General hints for code submissions To make it easier for us and others to understand your solutions please follow these guidelines:

- If available use the template file to create your solution.
- Please add comments so we can understand your solution.
- Please make sure to load all required packages at the beginning of your code.
- Only use relative file paths for `source()`, `load()`, etc.
- Each exercise directory contains a `skeleton` folder where preliminary `R` files are located.
- Use these `R` files as a basis for creating your solution which should be contained in a main `R` file named as the content of the exercise, e.g., `evaluation.R`.
- You can (and sometimes have to) reuse code from previous exercises.
- The points indicate the difficulty of the task.
- If not stated otherwise, we will use exclusively R 4.0 or greater.

Having learned about different ways to empirically evaluate the performances of algorithms and AutoML systems, in this exercise you will now implement some of these techniques.

1. **McNemar Test** [3 points]

   Two models are trained to classify images of cats and dogs. The result is stored in *MCTestData.csv* with $n = 500$ images. The function *load_data_MNTest()* loads the data as an $n \times 3$ *data.table*, where the first column represents the ground truth. The 2nd and the 3rd columns represent the output from model 1 and 2 respectively.

   Implement a *McNemar Test* to determine whether the two models perform equally well on the dataset. In your solution state what is H_0, H_1 and return χ^2 for this evaluation.

2. **Two-Matched Samples t-Test** [3 points]

   TMStTestData.csv contains *error* values of two algorithms on $n = 419$ datasets, the function *load_data_TMStTest()* loads the data as an $n \times 2$ *data.table*.

   Implement a *Two-Matched-Samples t-Test* to determine whether the two algorithms perform equally well on the dataset and return the test statistic t value for this evaluation.

3. **Friedman Test** [3 points]

   FTestData.csv contains *error* values of $k = 5$ algorithms on $n = 15$ datasets, the function *load_data_FTest()* loads the data as an $n \times k$ *matrix* Err, where Err_{ij} represents the error of the jth algorithm on the ith dataset.

   Implement a *Friedman Test* to determine if all algorithms are equivalent in their performance and return χ^2_F for this evaluation. If this hypothesis is not rejected, you can skip the next question.

4. **Post-hoc Nemenyi Test** [3 points]

   Having found that all the algorithms are not ranked equally, now we need to utilize the *Post-hoc Nemenyi Test* to find the best-performing algorithm.

   Compute the test statistic for all the algorithms pairs $\{j_1, j_2\}$. The results should be stored in a upper triangular matrix $\mathbf{Q}$, where $Q_{m,n}$ is the q value between the algorithms j_m and j_n. Compute the critical values of the test distribution, derive p-values and test decisions.

5. **Boxplots** [2 points]

   Create a boxplot for error value of the algorithms which have the best and the worst average ranks stored in *FTestData.csv.* We expect all plots to have axes labels.