Machine Learning
Exercise 3 - Kaggle competition 1/2
Data Set: Breast Cancer

Student: se21m024, Thomas Stummer, matriculation number: 1425616

Introduction

This document describes the process to participate at the Kaggle competition for the data set 'breastcancer' and summarizes the results.

Related Files and Links

- Locally developed and tested notebook:***se21m024_Stummer_ml_ex3_breastcancer.ipynb***
- Kaggle submission description:***se21m024_Stummer_ml_ex3_breastcancer_kaggle.pdf***
- Online Kaggle notebook: ***https://www.kaggle.com/code/se21m024/notebook-se21m024-breastcancer***
- Submitted Kaggle prediction: ***breastcancer_prediction_2022-06-05_06-20-31_utc.csv***

Kaggle Submission

The notebook *se21m024_Stummer_ml_ex3_breastcancer.ipynb* was locally developed and tested. The notebook was than imported in Kaggle. The only change in the Kaggle notebook after the import was setting the flag use_kaggle_paths to 'True' to adjust the path for the input data files. Then the notebook was run online and the solution file was submitted directly via the Kaggle web GUI. The solution file *breastcancer_prediction_2022-06-05_06-20-31_utc.csv* was then downloaded and added to the Moodle upload. The document *se21m024_Stummer_ml_ex3_breastcancer_kaggle.pdf* contains exactly the information that was provided in the description field of the Kaggle submission.

Data Set Analysis

The data set contains different health parameters for patients including the information whether a patient had a recurrence event of breast cancer or not. The features appear to be detailed parameters of the health state (regarding the cancer) of the patient. Without further domain knowledge I was not able to interpret the data provided in more detail. The distribution for the individual features is mainly nearly normal distributed or with a shift to one side (e.g. for the convacityMean) or with a tendency to multi modality (e.g. concavePointsMean).

Utilized Software

Operating System:
Windows 10

Software Version:
Python 3.8.5

sklearn Version:
0.23.2

Data Preparation and Algorithms

Processing applied to the data before training / predicting:
The learning data was split into the input features (everything except of the 'ID' and 'stroke' column) and the target feature (the 'stroke' column).

Then the learning data was split into 2/3 training set and 1/3 test set. With this split 4 different algorithms with different parameters were trained:

- k-NN (1-NN)
- k-NN (2-NN)
- k-NN (3-NN)
- Decision Tree (max features: None)
- Decision Tree (max features: sqrt)
- Decision Tree (max features: log2)
- SVM (SVC classifier)
- Random Forests (num trees: 10, max features: sqrt)
- Random Forests (num trees: 10, max features: log2)
- Random Forests (num trees: 100, max features: sqrt)
- Random Forests (num trees: 100, max features: log2)

The best performing algorithm on this test split was than retrained with the whole learning data and used to create the prediction on the test set. The exact results for all algorithms can be seen in the notebook itself.

Results

Local Results

The best performing setting was the SVM algorithm using default settings and the SVC classifier which accomplished an accuracy of 0.968 and an F1 measure of 0.968 on the learning set.

Kaggle Results

The results on the learning set achieved by the local notebook was exactly the same when executing the notebook online in Kaggle (accuracy of 0.968 and an F1 measure of 0.968). The submitted solution for the Kaggle test set achieved a score of 0.9600 (public leaderboard) which is pretty much the same accuracy as for the learning set.