University of St. Gallen
Institute of Computer Science

Fundamentals of Computer Science (3,125), HS2019
K. Schürholt, M. Schreyer, D. Borth
konstantin.schuerholt@unisg.ch

Assignment 9: Machine Learning Performance Metrics

Deadline: Dec 11, 2019; 12:00 CET

Introduction This exercise will be the first of three on Machine Learning. Before we go into how to *train* or *learn*, we first have to figure out how to evaluate their performance. So this week, we provide you with the data for a classification problem: actual classes $\mathbf{y}$ and predicted classes $\hat{\mathbf{y}}$. You will implement several methods to evaluate the classifier that made the predictions.

Rationale In Machine Learning, there are many different methods and configurations that can be trained on data. To distinguish between the good and the bad, performance metrics are necessary. Further, good and bad depends on the data and the task. This exercise prepares you with the tools necessary for the Machine Learning exercises.

Prerequisites Use your favorite method to open a Jupyter Notebook, e.g. any of these options below:

- local: run **jupyter lab** in your terminal.
- local: open the *.ipynb file in visual studio code.
- cloud: open https://notebooks.azure.com, login with your university account, create and run a new project and upload the *.ipynb file.
- cloud: use any other jupyter notebook environment.

This week, the notebook contains not only code and instruction, but also background information and motivation for the methods we introduce.
As before, for this assignment, you **ONLY** need to edit the notebook!
To run your code, click into one of the cells (the blocks of text that you can edit) and then click the *Run* button on the side of the cell.

(1) Machine Learning Performance Metrics - Basic (4 Points)

In this task, you will compute confusion matrices, compute true/false positive/negatives and compute some basic performance measures.

1.) load the data in `task_1.csv`.

2.) implement the `compute_confusion_matrix` method.

3.) implement the `plot_confusion_matrix` method.

4.) implement the `compute_tp_fp` method.

5.) implement the `compute_performance` method.

(2) Training vs. Test Performance (1 Point)

1.) load the data in `task_2_train.csv` and `task_2_test.csv`.

2.) implement the `compute_train_test_performance` method.

(3) Machine Learning Performance Metrics - Advanced (3 Points)

1.) load the data in `task_3.csv`.

2.) implement the `compute_performance_w_treshold` method.

3.) implement the `compute_ROC_curve` method.

4.) implement the `plot_ROC_curve` method.

(4) Machine Learning Performance Metrics - Understanding (2 Point)

1.) read and understand the content we provide throughout the notebook.

2.) answer the multiple choice questions in the jupyter notebook.

Hand-in Instructions Create a file `README.txt` that includes 1.) any reference that you used to complete this assignment, 2.) pitfalls you encountered and 3.) short explanations of your solution if necessary. Fill in your name and student ID in the comment header of files you edited.

Depending on your environment, export or download your *ipynb **and** *py files, which contains the same code. Include both of these files in your `zip` file.

Compress the whole folder with the Python files and the `README.txt` to a `zip` file named "`assignment8.zip`". Upload the zip file to your exercise group's Course page on Canvas. See Figure.1 for the list of files that should be included in the zip file.

```
assignment9.zip
├── README.txt
├── assignment_9.ipynb
└── assignment_9.py
```

Figure 1: The required files for submission.