University of St. Gallen

Institute of Computer Science

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

Assignment 10: Machine Learning II - Supervised Learning

Deadline: Dec 18, 2019; 12:00 CET

Introduction Last week, you have leaned to evaluate the performance of classifiers. This week, the second of three exercises on Machine Learning takes you to supervised learning. You will implement two common classifiers: *Gaussian Naive Bayes* and *k-Nearest-Neighbors*. We take both methods apart, understand and implement the components and put them back together.

Rationale Classifying still is the most common task for Machine Learning. While there are many different models, the concept of training and prediction remains the same. In this exercise, we go deep into two common and functionally very different methods that still perform the same task.

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. To avoid compatibility issues, change the kernel to python 3.6 (Kernel: change kernel: python 3.6).
- 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 Gaussian Naive Bayes Classification (4 Points)

In the first task, you will compute the components of a Gaussian Naive Bayes Classifier and put it together.

1.) implement the `calculate_prior` method.

2.) implement the `calculate_evidence_mu_sigma` and `calculate_evidence` methods.

3.) implement the `calculate_likelihood_mu_sigma` and `calculate_likelihood` methods.

4.) implement the `train_nb_parameters` and `predict_probability_and_class` method.

② k-Nearest-Neighbors Classification (4 Points)

1.) implement the `preprocess_data` method.

2.) implement the `compute_euclidean_distance` method.

3.) implement the `select_class_label_from_neighbors` method.

4.) implement the `make_knn_prediction` method.

5.) implement the `plot_accuracy_over_k` method.

3 Supervised Learning - 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 "`assignment10.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.

```
assignment10.zip
  |
  ├──── README.txt
  |
  ├──── assignment_10.ipynb
  |
  └──── assignment_10.py
```

Figure 1: The required files for submission.