Introduction to Focus Areas in Bioinformatics – WS21/22
Lecturer: Tim Conrad

2 Project 2

- Deadlines: For the **REPORT**: 06.11.2021, 18:00; For the **REVIEWS**: 09.11.2021, 18:00
- All files need to be available through your GIT repository, in the directory "Project 2".
- Remark: everything that is ~~crossed-out~~ is optional and is not mandatory.

2.1 The Data

We will use the Breast Cancer Wisconsin (Diagnostic) Data
(see https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Diagnostic%29)

There are three databases ("data"/"name" pairs). A database contains an ID, a diagnosis and bio-medical attributes extracted from digitized images of fine needle aspirate tissues. In the following, choose one database to perform the tasks.

2.2 Task 1: Take a closer look at the data - Statistics

- Gather statistical information about the data, such as summary statistics, histograms, outliers, or variable relationships (e.g. correlation) about at least three of the attributes.
- You can start with this example code:
  - https://www.kaggle.com/kanncaa1/statistical-learning-tutorial-for-beginners

2.3 Task 2: Build (train) Classifiers

- Develop (or: train) two classifiers to allow classification of the given class (diagnosis).
- Perform tuning of the hyper-parameters (if any).
- You can start with this example code:
  - https://www.kaggle.com/kanncaa1/feature-selection-and-data-visualization
- Hint 1: Check the next task before you choose a classifier.
- Hint 2: Make sure to define and use appropriate train- and test-sets to determine the error on unseen data.

2.4 Task 3: Analyze the Classifier

- Find and use a method that gives you information about the most important features (or attributes) of the individual classifiers.
- You can start with this example code:
  https://towardsdatascience.com/explaining-feature-importance-by-example-of-a-random-forest-d9166011959e

2.5 Task 4: Evaluate the Classifiers

- Perform the evaluation as described in the lecture (accuracy, precision/recall, ROC analysis) and two more methods that you find from the literature. Compare the results for the two classifiers in each category – use a table to display the results. And make sure to discuss the evaluation results in the results section, for example what it means that a classifier is better in precision compared to recall (or the other way round) and what that means for a potential application.
- ~~Use the trained classifier to evaluate it on the other two datasets that you didn't use.~~

~~2.6 Task 5: Use a "Sparse" Classifier~~

- ~~For both classifiers: take only the 3 most important features (basically create a new dataset that only contains these features) and perform training and evaluation of the classifier again.~~

2.7 Deliverables

Your need to upload all source codes and the report to your GIT repository AND the report to the Eduflow system.

If your code is in Python, I must be able to run your code within a Google Colab notebook (https://colab.research.google.com – so you can test it yourself). If your code is not in Python or R, you must provide a manual how to compile and run it on a Linux machine.

- The report should be about **1500-2000** words in length.
- The report must be delivered in PDF format.
- The report must contain a screenshot of the final classifier results.
- The following sections must be present (you can add more if needed):
    - Scientific background
    - Goal
    - Data
    - Results
      For each task, describe what you did (e.g. which steps you took and which methods you used) and what the results are.
    - ~~Discussion 1: Discuss your results from a methods point of view and from an application point of view.~~
    - Discussion 2: Discuss briefly, whether this is a typical project for a data-scientist? (Or why not?)