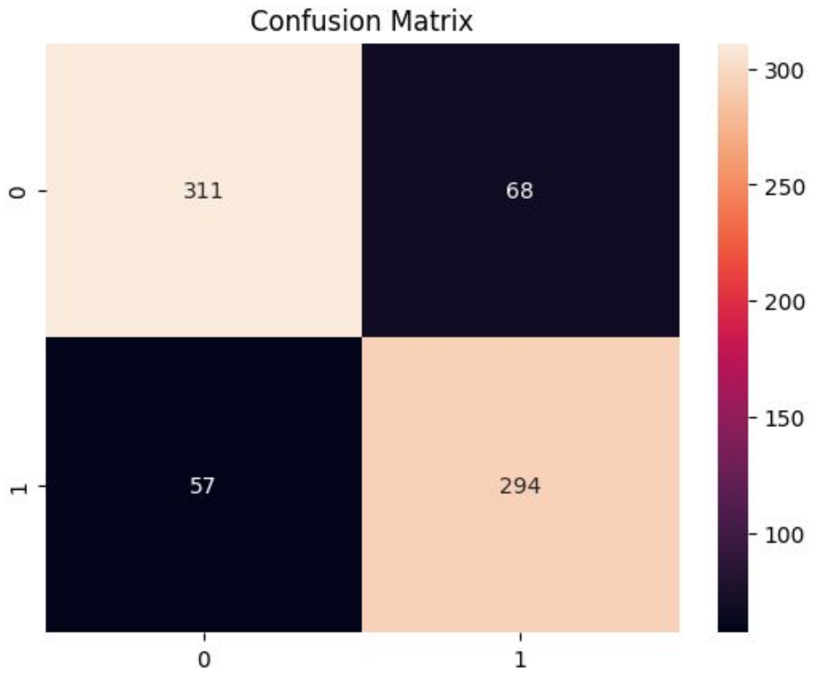
The models

Random Forest

Random forest is a learning method for classification and regression that operates by constructing a multitude of decision trees at training time, a decision tree is a graph of decisions

```
Results of test set prediction
analyzing:

Accuracy of prediction: 82.88%
roc-auc score: 82.91%
f1 score: 82.47%
recall score: 83.76%
```

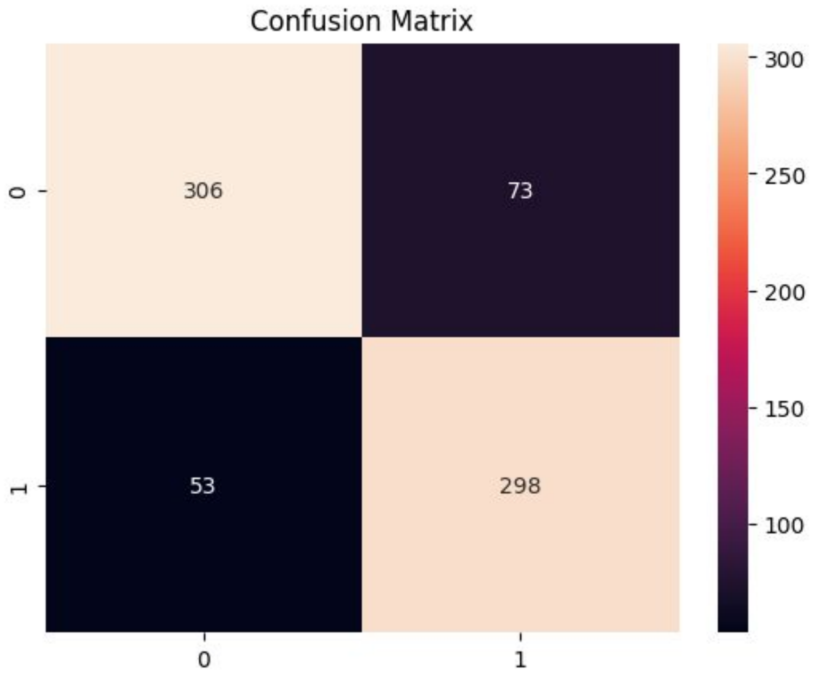


The models

Tuned Random Forest

```
Results of test set prediction
analyzing:

Accuracy of prediction: 82.74%
roc-auc score: 82.82%
f1 score: 82.55%
recall score: 84.9%
```



The models

Random Forest Tuning

Hyperparameters	Brief description	Values to be selected	Default Value	Selected values
max_features	number of features to consider when looking for the best split	['sqrt', 'log2', None]	‘sqrt’	log2
max_depth	The maximum depth of the tree	[10, 30, 60, None]	None	60
min_samples_leaf	The minimum number of samples required to be at a leaf node.	[1, 2, 4]	1	4
min_samples_split	The minimum number of samples required to split an internal node	[2,5]	2	5
n_estimators	number of trees in the forest	[50,100, 200, 300]	100	100
bootstrap	Whether bootstrap samples are used when building trees	[True,False]	True	True

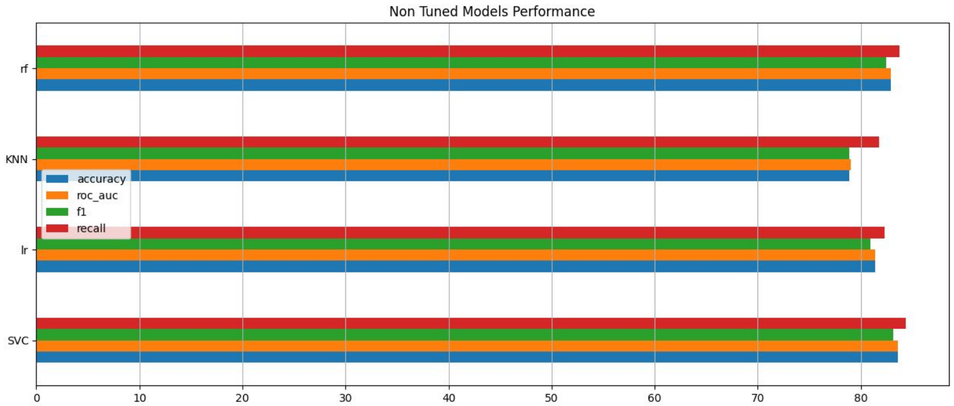
Metrics Comparison

Before and after hyperparameter tuning

Metrics Comparison

Non tuned performance

	accuracy	roc_auc	f1	recall
SVC	83.56	83.59	83.15	84.33
lr	81.37	81.41	80.95	82.34
KNN	78.90	79.01	78.85	81.77
rf	82.88	82.91	82.47	83.76

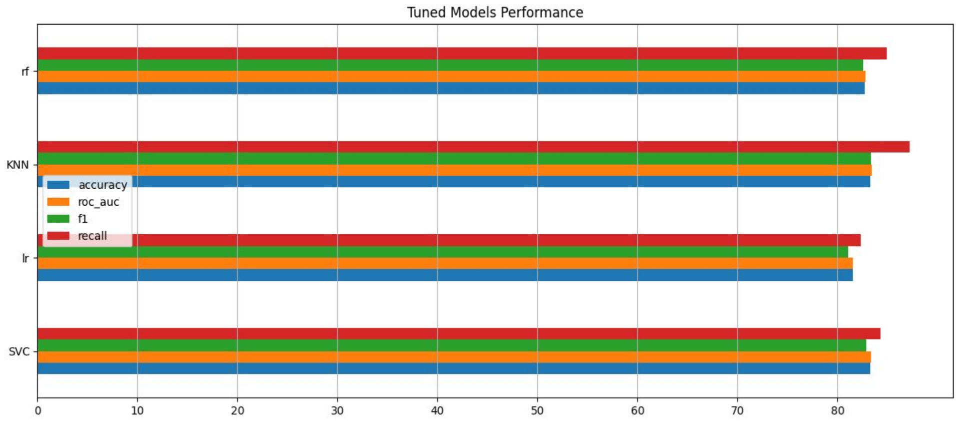


{'accuracy': 'rf', 'roc_auc': 'rf', 'f1': 'rf', 'recall': 'rf'}

Metrics Comparison

Tuned Performance

	accuracy	roc_auc	f1	recall
SVC	83.29	83.33	82.91	84.33
lr	81.51	81.54	81.07	82.34
KNN	83.29	83.43	83.38	87.18
rf	82.74	82.82	82.55	84.90



{'accuracy': 'SVC', 'roc_auc': 'KNN', 'f1': 'KNN', 'recall': 'KNN'}

UniGe

DIBRIS