Homework 4

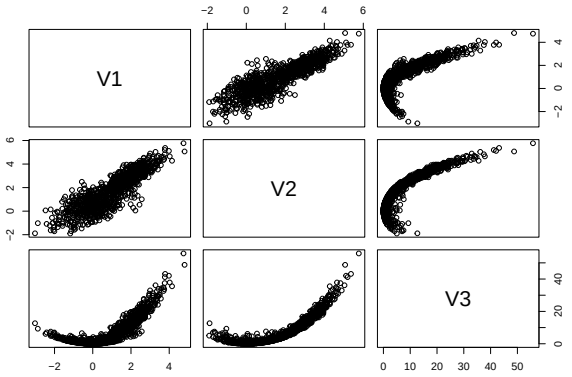
Giang Vu

1. Exploratory cluster analysis of each data set

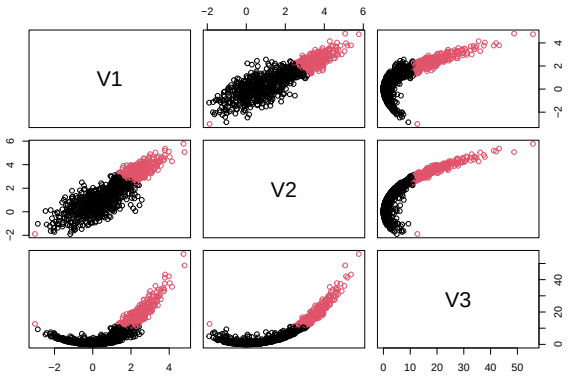
Below is the results of exploratory cluster analysis of the three given data sets. With each of them, I applied three methods, *k-means*, *Gaussian mixture modeling*, and *hierarchical clustering*. To demonstrate the results, I included interactive (multidimensional) plots of the data with colors corresponding to the cluster outputs of each method.

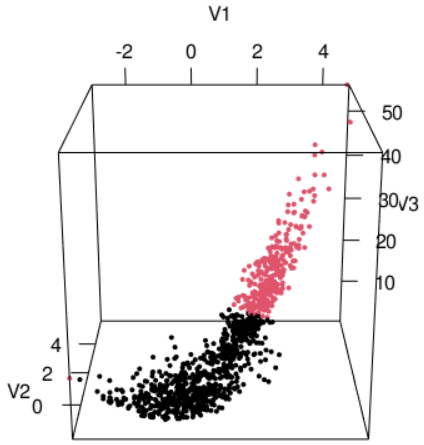
(a) Data set "data1"

Before applying any methods, I tried making pair-wise scatter plots for the three variables in this data set. There seems to be a linear correlation between X_1 and X_2, while the scatter plots of X_1 and X_2 versus X_3 both have a curve pattern.
Looking at the scatter plot between X_1 and X_2 alone, there seems to be two groups of points separated in the middle of the plot, one group has points that are more spread out, and the other one is more packed together.
Therefore, I decided to pre-specify the number of clusters for this data set is $k = 2$.

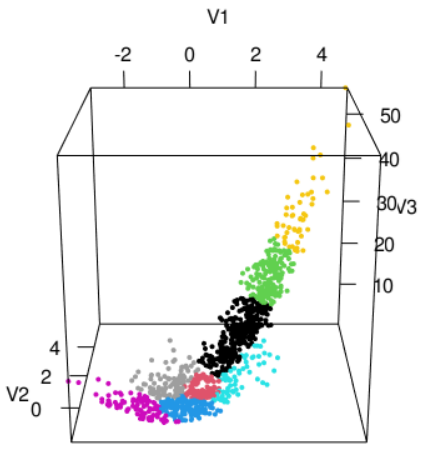


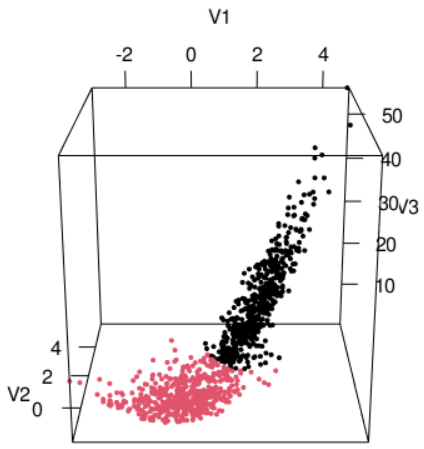
k-means The k-means method gave me the cluster outputs below, which agrees with my assumption about the 2 groups of points that I described above.
The **advantages** of this method include that it's relatively simple to implement, it scales to large data sets, and always guarantees to converge.
However, it does have certain **drawbacks**, like it is dependent on our initial choice of k, which we had to choose manually like what I did above. And to choose this number k is not always easy like in our example, especially if you have many variables. And because of the usage of the Euclidean norm in the algorithm, this method implicitly assumes spherical clusters and balanced cluster sizes, which will not apply well to data with non-spherical cluster shape.



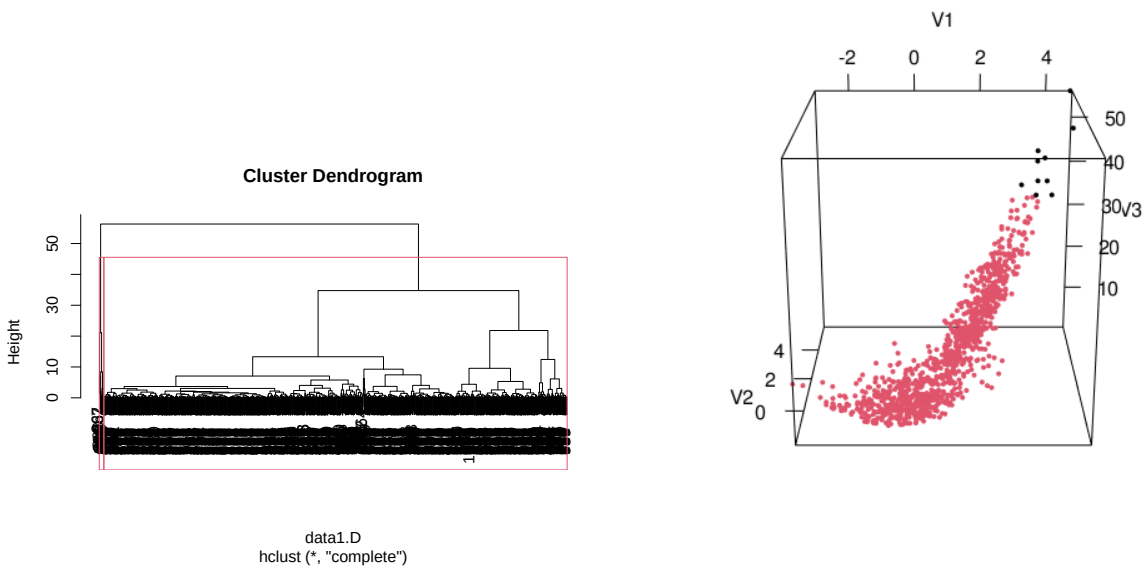


Gaussian mixture For the Gaussian mixture modelling approach, if we don't specify the number of clusters beforehand, it will give us 9 clusters like it gave me below.
When I specified the number of clusters to be 2, it gave me a result that's a little different from the result of k-means. The cutoff that separates the two clusters is different from k-means.
The **advantages** of Gaussian mixture modelling include its flexibility (it doesn't assume spherical clusters like k-means), and the fact that it allows a points to belong in multiple clusters (unlike k-means where a point can only belong to one cluster). The flexibility can be seen quite clearly in the default output with 9 clusters below.
The **cons** of this method include that it's not as fast and easy to implement like k-means, and thus it doesn't scale well to very large data sets with large dimensions.



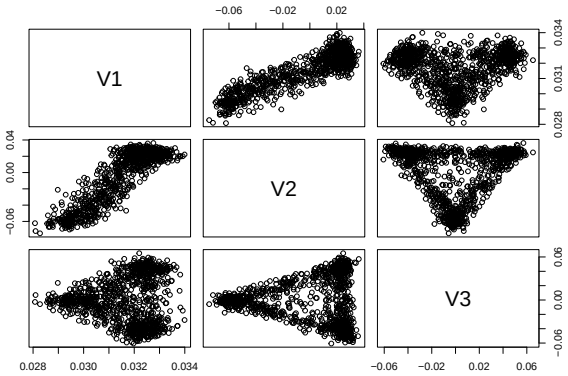


Hierarchical With the hierarchical clustering method, I include the dendrogram as well as the result cluster 3D plot below. Because of the initial assumption of 2 clusters, I cut the dendrogram such that $k = 2$, and as we can see from the 3D plot, this is a very different result compared to the previous methods. Most of the data are in one cluster, while only 10 of the points belong in the other cluster. It doesn't agree with our initial assumption.
About this method, the **advantages** include the dendrogram, which gives us an informative structure of the points, letting us choose the number of k by looking at the dendrogram. It is also fairly easy to implement with small data set.
The *disadvantages* include the inflexibility, once a point has been assigned to a cluster, it can not be moved around. It also doesn't scale well for larger data sets and this method is very sensitive to the order of the data as well as outliers compared to k-means and Gaussian mixture modeling.

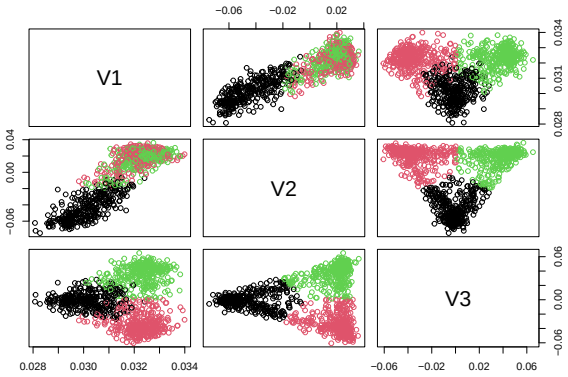


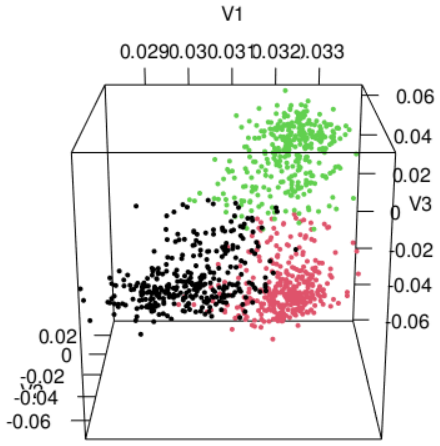
(b) Data set "data2"

Looking at the scatter plot of the data, I pre-specified $k = 3$ based on the triangular pattern in the plot, and then applied the 3 methods to the data set just like what I did for data set 1.

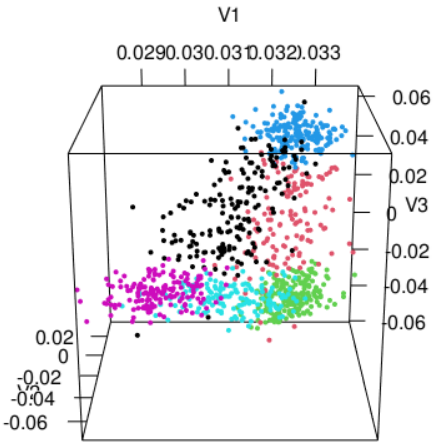


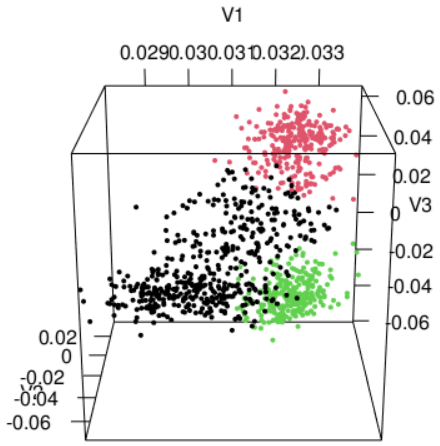
k-means The k-means method gave clear cluster outputs that agree with my assumption. The clusters are partitioned by the 3 corners of the triangular shape of the data set.



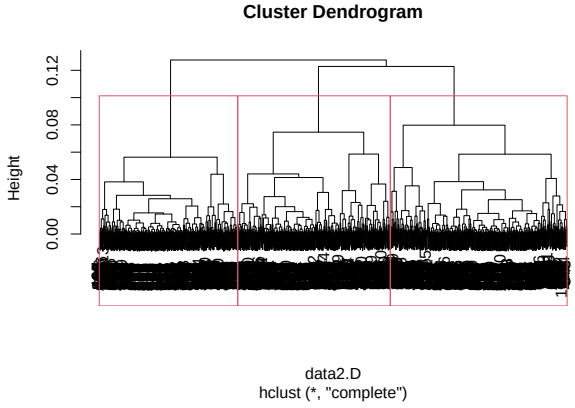


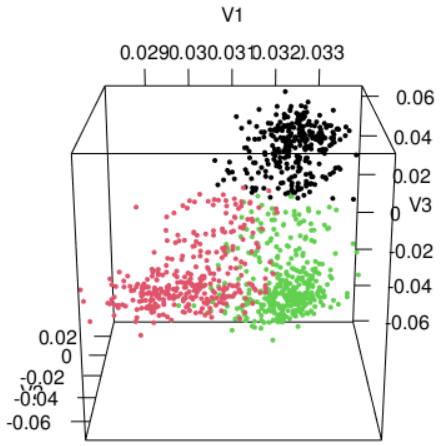
Gaussian mixture For the Gaussian mixture modelling approach, if we don't specify the number of clusters beforehand, it will give us 6 clusters like it gave me below.
When I specified the number of clusters to be 3, it gave me a result that's a little different from the result of k-means. The cutoff that separates the 3 clusters is different from k-means, the size of each cluster are not roughly equal to each other like k-means, with one clusters (in black color) taking more points than the other two.





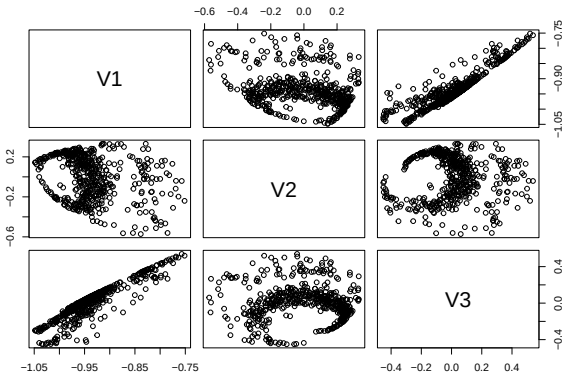
Hierarchical Because of the initial assumption of 3 clusters, I cut the dendrogram such that $k = 3$, and as we can see from the 3D plot, this is a very similar result compared to the k-means method. In fact it is closer to k-means than the Gaussian mixture modeling method. It agrees with our initial assumption.

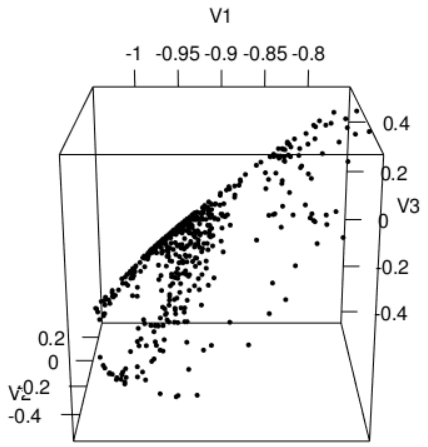




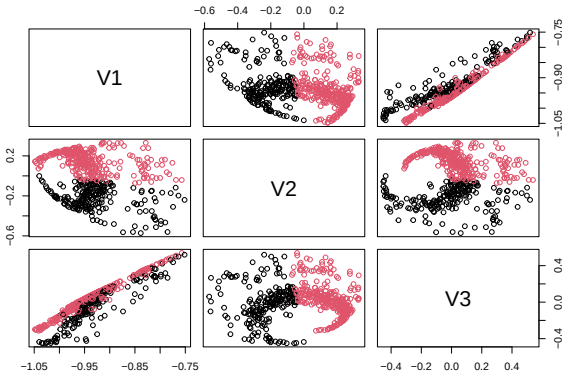
(c) Data set "data3"

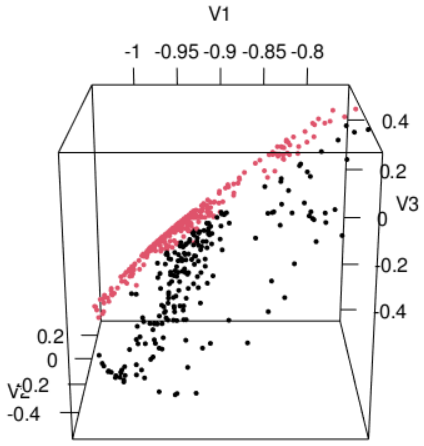
Looking at the scatter plot of the data, I pre-specified $k = 2$ based on the pattern in the plot, where the first cluster is the spread out points the end edge of the fan shape, and the rest of the points that are crammed together like a letter "c" is the other cluster. There's a clear separation between the two parts, that's the reason for my assumption.



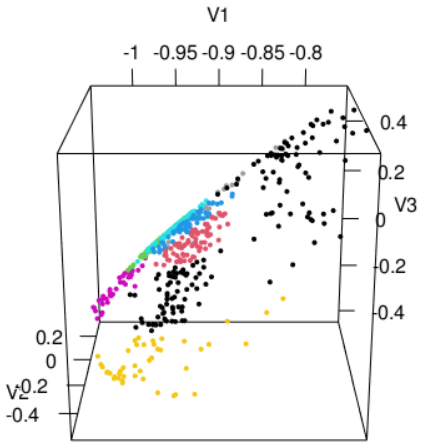


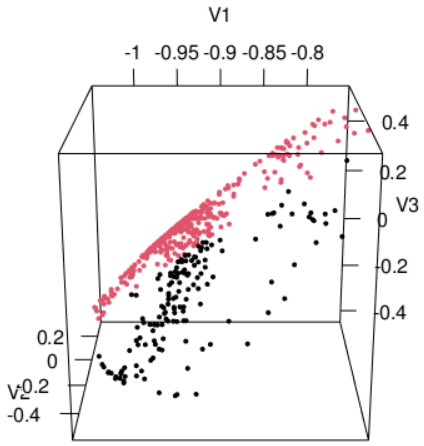
k-means The k-means method gave cluster outputs that doesn't agree with my assumption. The method cuts the "c" shape horizontally, with one cluster on top and one on the bottom.





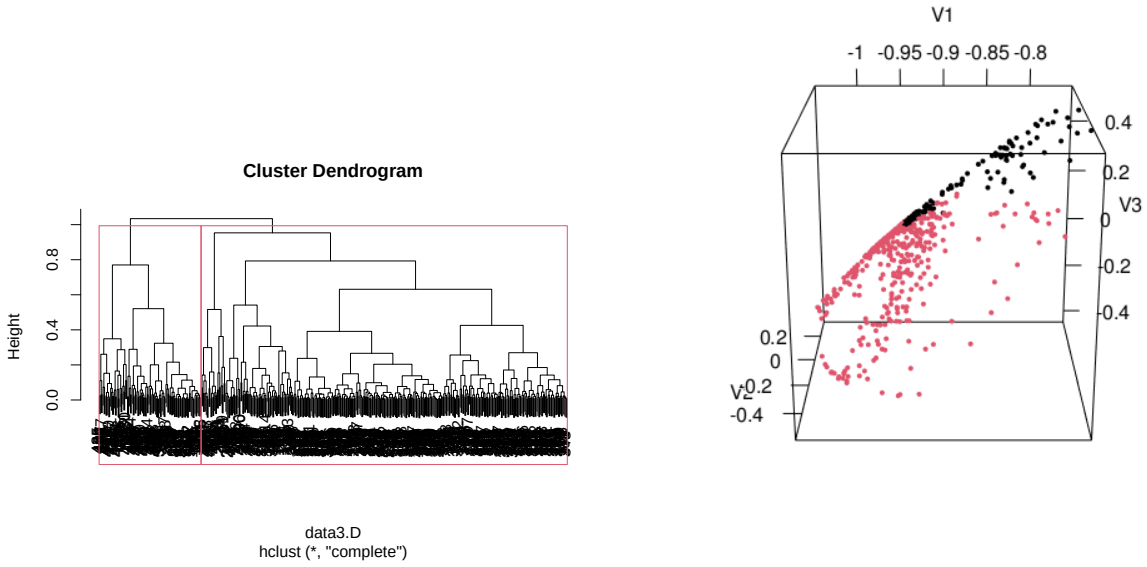
Gaussian mixture For the Gaussian mixture modelling approach, if we don't specify the number of clusters beforehand, it will give us 8 clusters like it gave me below.
When I specified the number of clusters to be 2, it gave me a result that's very similar to the result of k-means. The cutoff that separates the 2 clusters is also at the middle of the "c" shape.





Hierarchical Because of the initial assumption of 2 clusters, I cut the dendrogram such that $k = 2$, and as we can see from the 3D plot, this is a little different from the previous 2 methods, and a little closer to

my initial assumption, but not exactly in line. The entire "c" shape roughly belongs to one cluster, along with half of the fan shape tail, while the other cluster contains the rest of the fan shape tail.
All 3 methods are far from my assumption, so it could be that my initial assumption is not reasonable for this data set.



2. Compare cluster outputs

Below is the description of each similarity index, as well as the summary of comparison results of our clusterings with the original sequence. For more details, please refer to the full code in my markdown file attached.

(a) Jaccard index

The Jaccard similarity index measures the similarity between two sets of data (applied in clustering, it would be measuring the similarity between two sets of clusterings). This index can range from 0 to 1. The higher the number, the more similar the two sequences.
Suppose we have two clustering sequences A and B, then the Jaccard similarity is calculated below:

$$J_{A,B} = \frac{|A \cap B|}{|A \cup B|}$$

Using this index for *data1.txt*, I compared the clustering outputs of 3 methods to the actual clustering given in *alloc_vecs.txt*, and the comparisons suggest that the Gaussian mixture modeling gives the closest result to the origin data with biggest similarity index, and followed by the k-means method, and finally the hierarchical method with the least similarity to the original vector.
Regarding comparisons among themselves, they're all very different from each other, with the k-means and Gaussian mixture modeling being the most similar.

Using this index for *data2.txt*, the comparisons suggest that the k-means method gave the most similar clustering, followed by the hierarchical method, and finally the Gaussian mixture modeling is the least similar.

When compared with each other, they are all very different as well, with the hierarchical and k-means result being the most similar, while k-means and Gaussian mixture modeling, as well as Gaussian mixtrue modeling and hierarchical are not similar at all.

(b) Rand index

Rand index is a method to compare the similarity of results between two different clustering methods. The calculation is given below:

$$R = \frac{a+b}{{}_nC_2}$$

where a is the number of times a pair of elements belongs to the same cluster across two clustering methods.
b is the number of times a pair of elements belong to difference clusters across two clustering methods.
${}_nC_2$: The number of unordered pairs in a set of n elements.

For *data1.txt*, applying this index to compare our results, I also got that the Gaussian mixture modeling gives the closest result to the origin data with biggest similarity index, and followed by the k-means method, and finally the hierarchical method with the least similarity to the original vector. But now the similarity index of the two methods are much closer to each other than compared to the Jaccard index.
Regarding comparisons among themselves, their similarity score with each other are all around 0.5, so they're not too different but not too similar from each other.

Similarly for *data2.txt*, we obtained the same comparison results as with Jaccard index, where the k-means is the most similar, followed by hierarchical and finally the Gaussian mixture modeling is the least similar.
Regarding comparisons among themselves, their similarity score with each other are all around 0.8, so we can say they're fairly similar with each other.

(c) Adjusted Rand index

Adjusted Rand Index (ARI) is a corrected-for-chance version of RI that establishes a baseline by using the expected similarity of all pair-wise comparisons between clusterings specified by the generalized hypergeometric distribution. The formula is given in our lecture slides.

Applying on *data1.txt*, again we got the same result as the two previous indices, where Gaussian mixture leads in similarity, followed by k-means, and then hierarchical.
Regarding comparisons among themselves, their similarity score with each other are all less than 0.3, so they're quite different from each other, with k-means and Gaussian mixture modeling being the most similar.

Similarly for *data2.txt*, we obtained the same comparison results as with Jaccard and Rand indices, where the k-means is the most similar, followed by hierarchical and finally the Gaussian mixture modeling is the least similar.
Regarding comparisons among themselves, their similarity score range from 0.6 to 0.86, so we can say they're fairly similar with each other, with hierarchical and k-means being the most similar.

(d) Normalized mutual information

A normalized mutual information metric is a mutual information metric whose range is normalized to [0,1]. The formula is given below.

$$NMI(X,Y) = \frac{I(X,Y)}{\sqrt{H(X)H(Y)}}$$

Where
$I()$ is the mutual information metric (a metric that measures the mutual dependence of two random variables X and Y)
$H()$ is the entropy metric (a metric that measures the uncertainty associated with a random variable)

I used the function **NMI()** from the package **aricode** to carry out comparisions of the 3 clustering results we got for each data with the true clustering using the NMI index.
The result is, again, exactly the same like the previous indices.

For *data1.txt*, Gaussian mixture modeling result is closest to the truth, followed by k-means, and then hierarchical.
Regarding comparisons among themselves, their similarity score with each other are all less than 0.4, so they're quite different from each other, with k-means and Gaussian mixture modeling being the most similar.

Similarly for *data2.txt*, where the k-means result is the most similar to the truth, followed by hierarchical and finally the Gaussian mixture modeling is the least similar.
Regarding comparisons among themselves, their similarity score with each other range from 0.65 to 0.8, so they're quite similar with each other, with k-means and hierarchical being the most similar.