DAT405 Assignment 3 – Group 11

Hannes Skoog - (8 hrs)
Gabriel Wendel - (8 hrs)

April 19, 2023

Problem 1

a.

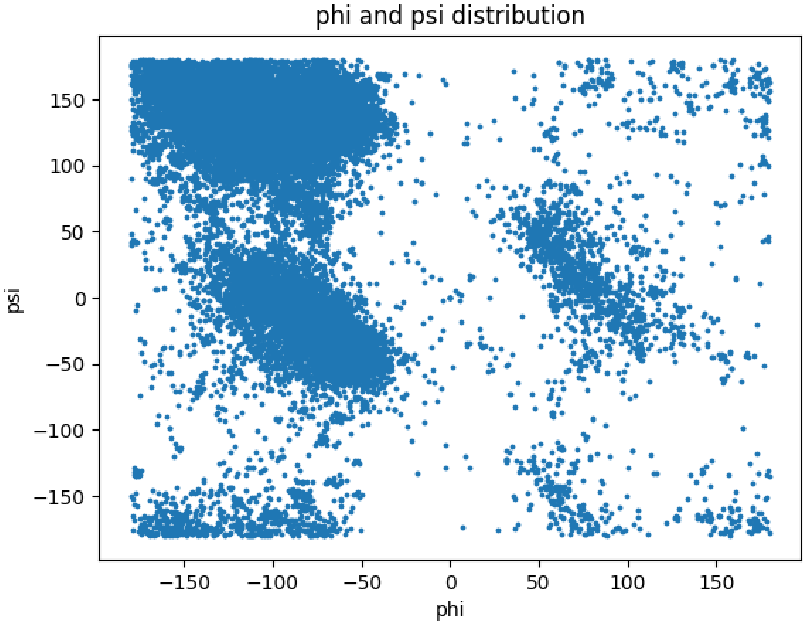


Figure 1: Scatter plot of phi and psi

b.

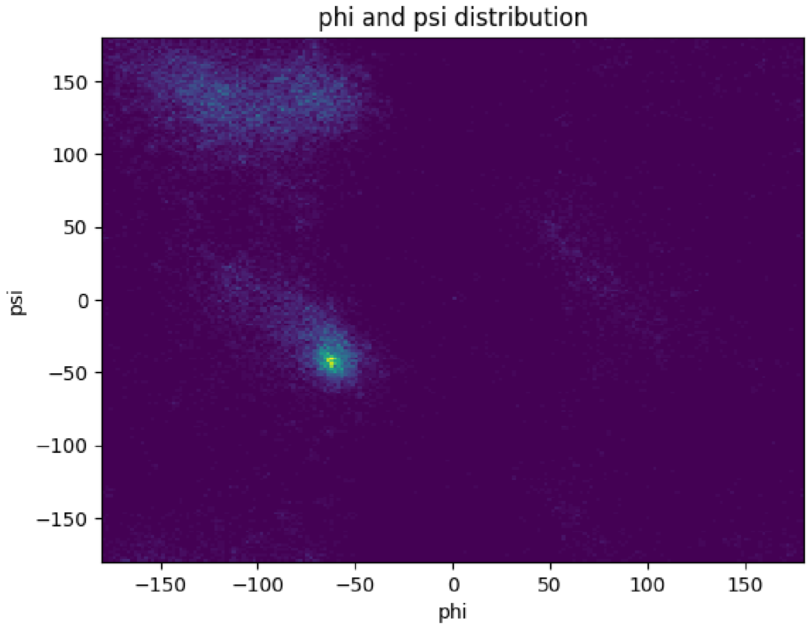


Figure 2: 2D histogram of data

c.

To transform the data to account for the data wrapping around, we used the modulo operator. If we use the formula,

$$x \,\%\, 360 \cdot \frac{\pi}{180^\circ} \tag{1}$$

where x is the degree, we will get a radian that is fitted between $[0, 2\pi]$. If we do this with all data points, the resulting scatter plot of the transformed data is the result we see in fig.3.

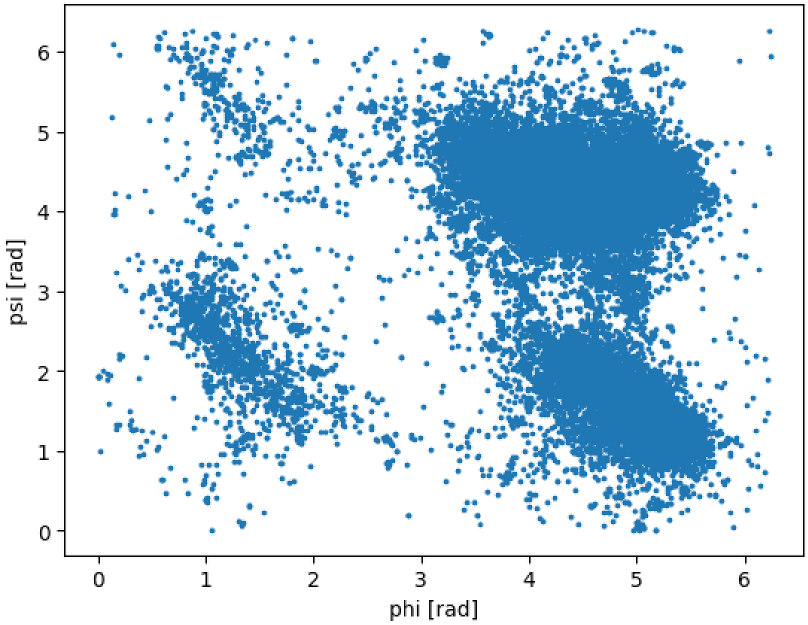


Figure 3: Transformed data in a scatter plot

In this case we also had to shift psi with +110° in order to not have any clusters wrap around, but this is an arbirtary choice.

Problem 2

a.

To evaluate what value of k would give a good clustering, we used the elbow method. This method requires you to record the sum of squared errors for different values of k which then can be plotted with the sum of squared errors on the y-axis and the different K:s on the x-axis.

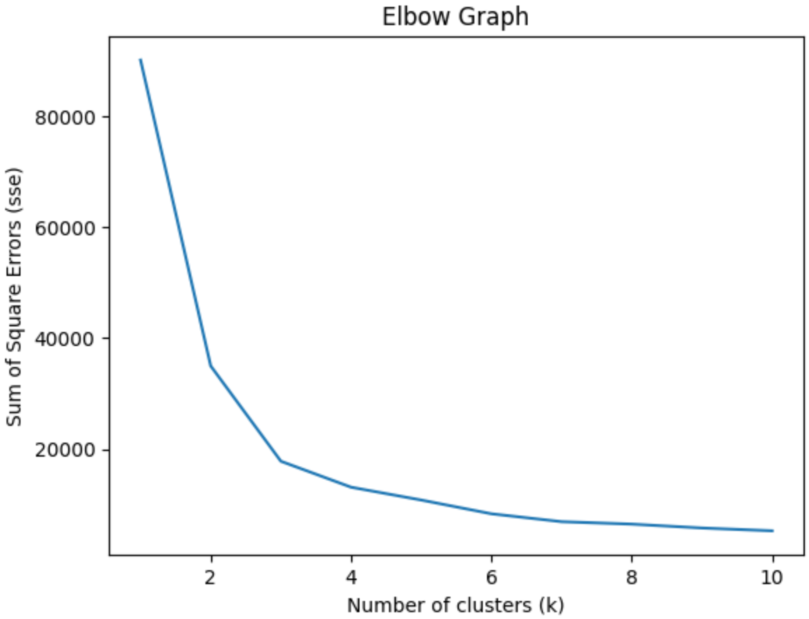


Figure 4: Elbow graph showing the effect of different k

From this plot, we can derive that the change in the sum of squared errors is big at low values of k. We can see how the change in sum of squared errors mellows out from $k > 3$. This indicates that $k = 3$ would make a good choice for the number of clusters in our K-means clustering. One could also argue $k = 4$ being a reasonable choice here. Using higher values of k would result in lower sum of squared errors, but generally, we want to keep the number of clusters as low as possible to avoid creating unnecessarily many clusters. Therefore, $k = 3$ is our best choice to avoid having too many clusters while still retaining a relatively low sum of squared errors.

The clusters created with $k = 3$ can be seen in fig.5.

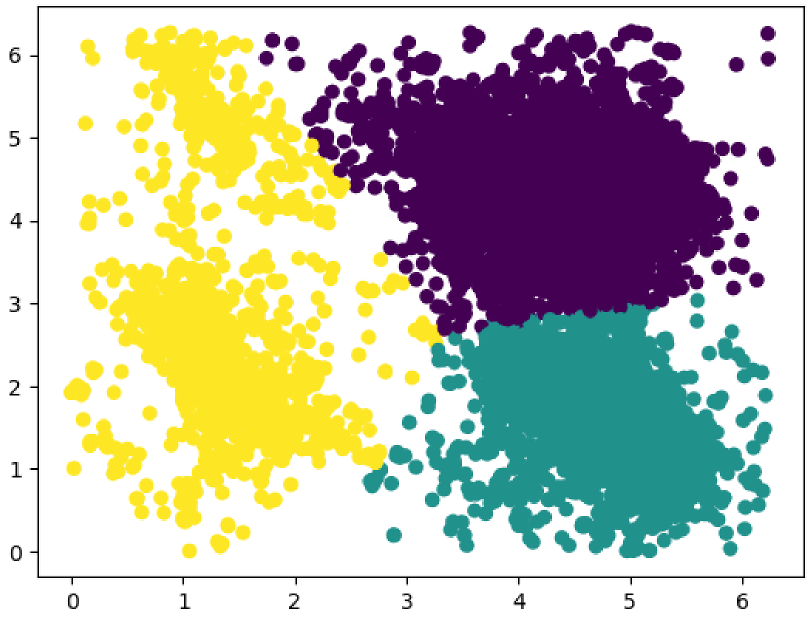


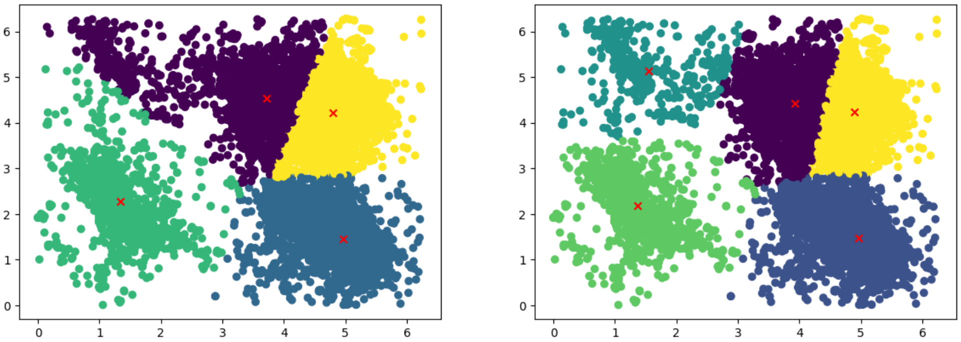
Figure 5: K-means clustered data with k=3

b.

These clusters are somewhat reasonable, although not optimal, since the data isn't really optimized for using K-means. We don't have any visibly clear circular clusters of similar sizes, which can create issues for K-means. The data is also fairly spread out with some outliers which affect the centroids, which in turn can skew the clusters.

One could argue that the yellow cluster should be split into a top and bottom cluster by just looking at the scatter plot. However, since we are using K-means, the irregular shape and size of the top part of the yellow cluster makes it hard to cluster this properly. Attempting to correct this with other values of k gave us the results displayed in fig6. Using $k = 4$ didn't really give the results wanted. The yellow cluster from $k = 3$ is split, but so is the old purple cluster. With $k = 5$ we get a more resonable clustering with the old yellow cluster split into two parts. Though here, the old purple cluster is also split into two clusters which doesn't seem necessary.

From this we can gain that experimenting with different values of k can be worthwhile to get more preferable results, even if the elbow method tells us something else.



(a) $k = 4$

(b) $k = 5$

Figure 6: Different values of k

Problem 3

a.

To motivate the choice of a number of neighborhood samples for a core point we need to take into account the level of noise, the density and the cluster size of our data. Just by looking at the scatter plot it would seem most reasonable to have three clusters, one right cluster and two left cluster, top and bottom. Now with DBSCAN we have a better chance at getting these results since these clusters are of irregular size and shape, something that DBSCAN can handle better than K-means.

Choosing minimum neighborhood samples and distance, we simply went with trial and error. We adjusted our parameters after how the clustering looked. The final result was $\epsilon = 0.2$ & min samples = 24. Using these parameters we get the clustering seen in fig7.

Using any lower value of min samples would give us more than three clusters since small groupings of data points that are now outliers in the other clusters vicinity would be added to small new clusters. This is a bit of a trade off, since we now have a lot of outliers due to how noisy the data is.

Ultimately, hese paremeter choices give us the three main clusters that we were looking for.

b.

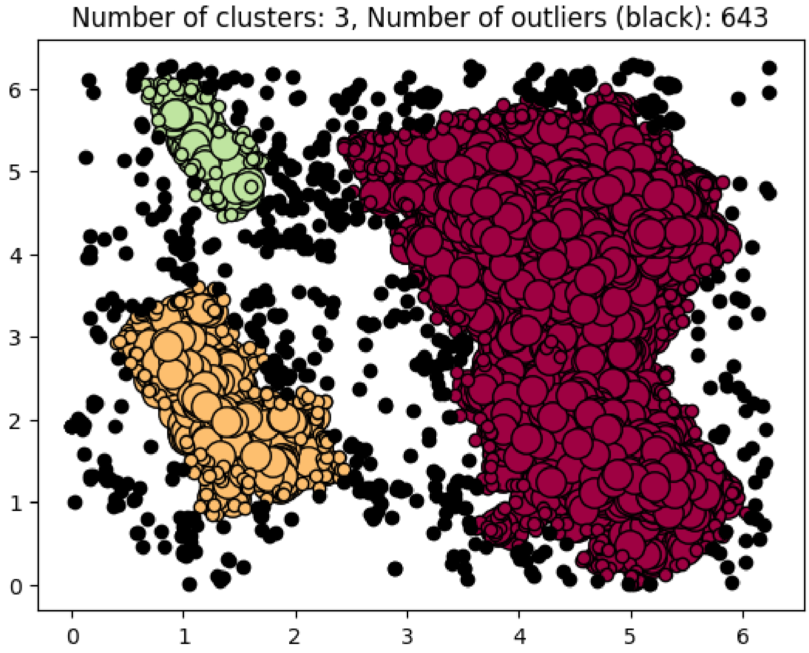


Figure 7: DBSCAN with $\epsilon = 0.2$ & min samples = 24

c.

643 outliers are found by DBSCAN. Just looking by the eye, some of them should belong to the main clusters. But as mentioned before, attempting to include them can create small unwanted clusters.

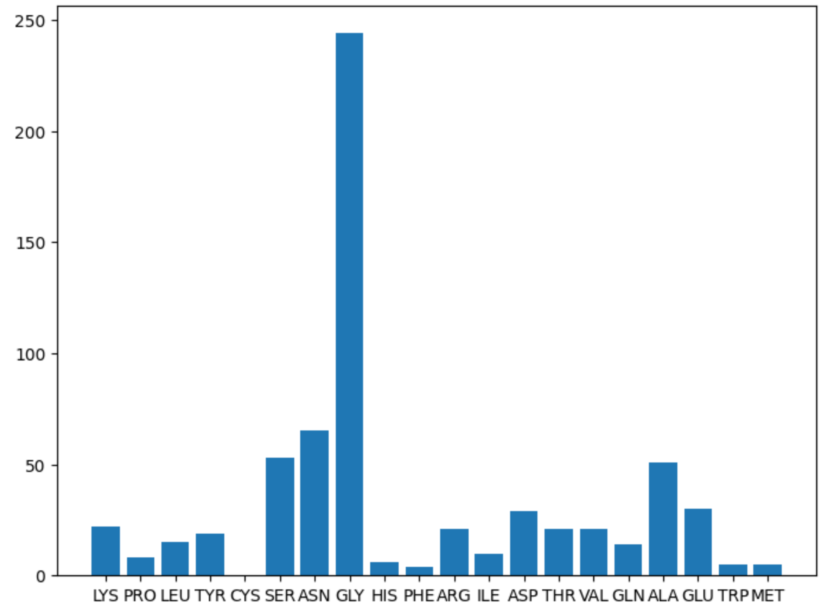


Figure 8: Bar chart over outliers.

A noticable outlier is GLY as can be seen in fig8.

d.

Clearly, DBSCAN does a better job at creating the three main clusters that we can spot by eye. As mentioned, since the data clusters have irregular shapes and sizes, K-means have a hard time clustering the data properly. The noise of the data also adds the problemacy here. With so many outliers and noisy data, the centroids of K-means are skewed towards these outliers which create undesired results.

This is where DBSCAN does a better job at clustering. The irregular shapes and sizes is not a problem for DBSCAN. The noise still makes clustering a bit harder since some outliers can be grouped into very small clusters, which don't really make sense. The density of the data in some parts also creates problems. In less dense areas, like around $phi = 2, psi = 2.5$ in fig7, the outliers group in the crevice in the yellow cluster isn't dense enough to be included in the yellow cluster, even though it looks fairly obvious it should to the eye. This issue can of course be fixed by tweaking the paramters, but then again, we would have unwanted smaller clusters pop up as a side effect.

Problem 4

Figure 9 below illustrates DBSCAN clustering for amino acid containing residue of type ”PRO”.

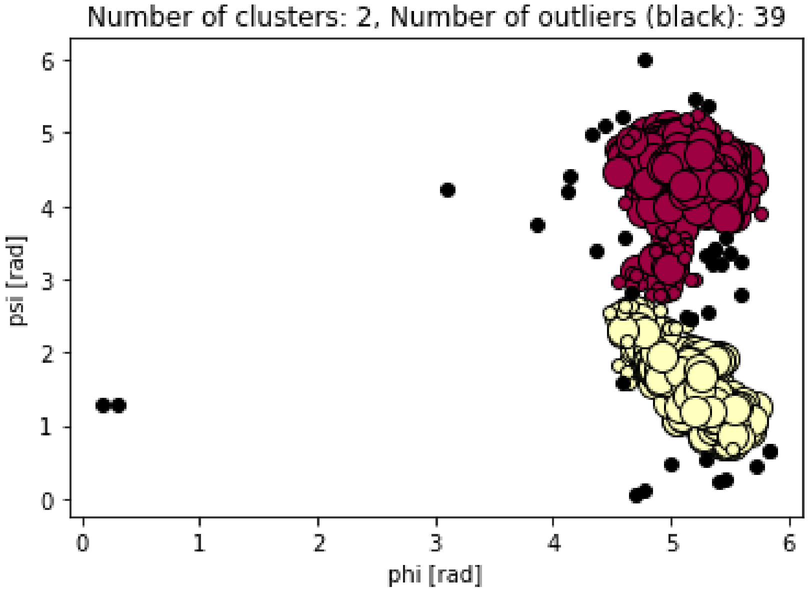


Figure 9: Clustered generated for amino acids with residue type ”PRO” using DBSCAN.

There are significant differences between the clusters identified for PRO amino acid residues and the general clusters, as evidenced by the two estimated clusters compared to three clusters in Task 3b. Moreover, the clusters have different shapes, and the number of outliers has decreased significantly from 643 to 60.

Despite changes in the data, DBSCAN continues to effectively cluster the data, albeit with some data being assigned to new clusters of different sizes compared to the clusters in Task 3b. As previously mentioned, DBSCAN is a useful clustering method for handling outliers, as is the case with this dataset. Unlike k-means clustering, it is capable of filtering out noisy data.

Due to their irregular shape, clustering the clusters with k-means would have been challenging. The parameters were selected through trial and error, with ϵ being increased from 0.2 to 0.25 and *min samples* remained the same.