k-nearest neighbors and classification tasks

Applied ML in Engineering - Exercise 08

TU Berlin, Summer Term 2023

Prof. Dr.-Ing. Merten Stender – merten.stender@tu-berlin.de

Problem 0: accuracy

Students are asked to implement a simple helper function `accuracy(y_gt, y_pred)` that returns the accuracy score for a binary prediction task.

(a) Ground truth labels are given in `y_gt` and predictions are given in `y_pred`, both one-dimensional numpy arrays.

(b) Validate your implementation by testing some edge cases, e.g. perfect predictions, completely inverse predictions, and predictions being only 50% correct.

Problem 1: k-nearest neighbors

Students are asked to implement a k-nearest neighbor classification algorithm from scratch and using an object-oriented programming.

(a) Implement a method `fit(X, y)` that is storing the ground truth data `X` and labels `y` as class attributes

(b) Implement a method `predict(X)` that returns predictions for query data samples `X`

(c) Implement a method `evaluate(X, y)` that returns the accuracy score of the model when making predictions on `X` and comparing them to ground truth labels `y`. This method will call `predict` and `accuracy`.

Problem 2: Fitting a data set

Students are asked to find the best-performing and generalizing model among $\{$k-nearest neighbors, decision tree, logistic regression$\}$ for the binary classification task described in the following. The data is given in the file `dataset_exercise_08.txt` and contains a two-dimensional binary classification problem. The feature dimensions are given in the first two columns, while the labels are given in the last column.

(a) Create a 5-fold cross validation splitting of the data set using `sklearn`'s built-in function

(b) Compute bias and variance for all three models using default hyperparameter settings.

(c) Select the model that you find most promising and perform a hyperparameter study to reduce bias and/or variance of that model.

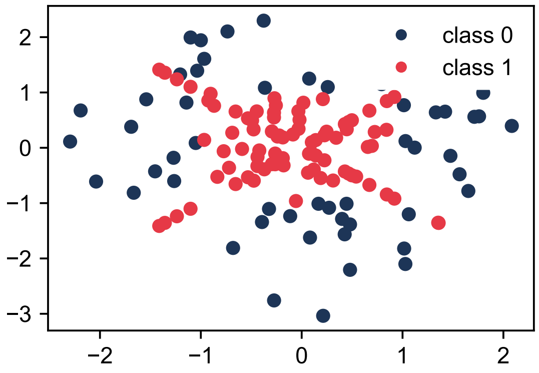

Figure 1: Data set for Problem 2

Problem 3: Further reading

Students are asked to read about the concepts of *confusion matrix*, *precision*, and *recall*.

(a) Understand the different classification scores derived from the confusion matrix

(b) Taking one of the models from the previous tasks, plot a confusion matrix and compute accuracy, precision and recall

(c) (optional): can you change the model parameters such that a model will increase either precision or recall?