MLPR – 2023 – Gender identification

The goal of the project is the development of a classifier to identify the gender from high-level features representing face images. These kind of tools find application, for example, as pre-processing for gender-dependent face recognition models.

The dataset consists of image embeddings, i.e. low-dimensional representations of images obtained by mapping face images to a common, low-dimensional manifold (typically few hundred dimensions), for example by means of suitable neural networks. To keep the model tractable, the dataset consists of synthetic data, and embeddings have significantly lower dimension than in real use-cases.

The embeddings are 12-dimensional, continuous-valued vectors, belonging to either the *male* (label 0) or the *female* (label 1) class. The embedding components do not have a physical interpretation.

File `Train.txt` contains the embeddings that can be employed to build the classification model (training set). The evaluation embeddings (evaluation set) are provided in file `Test.txt`. Each row of each file correspond to a sample. The features and corresponding label of each sample are separated by commas, with the first 12 columns of each file corresponding to the sample components, whereas the last column contains the corresponding label.

The datasets are imbalanced, with the training set having significantly more female samples, whereas the test set has significantly more male samples. The samples belong to 3 different age groups. Each age group may be characterized by different distributions for the embeddings, but the age information is not available. The target application considers a balanced use-case, with a working point defined by the triplet $(\pi_T = 0.5, C_{fn} = 1, C_{fp} = 1)$.

Test = 6000, # Train = 2400

Model training and model selection

The report should provide an analysis of the dataset and of suitable models for the task, and the methodology employed to select a candidate solution among different possible alternatives (e.g. different classification models, different values of the hyperparameters, different pre-processing strategies).
The models must be trained over the training set only. When needed, validation data can be extracted from the training set (for example, to compare competing models, to select suitable values for the hyperparameters of each model, or to train score calibration models). Models should be trained to optimize the target application, but performance of the models for alternative applications should also be analyzed and discussed. At the end of this stage, a candidate solution should be provided for the classification task.

Evaluation of the candidate model

The proposed solution must be evaluated on the evaluation set. The evaluation samples should be treated as independent, i.e. the value or the score of an evaluation sample should have no influence on the score of a different sample. The evaluation should report the performance for the target application, but also analyze how the model would perform for alternative use-cases.

Post-evaluation analysis

The choices made during model training should be analyzed to verify that the selected model is indeed competitive for the task. This requires comparing on the evaluation set alternative models that were analyzed and trained, but discarded, in the first phase, to verify whether the proposed solution is indeed optimal or close to optimal with respect to possible alternatives (e.g., the chosen models is effective, the chosen hyperparameters are optimal, etc.).