Artificial Intelligence and Data Analytics for Engineers

Exercise 6

Introduction

Solving exercises is not mandatory, and exercises will not be graded or corrected by the lecture team. However, we strongly advise you to do as many exercises as possible to prepare for the final exam. Some of the tasks will be discussed with a presentation of the solutions during the exercise session on Thursday. If you want to do exercises at home, we suggest installing Anaconda (https://www.anaconda.com/distribution/ - you will want to download the Python 3.7 version).

In case you have any questions, feel free to send us an e-Mail to: aidae@ima-ifu.rwth-aachen.de.

Task 0) Setting up your Python environment

Since you created a Python environment the first week, we only need to enable the environment from now on. This needs to be done **every time** you open a new terminal, for instance after a system restart or if you accidentally closed the Terminal. Open a terminal and enter:

```
source activate aidae
```

Now you are ready to start programming in Python. Remember to ensure that spyder is installed in your environment (`conda install spyder`) and launched from the terminal where you activated the environment.

Task 1) Regression

In this task you will explore the application of regression for the estimation of concrete compressive strength. The concrete compressive strength is a highly nonlinear function of age and ingredients. These ingredients include cement, blast furnace slag, fly ash, water, superplasticizer, coarse aggregate, and fine aggregate.

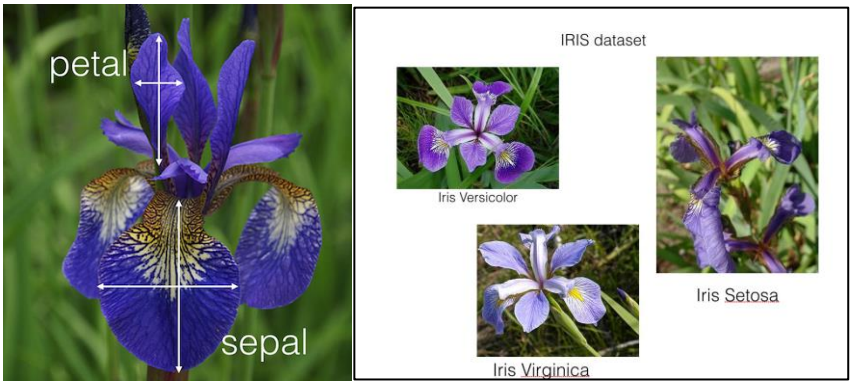
For the task you are given a dataset which can be downloaded from "https://archive.ics.uci.edu/ml/machine-learning-databases/concrete/compressive/Concrete_Data.xls". The dataset contains several (input) variables and one (output) variable, which are described as follows:

- Cement (component 1) -- quantitative -- kg in a m3 mixture -- Input Variable
- Blast Furnace Slag (component 2) -- quantitative -- kg in a m3 mixture -- Input Variable
- Fly Ash (component 3) -- quantitative -- kg in a m3 mixture -- Input Variable
- Water (component 4) -- quantitative -- kg in a m3 mixture -- Input Variable
- Superplasticizer (component 5) -- quantitative -- kg in a m3 mixture -- Input Variable
- Coarse Aggregate (component 6) -- quantitative -- kg in a m3 mixture -- Input Variable
- Fine Aggregate (component 7) -- quantitative -- kg in a m3 mixture -- Input Variable
- Age -- quantitative -- Day (1~365) -- Input Variable
- Concrete compressive strength -- quantitative -- MPa -- Output Variable

The estimation of the concrete compressive strength (output variable) given the input variables (e.g. Age) is the regression problem.

a) Import the data using pandas and the `read_excel` function.
b) Explore the data graphically using the `matplotlib` library, e.g. visualize the distribution of the different variables. As a bonus try to generate a scatter plot using the `plotting.scatter_matrix` function of pandas to show possible correlations within the dataset.
c) Split the available dataset into a train and test dataset using the `train_test_split` method from `sklearn.model_selection`.
d) Train the regression model using the `sklearn.linear_model.LinearRegression` class.
e) Test your model by making a prediction using the `sklearn.linear_model.LinearRegression.predict` method.
f) Evaluate your model using the two metrics "mean squared error" and "r2" from `sklearn.metrics`.
g) Bonus: Implement a 5-fold cross validation to train/test your model.
h) Bonus: Try different regression algorithms.

Task 2) Classification



In this task, you will explore the application of classification for one of the most famous dataset in machine learning: the "Iris database". The dataset contains 3 classes of 50 instances each, where each class refers to a type of iris plant. One class is linearly separable from the other 2, the latter are NOT linearly separable from each other. The dataset contains several (input) variables and one (output) variable (the Iris class), which are described as follows:

- sepal length in cm
- sepal width in cm
- petal length in cm
- petal width in cm
- class: Iris-Setosa, Iris-Versicolour, Iris-Virginica

The estimation of the correct Iris class given the input variables (e.g. sepal width) is the classification problem.

a) Import the dataset using the `sklearn.datasets.load_iris` method.
b) Store the dataset in a pandas using a pandas `DataFrame`.
c) Explore the data graphically using the `matplotlib` library, e.g. visualize the distribution of the different variables. As a bonus try to generate a scatter plot using the `plotting.scatter_matrix` function of pandas to show possible correlations within the dataset.
d) Split the available dataset into a train and test dataset using the `train_test_split` method from `sklearn.model_selection`.

e) Train the classification model using a Decision Tree Classifier. For that use the `sklearn.tree.DecisionTreeClassifier` class.
f) Test your model by making a classification using the `sklearn.tree.DecisionTreeClassifier.predict` method.
g) Evaluate your model by printing both the confusion matrix and the classification report from `sklearn.metrics`.
h) Bonus: Implement a 5-fold cross validation to train/test your model.
i) Bonus: Try different regression algorithms.

Note: If you want to practice your skills with supervised learning you can implement or follow the same approaches as above for various datasets available at sites like Kaggle or UCI (https://archive.ics.uci.edu/ml/machine-learning-databases)