Data Science | Lab 1: Exploratory Data Analysis

Learning Goals

- visualizing data (unlabeled and labeled)
- interpreting diagrams and graphs

Plotting in Python

There are a lot of different plotting libraries available in Python. Not every diagram type is available in each library. The basic library is `matplotlib`, which takes care of correctly rendering the graphs. Most other libraries are built on top of matplotlib and add high-level functionality for changing the plot's appearance or for advanced or combined diagram types.

- 🌐 matplotlib: basic rendering utilities
- 🌐 pandas: straightforward plotting of `DataFrame` objects
- 🌐 seaborn: pretty graphs
- 🌐 plotly: interactive graphs

Tasks

Reading the Dataset

```
from sklearn.datasets import load_wine
X, y = load_wine(as_frame=True, return_X_y=True)  # split into features X and labels y
df = X.copy()  # deep copy the features
df['target'] = 0  # make an unlabeled dataset (labels are used later)
# df contains features+label, X contains features only, y contains labels only
```

For a first impression, explore the data with the built-in functions `head` and `describe`.

Exploration of Unlabeled Data

1. Display all features in a single plot by using a parallel lines plot or a heatmap.
2. Decide whether the data needs scaling.
3. Detect any outliers in the features' distribution through means of a boxplot.
4. Find out which data points have outliers in any feature.
5. Detect correlations between features by plotting a scatter matrix or computing the correlation values.

```
import matplotlib.pyplot as plt
from pandas.plotting import parallel_coordinates, scatter_matrix
from sklearn.preprocessing import StandardScaler
```

Exploration of Labeled Data

Redo the above steps for the labeled dataset by producing grouped plots according to the labeling information.

```
df['target'] = y
```

Homework

Plot upload

Redo the above tasks for the Iris Dataset. Information on the dataset can be found in the 📝 sklearn documentation. Save your **favorite (iris-related) plot and upload** it to Moodle. Make sure that the plot is self-contained:

- Concise title
- (Readable) axes labels and titles
- Experiment with appropriate color schemes
- Add a legend if necessary

In Moodle, make sure to also **submit a short description** of your interpretation of what can be seen in the plot.

This is an **individual** assignment, meaning that you are graded individually. Collaboration in discussing the approach or some technical details is highly welcome, the assignments need to be prepared and uploaded individually. Make sure that essential original content(in structure, reasoning and interpretation) makes individual grading possible on the instructor's side.

🕘 Deadline: 18.10.2022 (5:00pm)

Quiz

Take the quiz in Moodle. Make sure to have your Python notebook open and have your code up and running for both datasets.

🕘 Deadline: 18.10.2022 (5:00pm)

Further Reading

✨ On the importance of data visualization: 🌐 Anscombe's Quartet (Wikipedia), German version also available