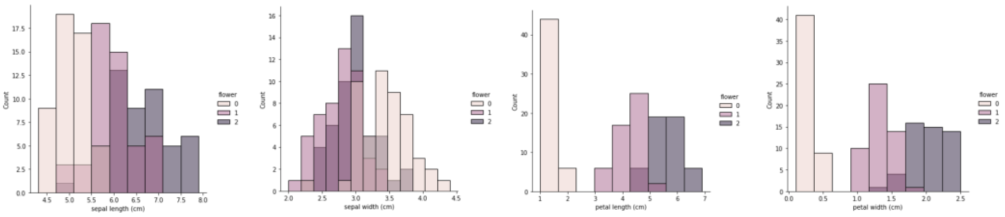
Tasks for passing grade

Analysis of the basic iris data set

- I used the built-in method ‘shape’ to get the output (150, 4), which means 150 rows (data points) and 4 columns (attributes). Based on this, and the knowledge that the indexes of data set go between 0 and 149, it shows that there’s 150 data points in the set.
- I grouped the flowers into three subgroups, to establish the data distribution between the data sets’ four attributes. My conclusion was the following:
  - The first flower group, with the id 0, has the shortest but the widest sepals and their petals are both the shortest and the least wide.
  - The second group (id 1) are in the middle of all attributes, except sepal width, where they have among the lowest values.
  - The last group (id 2) are ranked in the top for all attributes, except sepal width, where they are distributed in the middle.



K-Means Algorithm

- The k-means algorithm takes the data values and clusters them in *k* clusters. It takes *k* random data points and uses them as centres of the clusters. Then it measures the rest of the data points’ distances to these centre points and assign them to its nearest cluster (based on the closest centre point).
  When all data points are assigned to a cluster, it calculates the mean of the cluster. Then it starts over and repeats this process for a given number of maximal iterations. The algorithm can stop prematurely if the
  When the iterations are done, it analyses the results and decides the centre points where the clusters are most even.

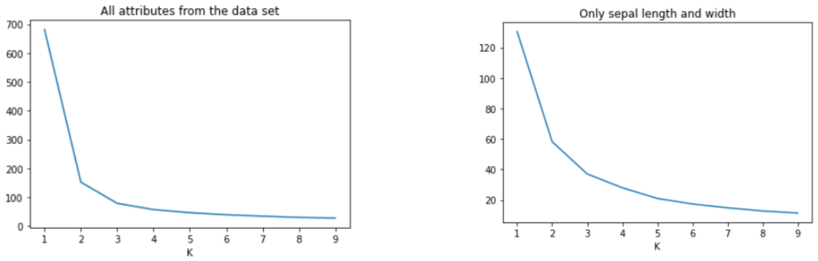
Clustering with K-Means

- I used the elbow method to get to the most optimized number of clusters to use. In the method you iterate a given number of times and test the *k*-value for each iteration. I looped ten times in my study. Then you plot the result and get this ‘elbow’. Where the curve flattens out, it indicates the ‘best’ number of clusters to use.
  First I ran the elbow method with the whole data frame and it’s clear that three is the most logical *k*-value. Then when I ran the test with just two attributes (*sepal length* and *sepal width*, I will later tell why I came to the conclusion to use these attributes), it showed that three is still a good value of *k*, but there’s a small flattening of the curve at five as well.
  So I tested the k-means with both three and five clusters, to see the difference, and my conclusion is that there isn’t much difference in regards to clustering spread. But five clusters look better in my opinion, because you can more easily see what data point belongs to which cluster and centre point.

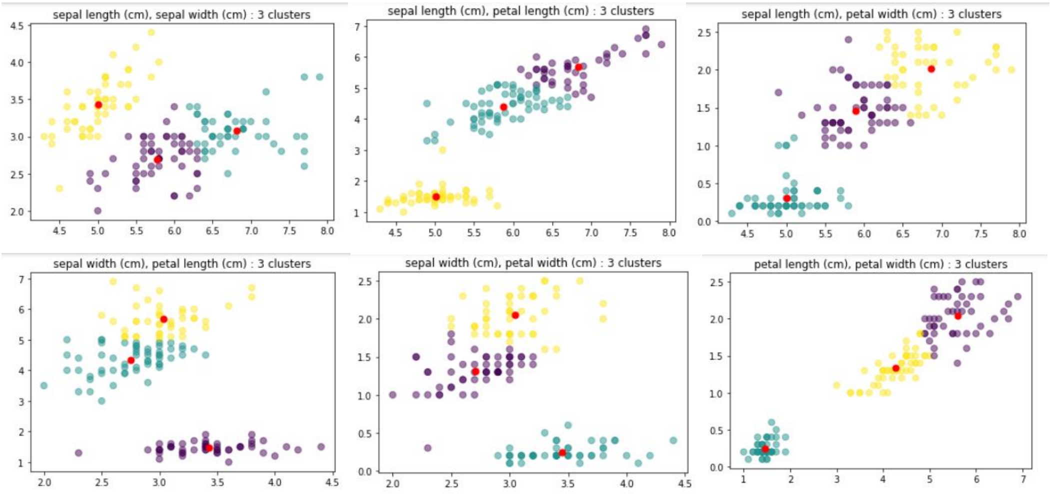
The Elbow Method

Left: The whole data frame

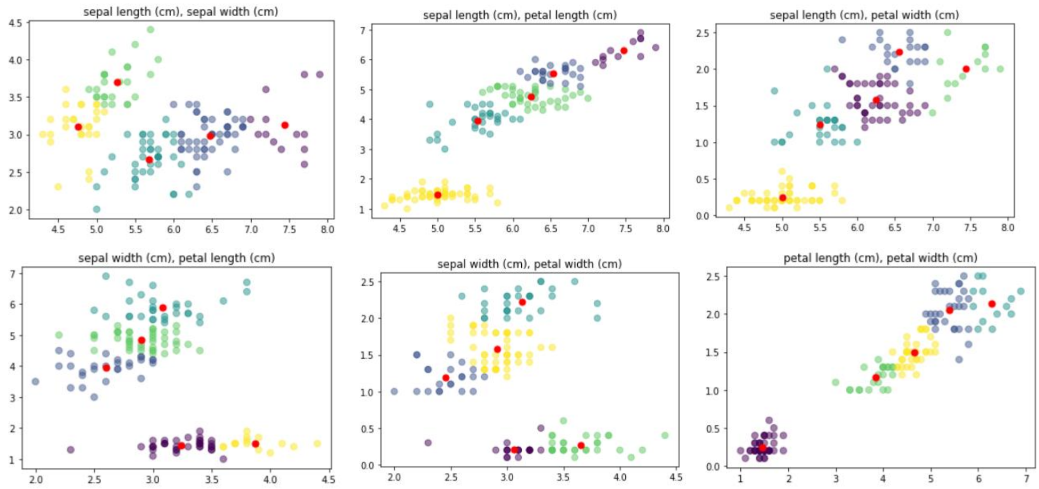
Right: Two attributes



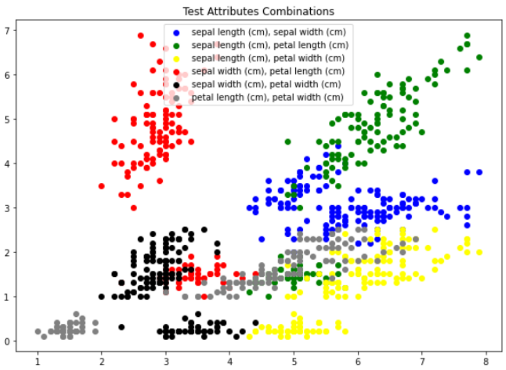
K-Means with three clusters



K-Means with five clusters



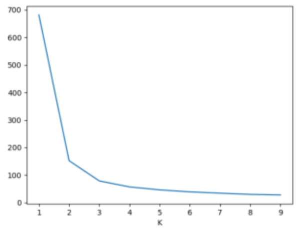
- I decided to use *sepal length* and *sepal width*, when using only two attributes. I tried all possible combinations (as you can see above) and then summarized my result to see which attributes that are the most centred with each other.

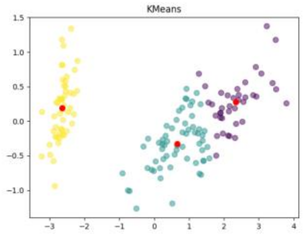


More advanced tasks

Clustering with all attributes from the iris data set

- When using all attributes, the most optimized number of clusters are three, at least based on my study. I used, like in the previous task, the elbow method to measure the *k*-value for the whole data set. But this time I made a Principal Component Analysis transformer to transform the data values from four attributes to two attributes.
  So my conclusion is to use three clusters for the whole data frame and five clusters when you use only two attributes.



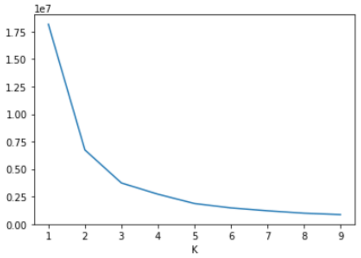


Analysis of the special iris data set (from csv)

- Once again, I used the ‘shape’ method to get the size of the data set. It shows that the data set contains 1,500,000 data points, with eight attributes. The attributes are these:
  - *sepal length* and *sepal width*
  - *petal length* and *petal width*
  - *extra sepal length* and *extra sepal width*
  - *extra petal length* and *extra petal width*

Clustering the special iris data set vs. the basic iris data set

- I used the ‘elbow method’ to get the *k*-value and it showed, like the previous data set, that three (or possibly five) clusters are most logical to use. I decided to time the difference to run the k-means algorithm for both three and five clusters.
  What I learned is that the elbow method takes almost 2 minutes to calculate, but my elbow function may not be the best time optimized function, so I have taken that in consideration.
  The difference in timing the k-means calculations are also large. The basic data set takes 0.02-0.03 seconds to calculate (and it doesn't matter if it's 3 or 5 clusters) and the special data set takes 5.5-6.5 seconds for 3 clusters, and approximately 14-16 seconds with 5 clusters.



- The time difference between the two data set is based on number of data points the algorithm must iterate through.
  The basic data set contains 150 data points with 4 attributes and the special data set contains 1,500,00 data points with eight attributes. So the algorithm has to iterate a great number of more times each iteration, than with the basic data set.