ADS1 exercises – Clustering

In this exercise we will look at weights and size measurements of fish for different species stored in `fish_measurements.csv`. Please consult `https://myfwc.com/fishing/saltwater/recreational/measurement/` for the meaning of the length measurements.

1. (a) Save the `cluster_tools` module in your working directory. Import it and inspect it using the `help()` function.

   (b) Inspect the file using Excel or a text editor. Read the csv file. This file has two header lines. Use the `skiprows` keyword argument to deal with this. The keyword argument `skiprows=(1,2)` will skip the second line[1].

   (c) Find a good combination of two columns for clustering. Columns which are highly correlated (or anti-correlated) are not good for clustering. Create a correlation heatmap (`map_corr()` function available in the attached module `map.py`) or the scatter matrix to identify promising combinations.

   After finding a good combination of two attributes extract the two columns you intend to use for clustering[2]. For a dataframe `df_old` with the columns "A", "B", "C", "D" this can be done as follows.

   ```
   df_new = df_old[["A", "C"]].copy()
   ```

   The copy method makes a new copy. Changes in one dataframe do not affect the other or cause other complications. The new dataframe can then be used as argument for the fitters.

   (d) As explained in the lecture distance measurements for clustering do not work well if the ranges are very different. Use the `scaler` function from the to normalise the values and save minimum and maximum[3].

   (e) Perform clustering using `KMean` clustering. Inspect the results by producing colour coded plots as shown in the lecture. Use the silhouette score to arrive at the correct number of clusters.

   (f) Check whether the cluster labels coincide with the name of the species. One quick way to do: write the labels into a new column of the dataframe, sort by label and write the result into a csv or excel file for inspection.

   (g) Use the backscale function to convert the cluster centres to the original scale. Plot them and the original data.

[1] You are free to preprocess your files used for the assignment using a text editor or excel

[2] The clustering functions can work with dataframes containing more than two columns, but handle with care. It can improve the clustering but also lead to spurious results.

[3] `sklearn.preprocessing`. This module contains several functions for scaling the data and more. The `MinMaxScaler` reproduces our scaling method. However, it does not support back scaling. Link `https://scikit-learn.org/stable/modules/preprocessing.html#preprocessing`