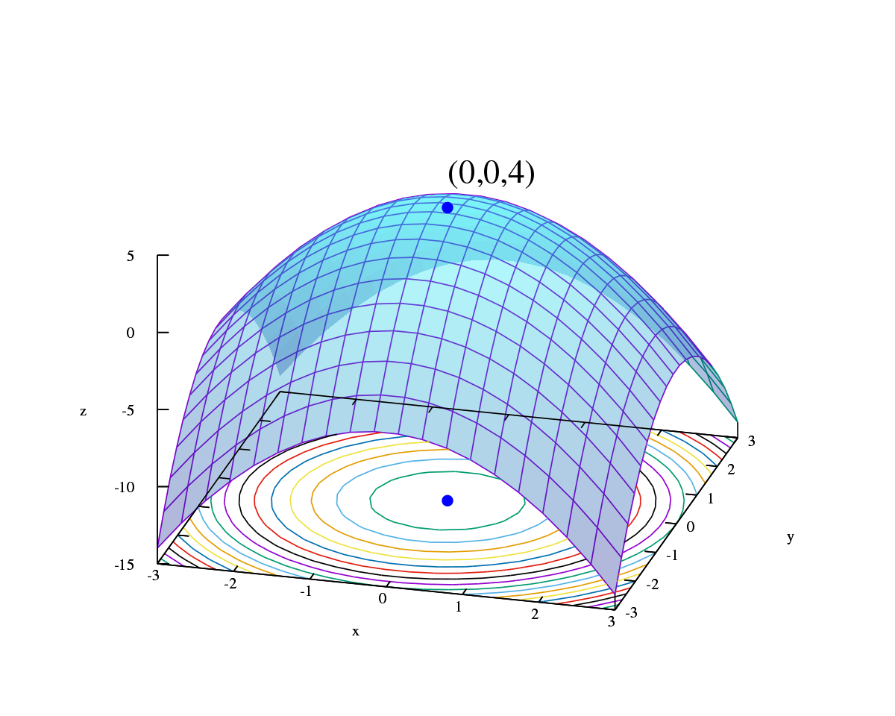
**DAT405: Introduction to data science and AI**

Module 3

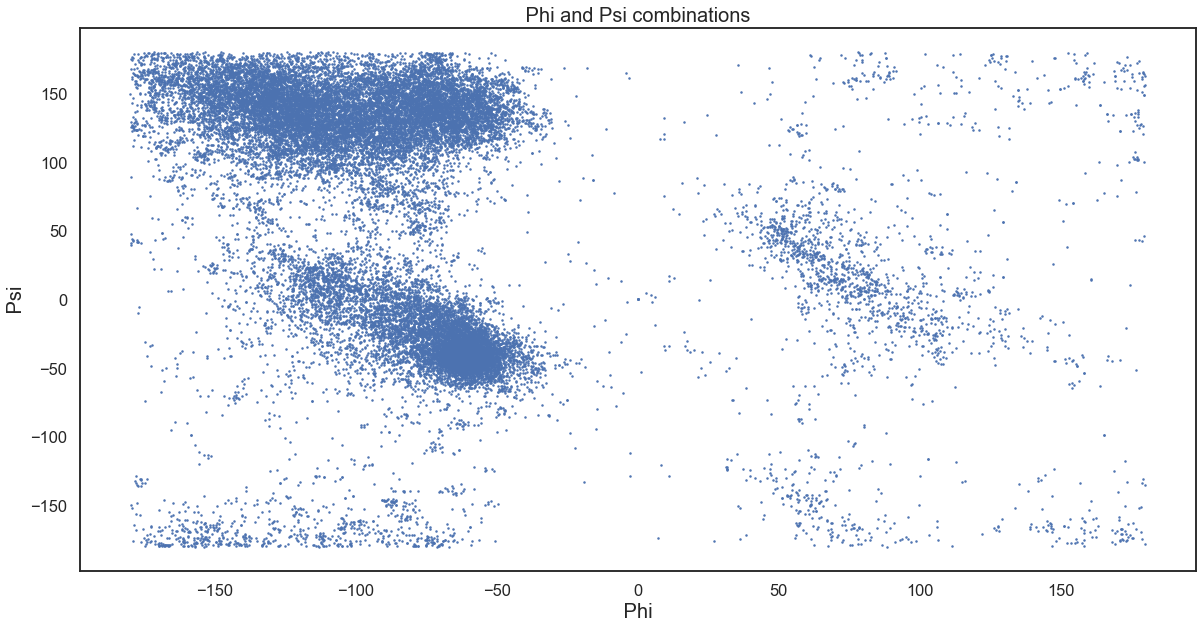


BROU BONI Joël

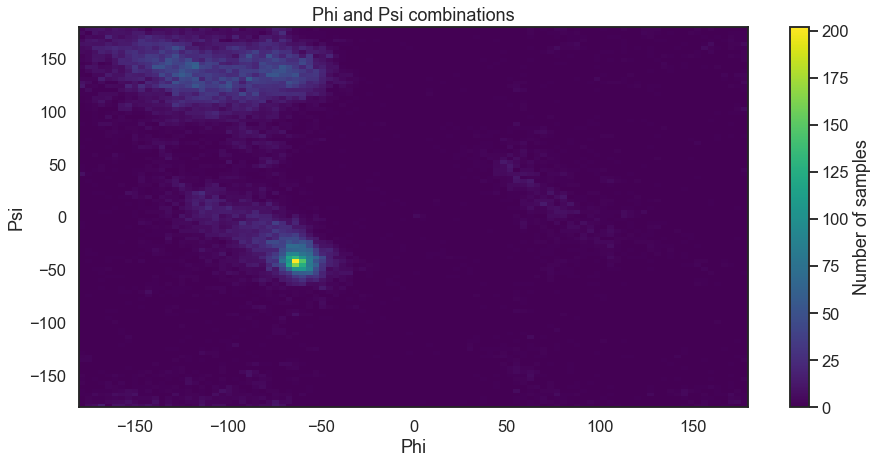
SANOGO Ibrahim Bechir

# Part 1

1. After downloading the dataset related to the combinations of Phi and Psi, we wrote a Python program that draws a scatter plot using this two information.

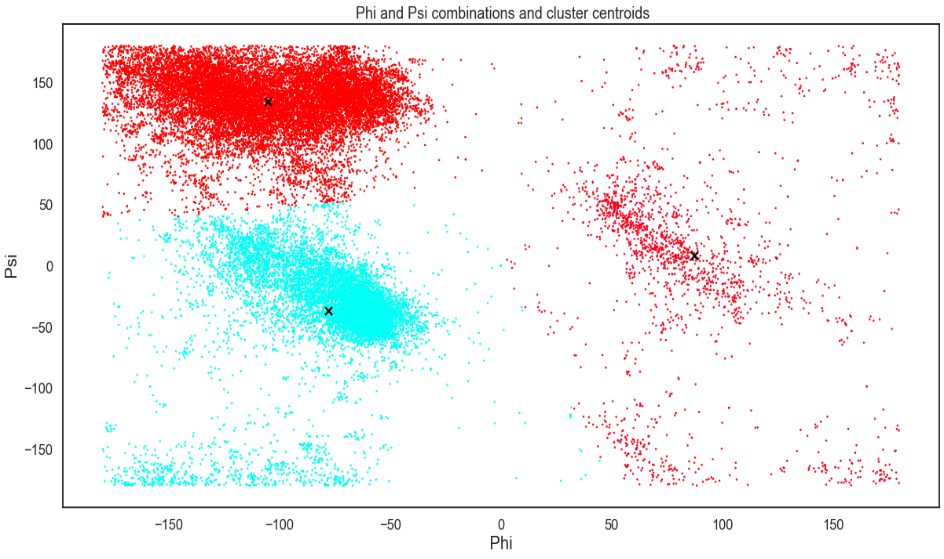


1. Moreover, we draw the distribution of phi and psi combinations using a heat map.

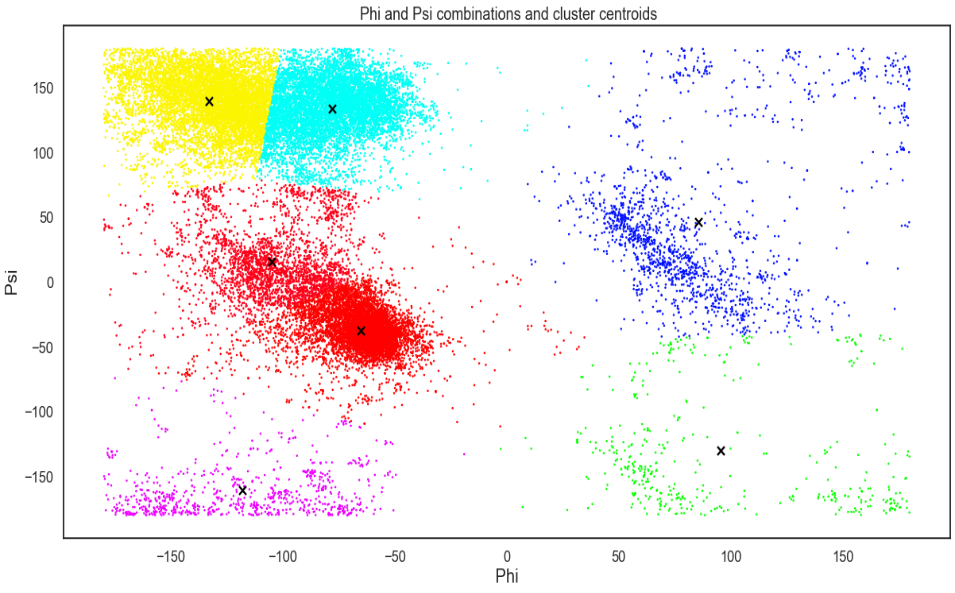


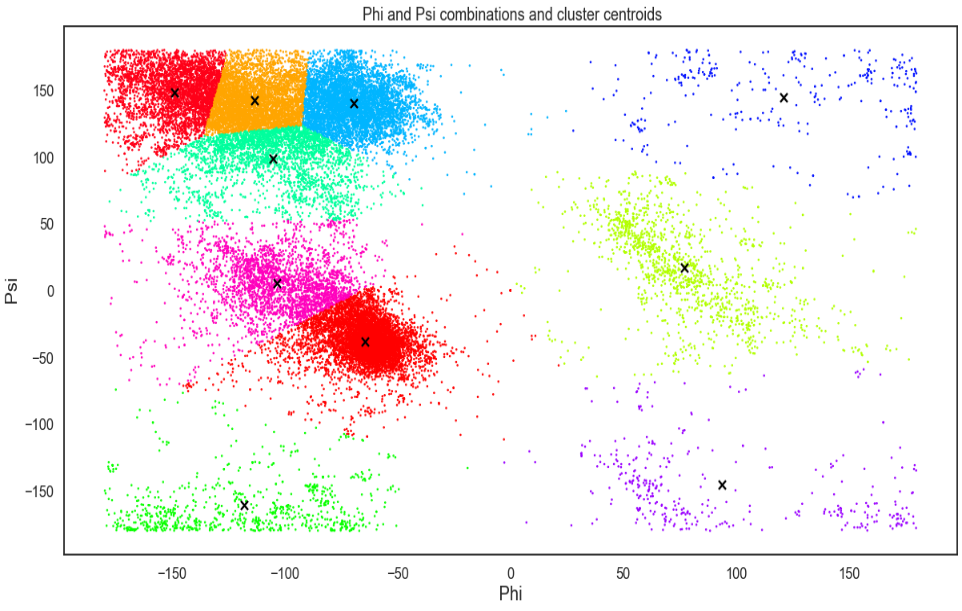
Part 2

In this part we will use the K-means clustering method to cluster the phi and psi angle combinations given in the data file.

1. We experimented different values of K for the clustering.

**K=3**

We can clearly identify 3 clusters. It looks similar to the three observed patterns in the heatmap and the scatter plot and it simple to understand!



**K=7**

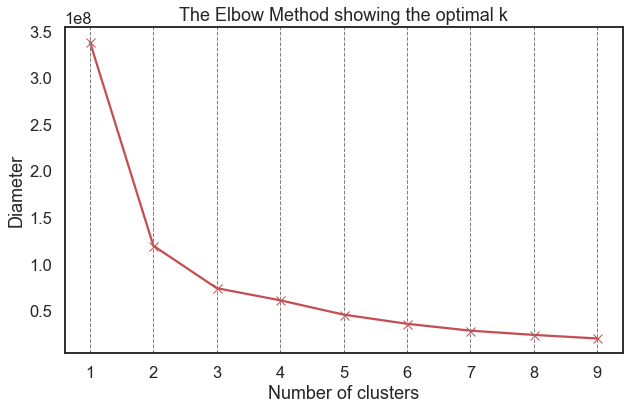
We observed clusters of approximately similar size, as K-means will always assign an object to the nearest centroid. This often leads to incorrectly cut border.

**K=10**

The clusters previously observed are totally fragmented. There are many mini clusters of similar size and it is complicated to extract information from this form of clustering.

We propose to validate the model with K=3 clusters and we will motivate the reason in the next question

1. The **Elbow Method** helps to determine this optimal value of k for K-means.

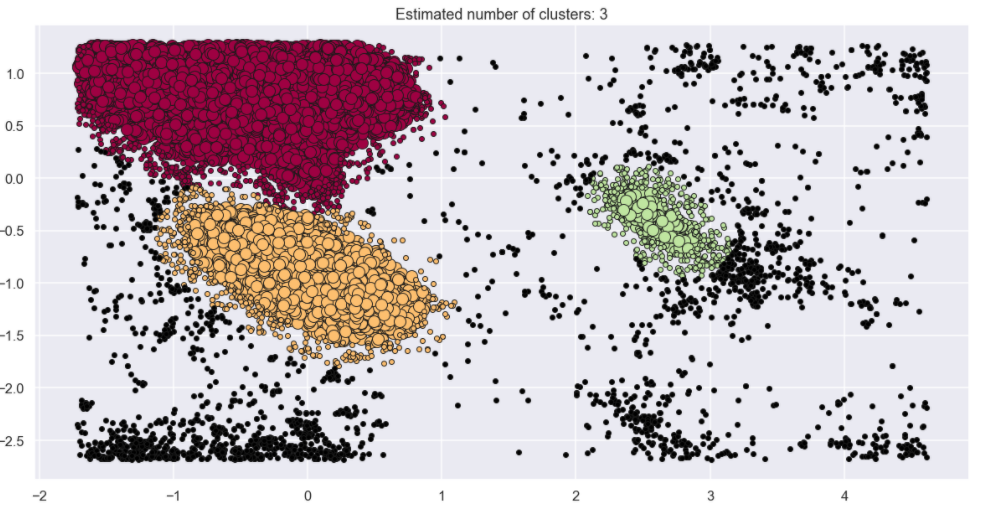
 We iterate the values of k from 1 to 9 and calculate the values of distortions for each value of k and calculate the distortion and inertia for each value of k in the given range.

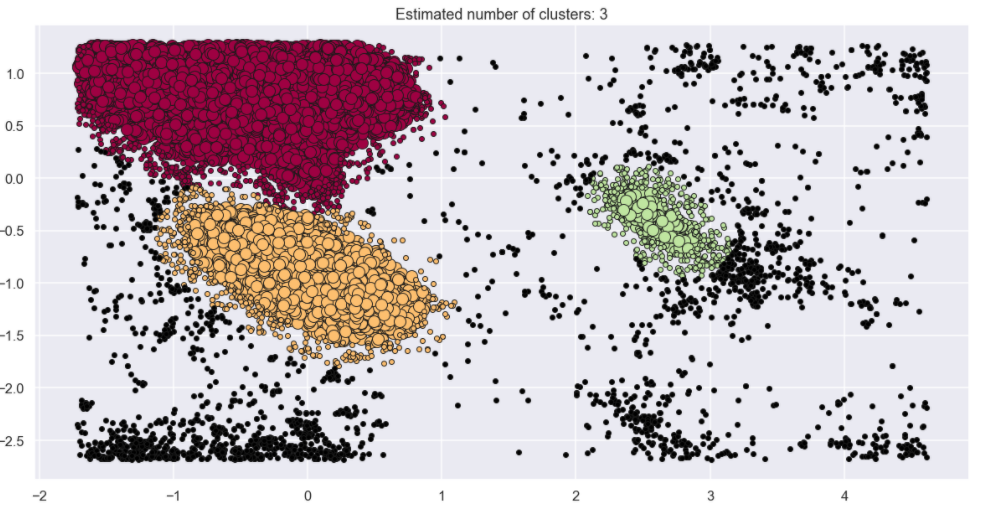
To determine the optimal number of clusters, we selected the value of k at the “elbow”: the point after which the diameter start decreasing in a linear fashion. For the given data, we conclude that the optimal number of clusters for the data is **3**

Moreover, the arguments for the validation this clustering were all verified: stability over repetitions, co-occurrence, stability over subsets…

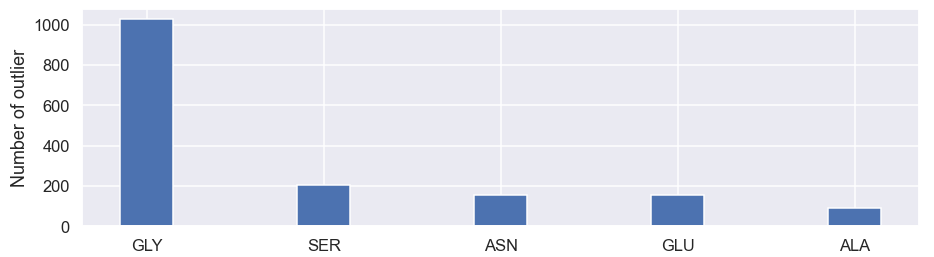
1. Yes, the three clusters found in part (a) seem reasonable.
2. Aucune idée pour l’instant

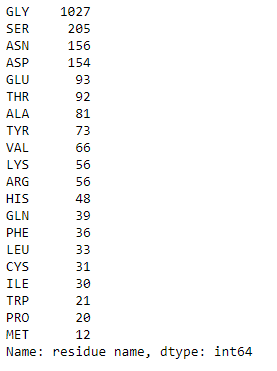
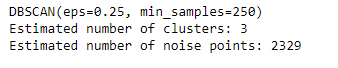
Part 3

1. DBSCAN is an algorithm than can be applied across various applications. The input parameters '***eps***' and '***minPts***' should be chosen guided by the problem domain. A larger value for eps results in broader clusters, while a smaller value establishes narrower clusters. The minPts parameter specifies the minimum number of points (i.e., cluster members) required to produce a new cluster. A larger value of minPts assures a more robust cluster but may exclude some potentially smaller areas as it attempts to merge them in a larger one. On the other hand, a smaller value extracts many clusters, but the resultant clusters may include noise as well. Based on our application and the number of datas we had, we decided to choose, an eps=0.22
2. Our scatter plot allows clear distinction of the characteristic regions of α-helices and β-sheets. The regions on the plot with the highest density of dots are the so-called “allowed” regions, also called low-energy regions. Some values of φ and ψ are forbidden since they will bring the atoms too close to each other, resulting in a so-called steric clash. We then decided to set up our model so It could highlight all the things said right before. For the min point we decided to have robust clusters so the DBSCAN could only detect the biggest point concentration. Knowing the number of data points, we had, we supposed that we in the low-energy regions they’re would be more that 250 points. This hypothesis seems to be right indeed, having for example a min points >20 for the eps value we fixed we get more than 20 clusters which is not what we want. The value of eps was then selected in such a way to make a compromise between a large scatter plot a small one. A higher value of eps would concatenate the three scatter plots on the left together and a smaller one would non coherent noise points.

After setting up the DBScan algorithm we got the following result. Our algorithm have found 3 clusters and 2300 noise plots. Our plot seeking to highlight characteristic regions of α-helices and β-sheets we are satisfied with the result. Moreover, it is matching with the heatmap e got in the previous questions.

To find which amino acid residue was most frequently an outlier we decided to make the following bar plot using pandas and matplotlib. We got the result downwards for the 5 residuals with the most outliers.

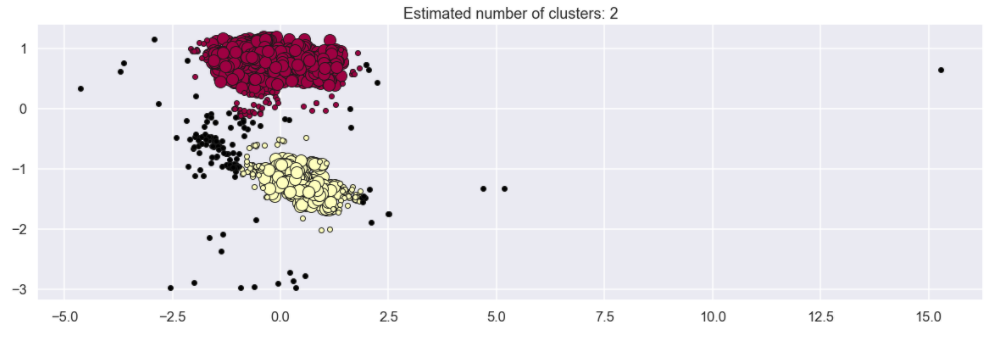




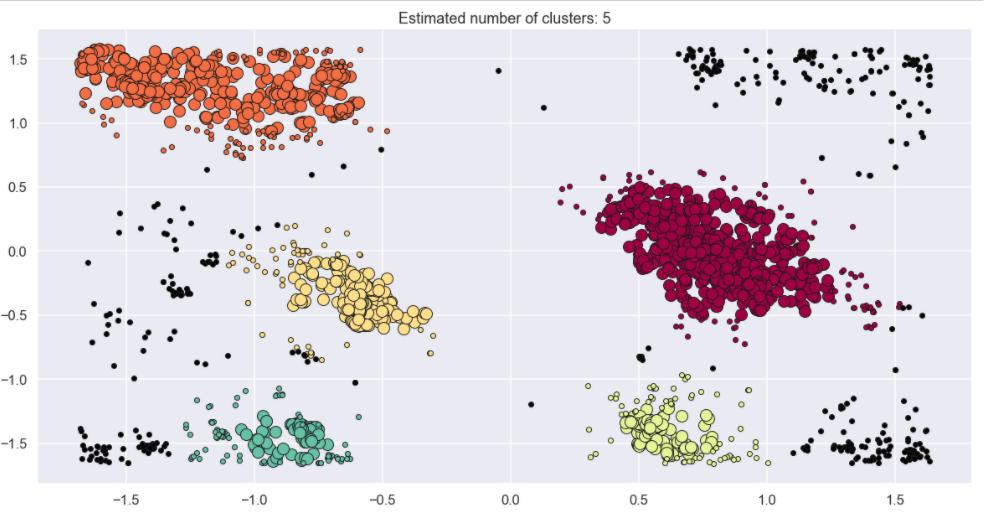
1. The cluster that we got from K-means and DBSAN are different. The clusters from Kmeans includes the outliers and non-coherent point whereas the DBSCAN excludes them. But at the same time the DBSCAN excludes some points that could be coherent just like we can see with the green cluster.

Part 4

1. After using the DBSCAN algorithm to cluster the data that have residue type PRO, we have the following result. The parameters of the DB Scan Model have been changed in order to make the clustering correctly (eps =0.5, minPts=100). As we can see most of the the residue type PRO are reparted in the the two biggest clusters alpha and Beta. This can explain why we don’t have a lot of outliers from this type.

Here we only have two clusters compared to 3 before. Moreover, it seems that we have percentage of outliers less important. This is because most of the points are in the same areas.

1. After using the DBSCAN algorithm to cluster the data that have residue type GLY, we have the following result. The parameters of the DB Scan Model have been changed in order to make the clustering correctly (eps =0.3, minPts=100). As we can see most of the residue type GLY are in different zones of the plot and a lot of them are not in the cluster from part 3. This is why the glycine is the residual with the most outliers.



Here we have 5 clusters compared to 3 before. The result looks coherent by looking at the scatter. Here we have more outliers than the previous case this is can be explain by the point repartitions, they not concentrated in the same just as before.