ingenieur
wissenschaften
htw saar

Hochschule für
Technik und Wirtschaft
des Saarlandes
University of
Applied Sciences

Machine Learning
Summer 2023

Prof. Dr.-Ing. Klaus Berberich
Phone: +49 681 58 67-243
klaus.berberich@htwsaar.de

Programming Assignment 2

The programming assignment will be discussed on **June 19th, 2023**. To obtain bonus points, you have to submit your solution as a Zip-file via Moodle by **June 15th, 2023 23:59**. Teams of up to **three** students are allowed. Please indicate your names and your student numbers in your submission.

2.1 Classifying Fashion Products (2.5)

For this exercise, we'll work on the task of classifying pieces of apparel. Zalando has released a dataset for this task, which consists of 28×28 pixel grayscale images belonging to 10 different classes. The dataset consists of $70,000$ images as training data and $10,000$ images as test data. More information about the dataset can be found at

https://www.kaggle.com/datasets/zalando-research/fashionmnist

You can download the dataset from the URL or from Moodle.

(a) Normalize both the training data and the test data upfront.

(b) Train a **Logistic Regression** classifier on the training data. Determine the **error rate** as well as the **micro**- and **macro-averaged Precision and Recall** on the test data.

(c) Employ a **k-Nearest Neighbor** classifier using $k = 7$. Determine the **error rate** as well as the **micro**- and **macro-averaged Precision and Recall** on the test data.

(d) Train an ensemble of 10 **Decision Trees** using bagging. The depth of the decision trees should be limited to 5. Determine the **error rate** as well as the **micro**- and **macro-averaged Precision and Recall** on the test data.

2.2 Identifying Spam SMS (2.5)

Our second exercise looks at a real-world dataset containing SMS classified as spam or ham. More details about the dataset can be found at:

https://www.kaggle.com/uciml/sms-spam-collection-dataset

You can download the dataset there or from Moodle. Our goal is to predict whether a SMS is spam or ham based on its content.

(a) As you will notice, the dataset is not as clean as the other datasets we've worked with so far. Apply **suitable pre-processing steps** to clean up the data (e.g., to remove encoding errors). In the end, each SMS should consist of its label (i.e., spam or ham) and a sequence of words.

(b) Train a **Naïve Bayes** classifier on the cleaned-up data. Compute **Precision**, **Recall**, and **F1** based on a random split of the dataset into 80% training and 20% test data.

(c) From the **Naïve Bayes** model trained in (c), extract the class priors for the classes `spam` and `ham` and identify for each of the two classes the top-10 words v having the highest value of

$$\log \frac{\mathrm{P}[v \mid \texttt{spam}]}{\mathrm{P}[v \mid \texttt{ham}]} \quad \log \frac{\mathrm{P}[v \mid \texttt{ham}]}{\mathrm{P}[v \mid \texttt{spam}]} \; .$$

These are words with high probability in one, but low probability in the other class. They are thus indicative of a sms belonging to either of the two classes.

(d) How can you use **k-Nearest Neighbors** to accomplish the same task? How do you encode SMS as data points? Which **distance measure** do you use? Compute **Precision**, **Recall**, and **F1** on the training-test split from (b) using $k = 7$.