**Introduction to Bioinformatic – Assignment 3**

By: Liron Naccache, Seggev Haimovich, William Manuel, and Jayden Johnson

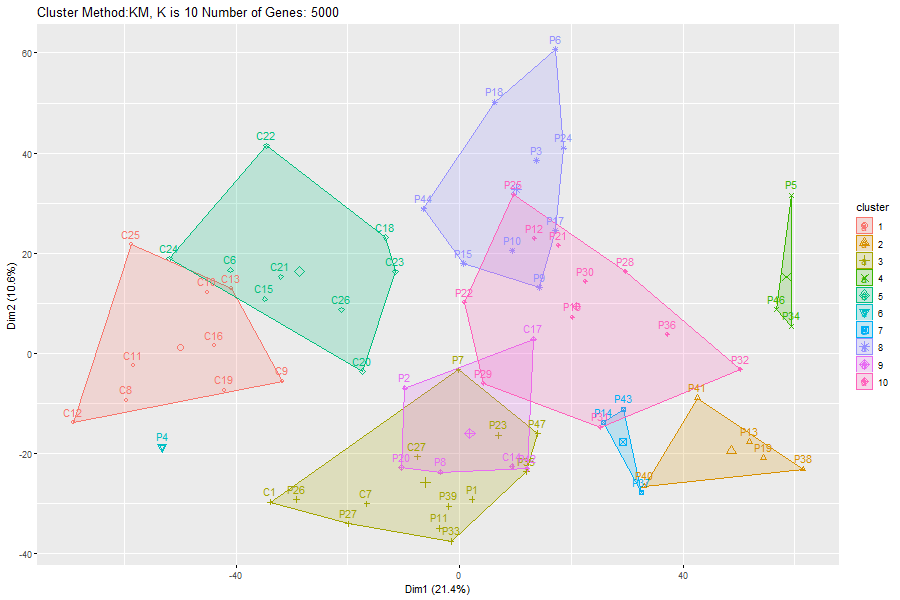
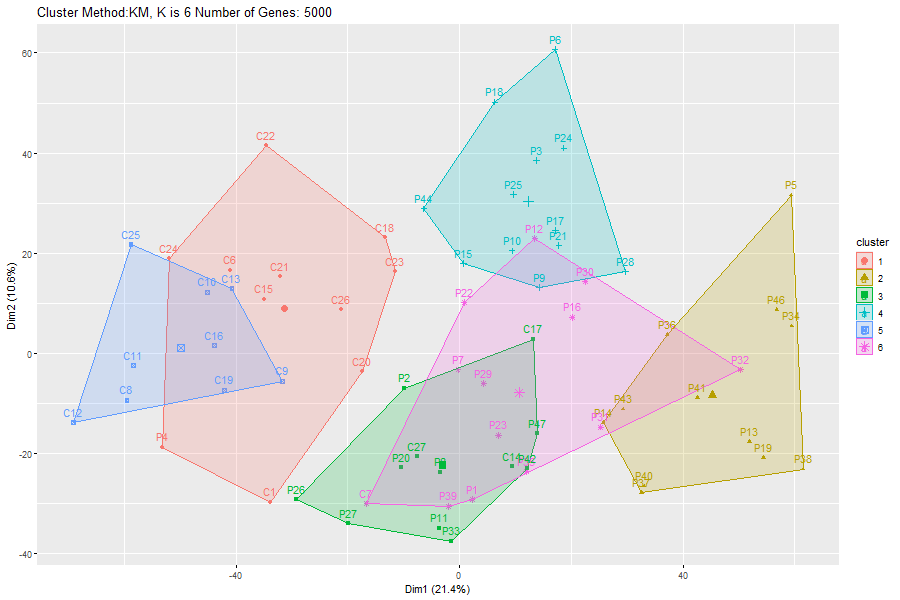
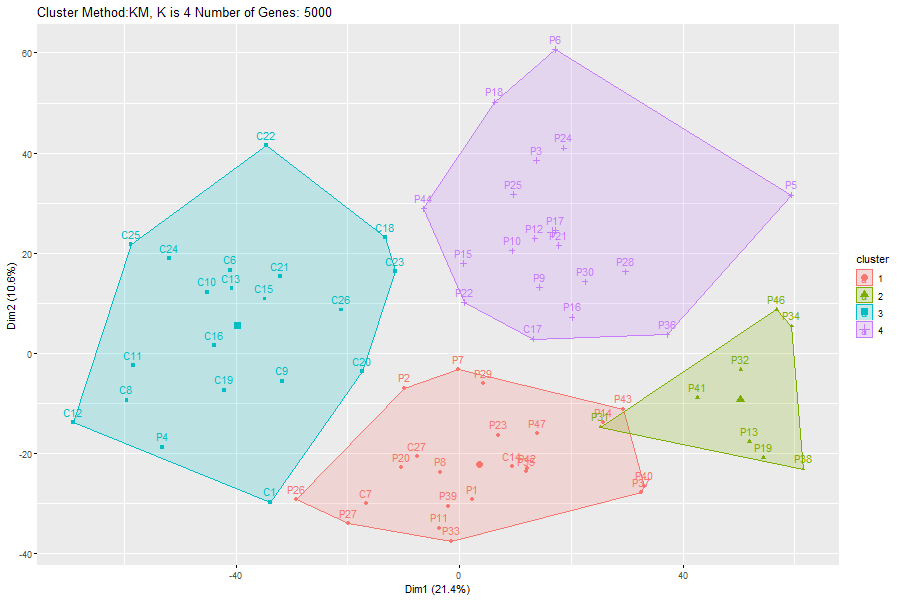
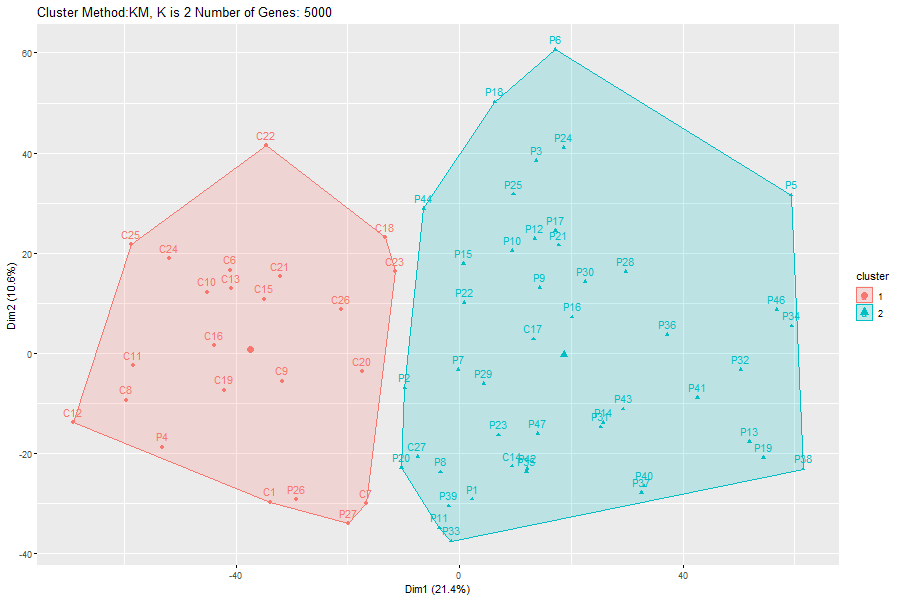
[GitHub](https://github.com/lironnaca/Bioinformatics-Project)

**Unsupervised Analysis**

We incorporated 4 different clustering algorithms within this assignment.

Each algorithm was completed 4 times, using the following K’s (number of clusters): 2, 4, 6, 10

1. K-means

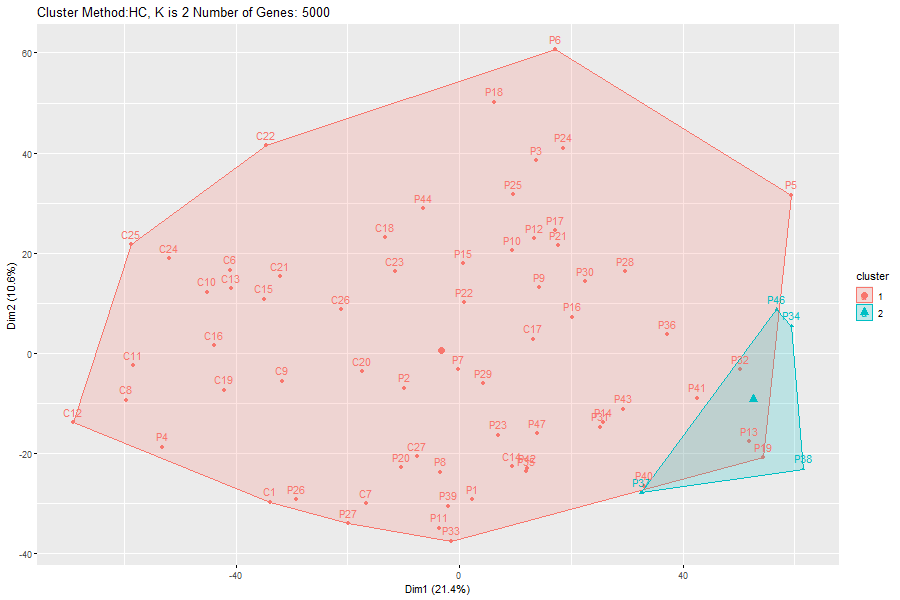
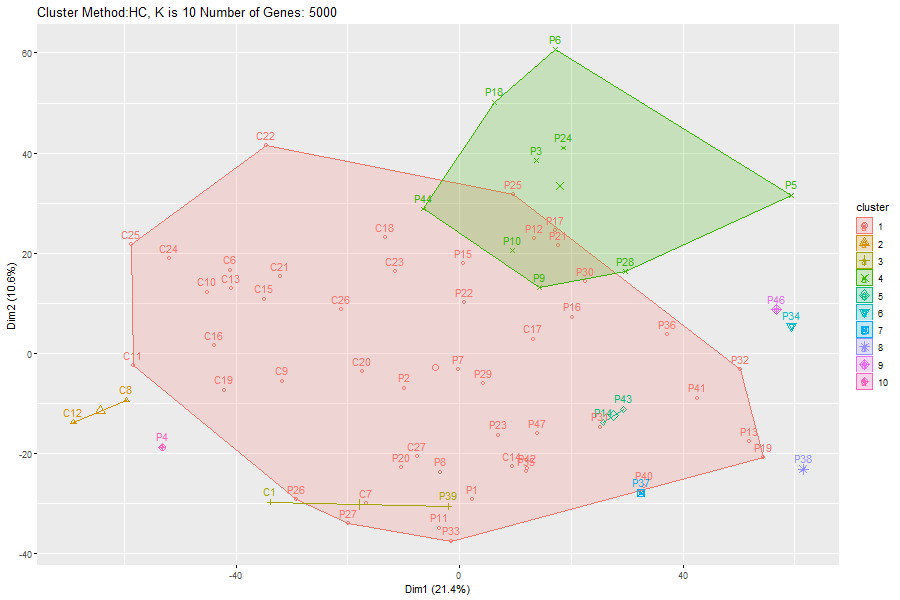
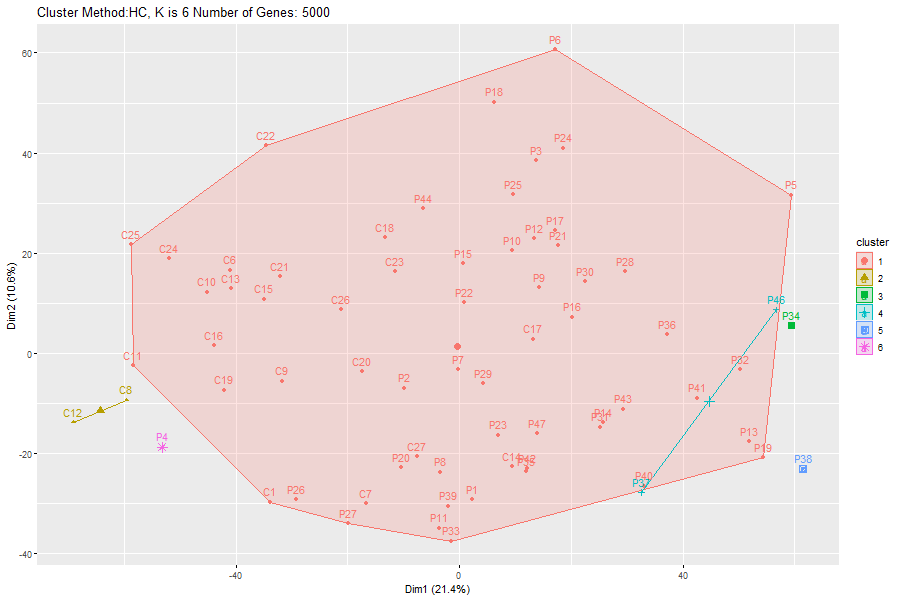
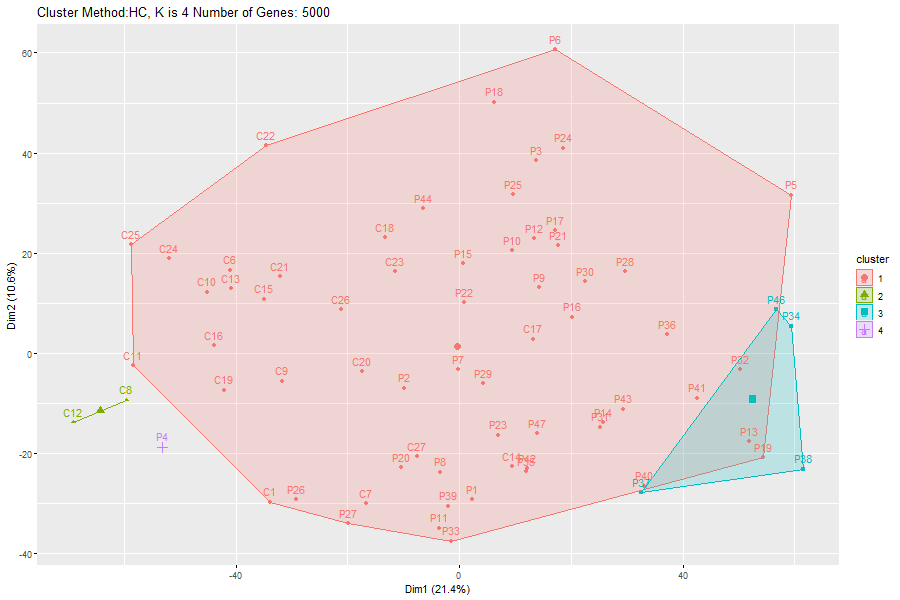


We can see that the bigger the K the more clusters there are.

In the k=2 clustering, we see that the cluster groups are very alike to our patient groups (p: critical, c: non-critical).

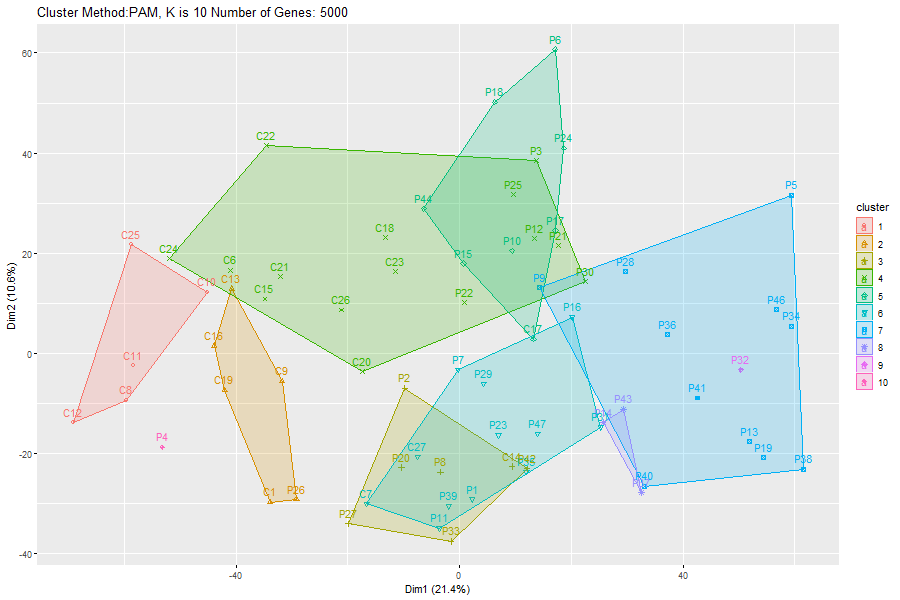
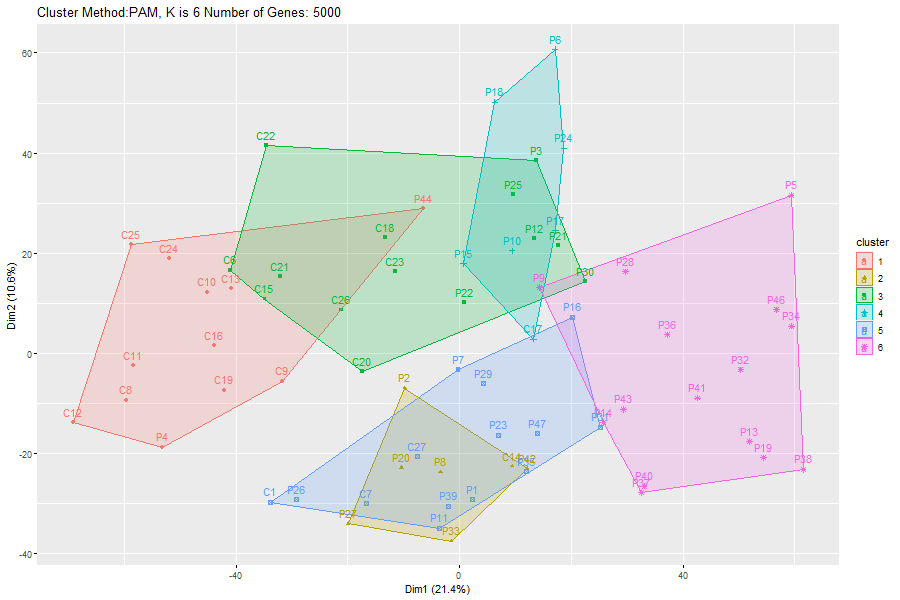
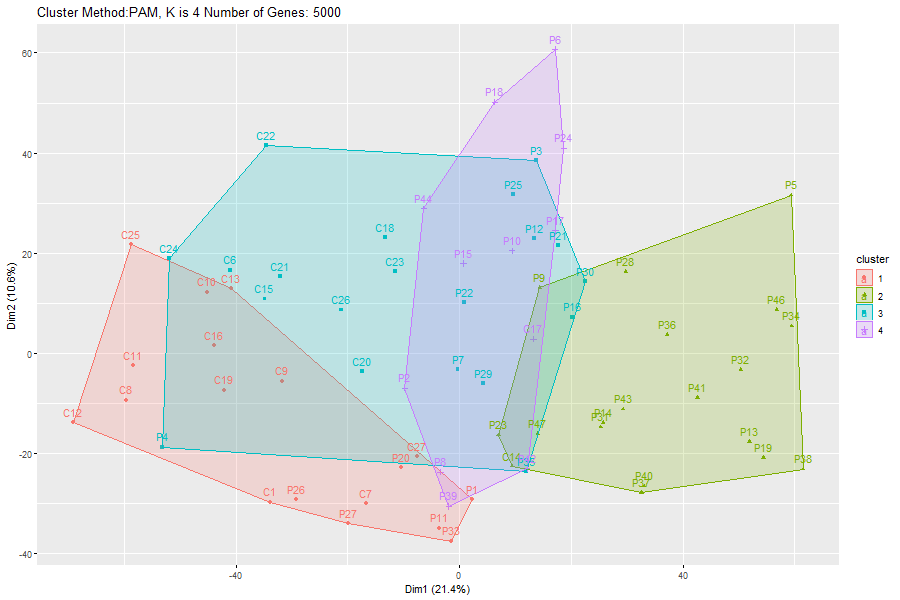
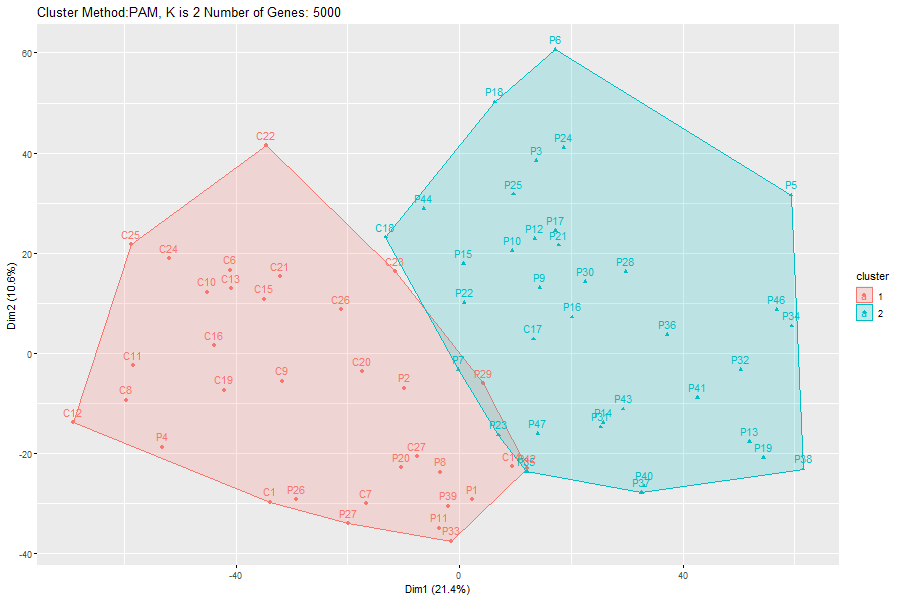
In addition, we see that even with 10 clusters, there are still clusters that contain both critical and non-critical patients.

1. Hierarchical clustering (hclust)



Our findings here signify that the algorithm had no success in separating the real group (critical and non-critical patients), no matter how many clusters we defined. Increasing the number of clusters had very little effect on the outcome.

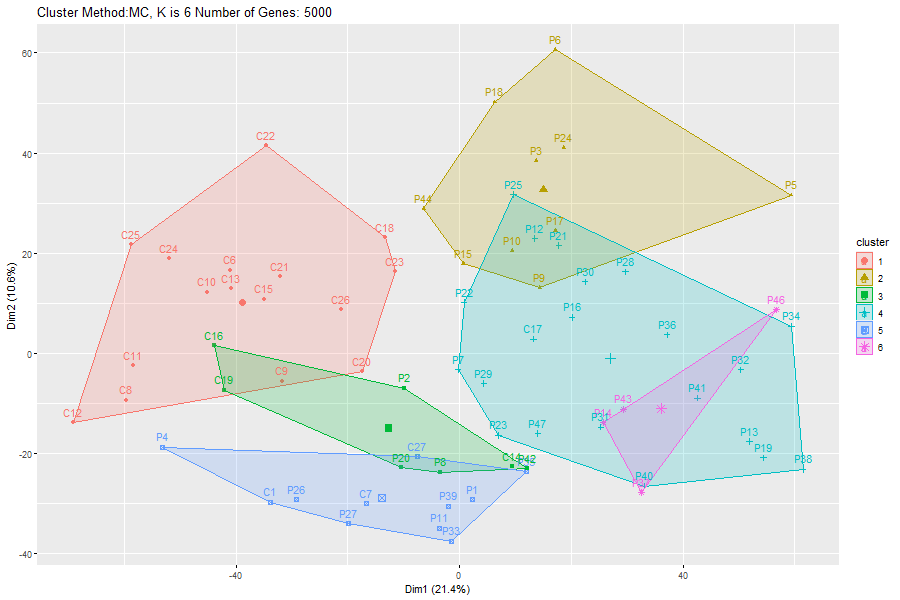
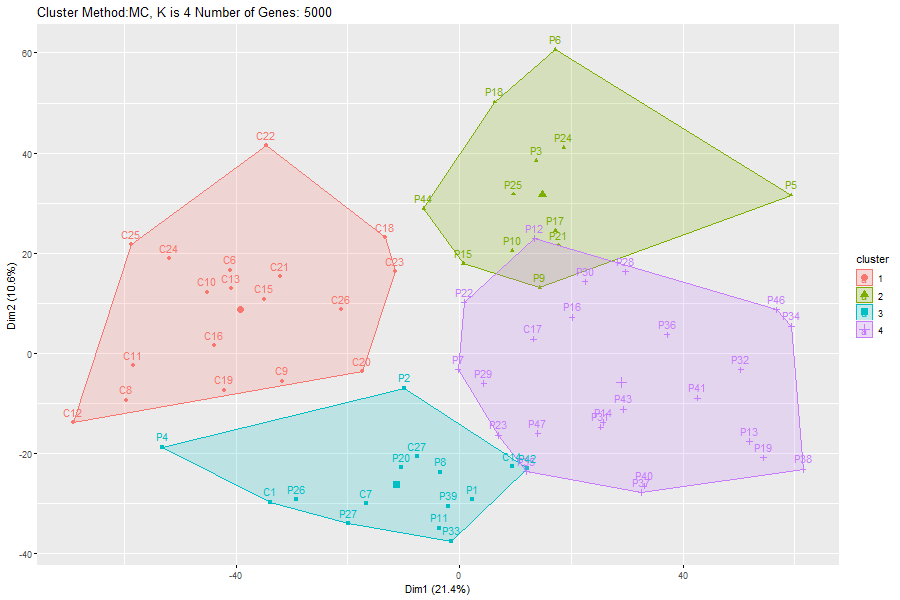
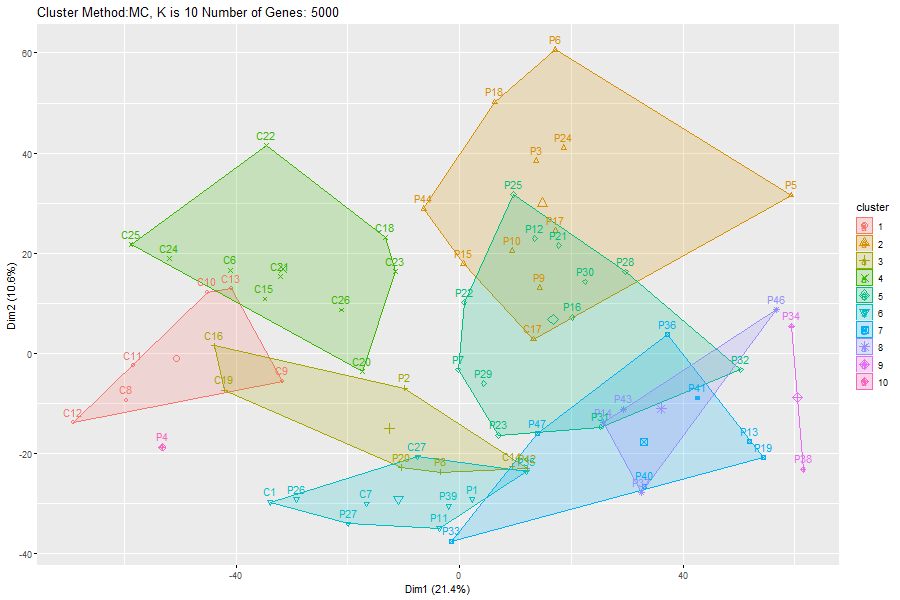
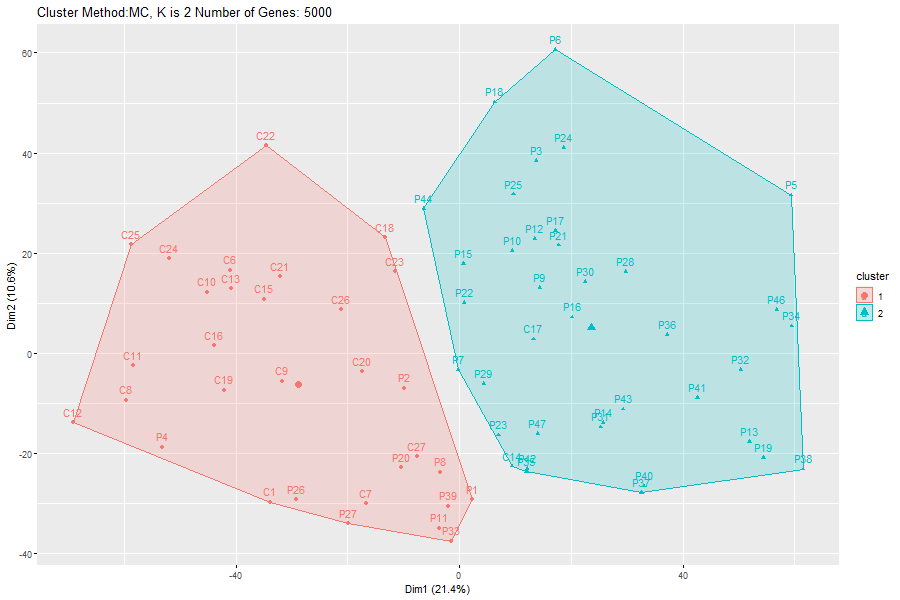
1. PAM clustering



In comparison to the HC clustering, we see that if we increase the number of clusters we give the algorithm, we generate clusters with similar sizes. However, the increase in the K given to the algorithm does not refine the clusters themselves. This is shown due to some clusters containing samples from critical and noncritical patients.

1. **Gaussian Mixture Models**

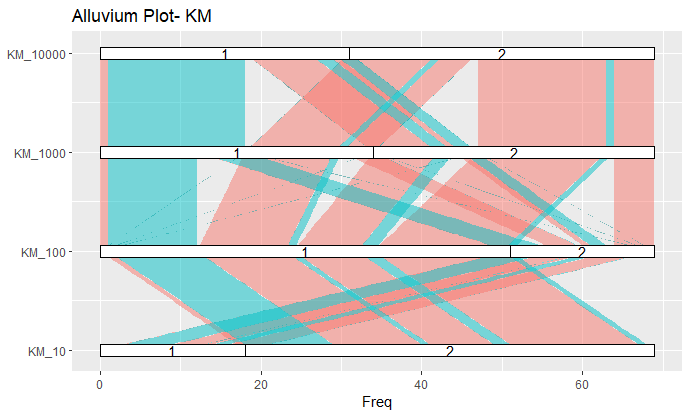
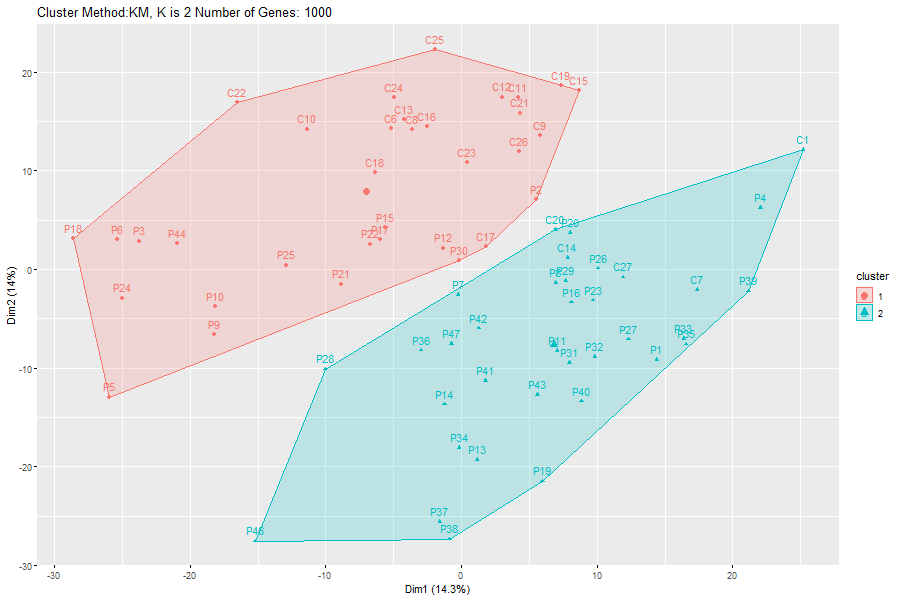
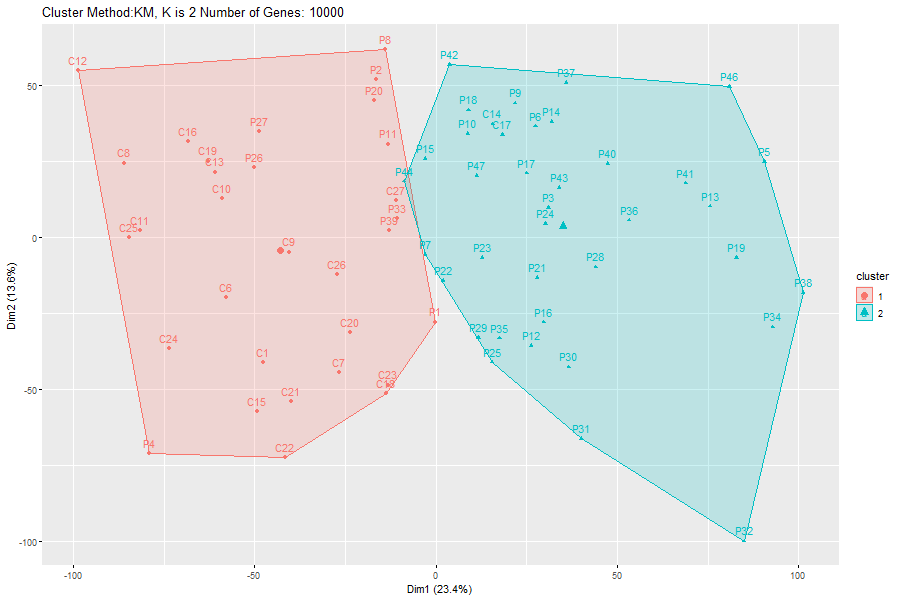
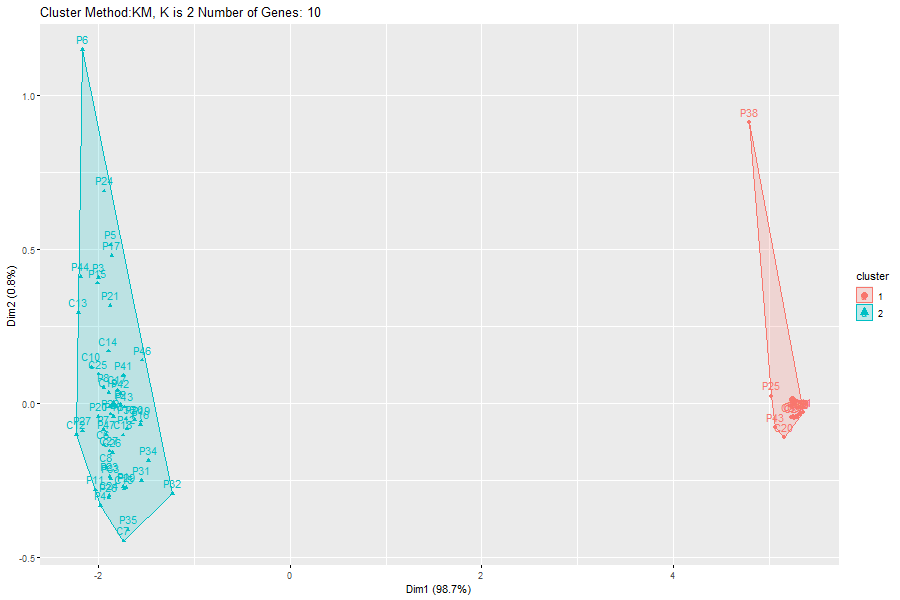
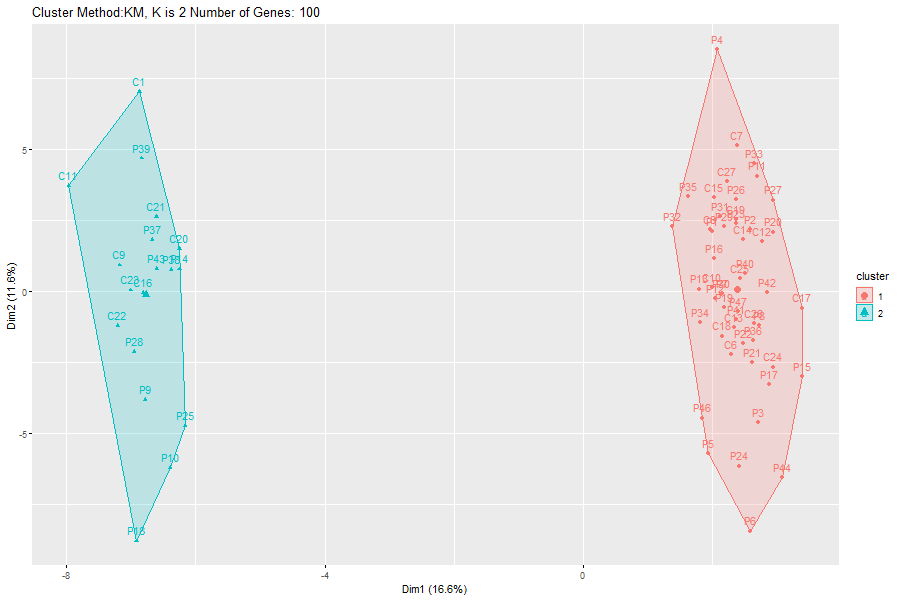
Overall, when running this modeling algorithm, we witness an improvement in the clusters as we increase the K value resulting in the clusters becoming more refined, which allows our graph to better separate the samples. However, even with k= 10, the graph still contains some degree of error due to some clusters containing noncritical and critical samples.



In the next step, we ran each algorithm with a different number of genes, using: 10, 100,1000, and 10000 genes. In all the runs, we gave the algorithm K = 2.

First, we must notice that all the clustering happens in a 2-dimensional plot, meaning that the R function had to run a dimensional reduction algorithm first. That's why when we increased the number of genes, we get different distributions of the points on the grid.

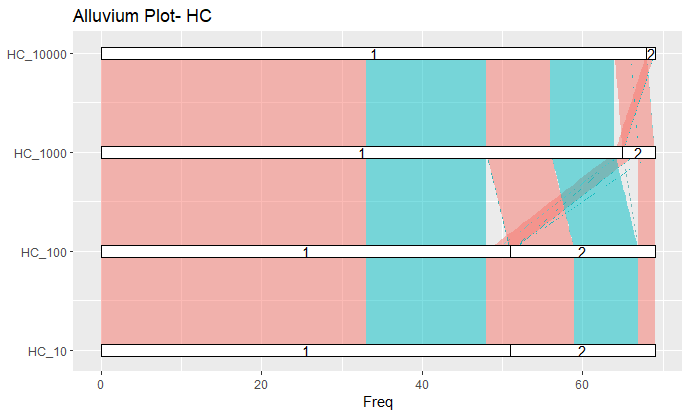
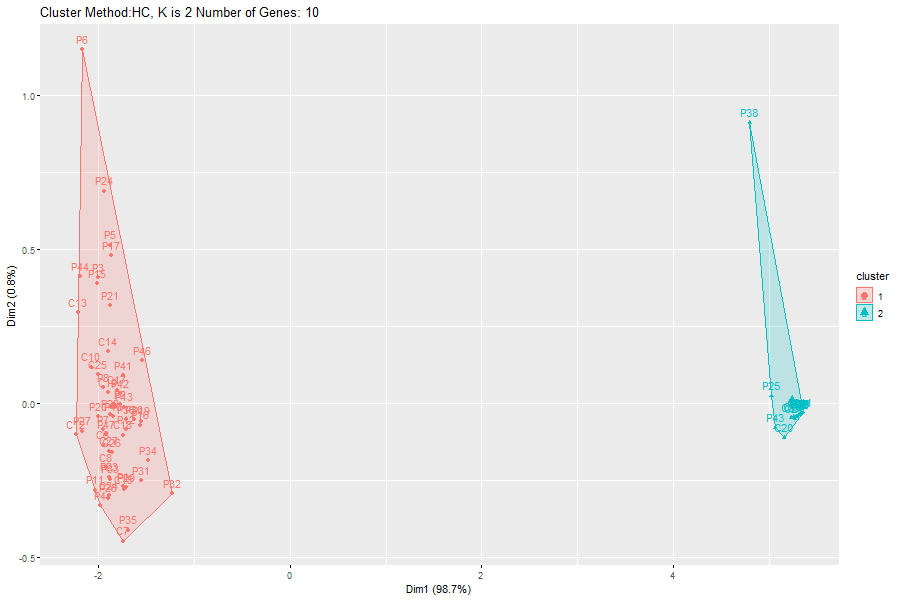
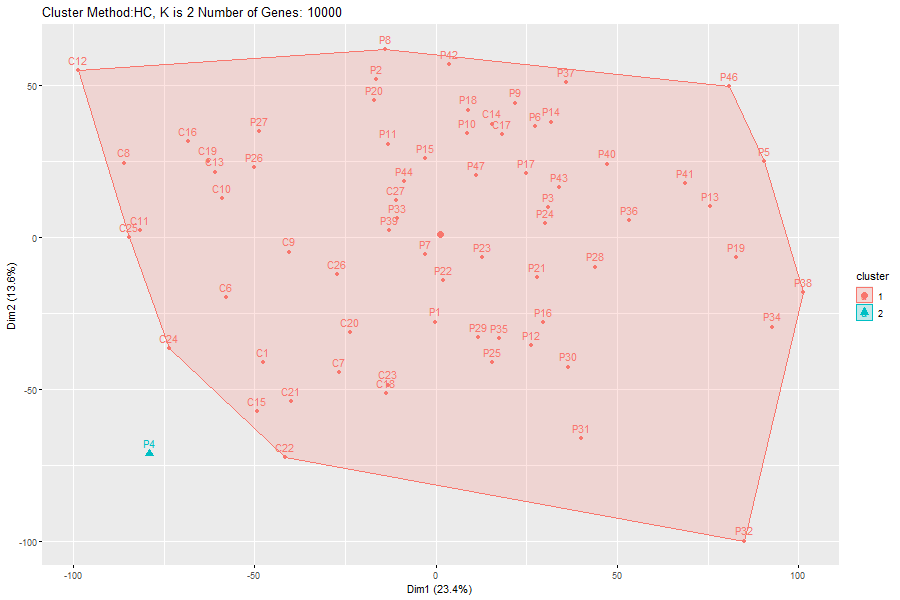
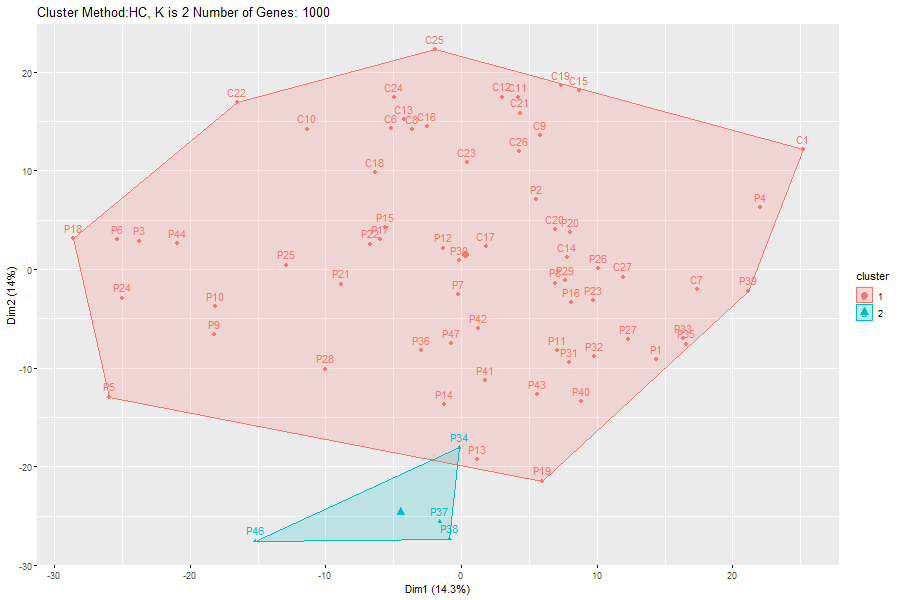
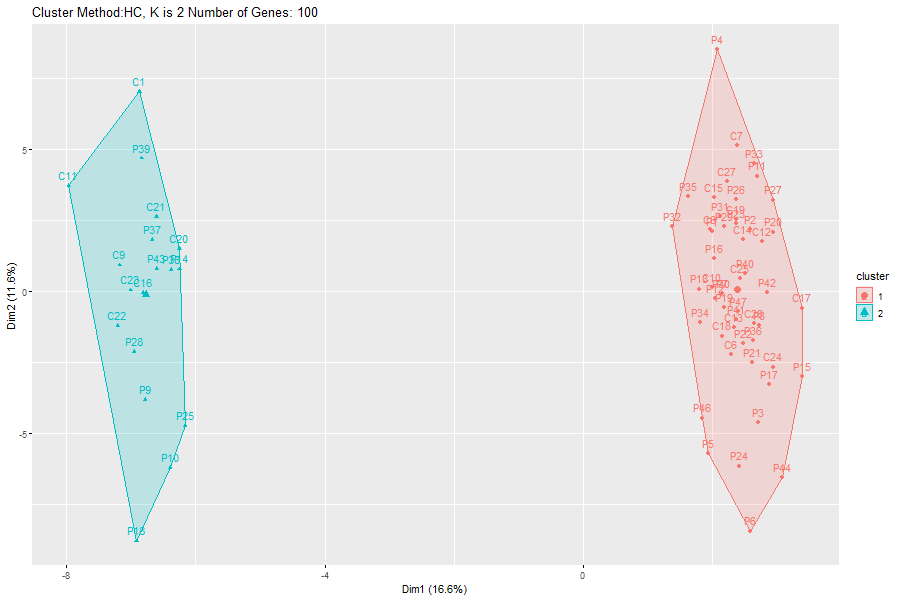
**K-Means:** We can see that as we increase the number of genes, the clusters become wider. Also, the more genes we use, the better the algorithms perform in terms of creating 2 clusters of critical and non-critical samples.



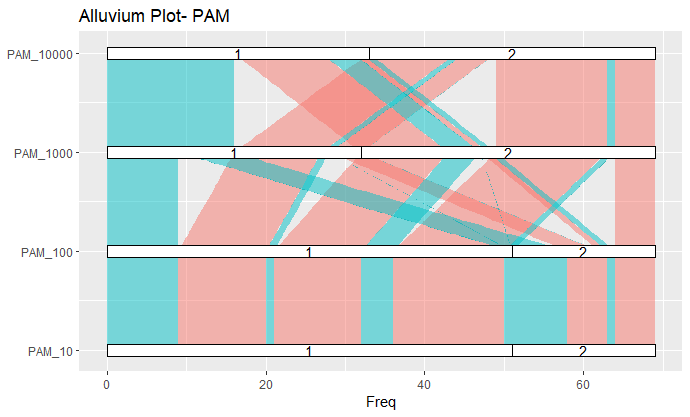
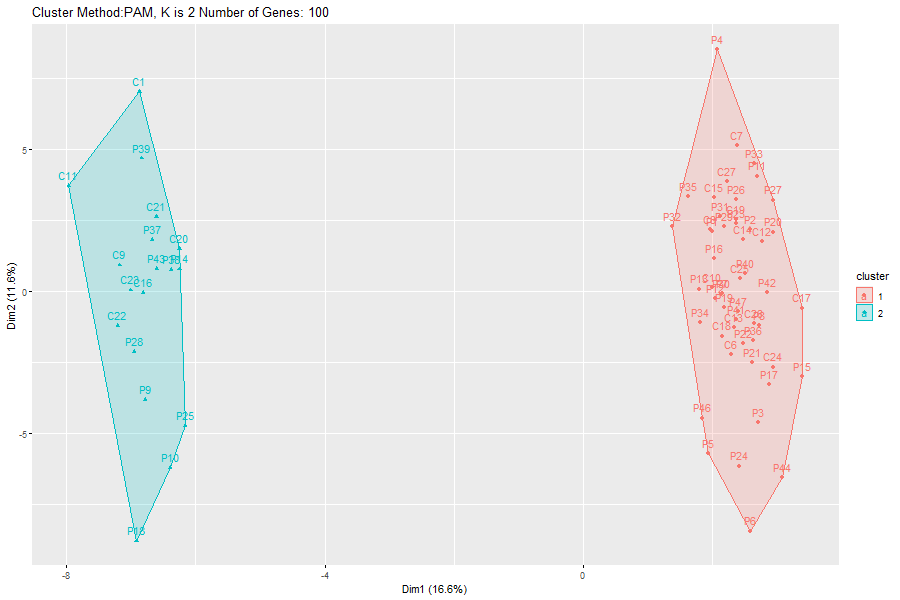
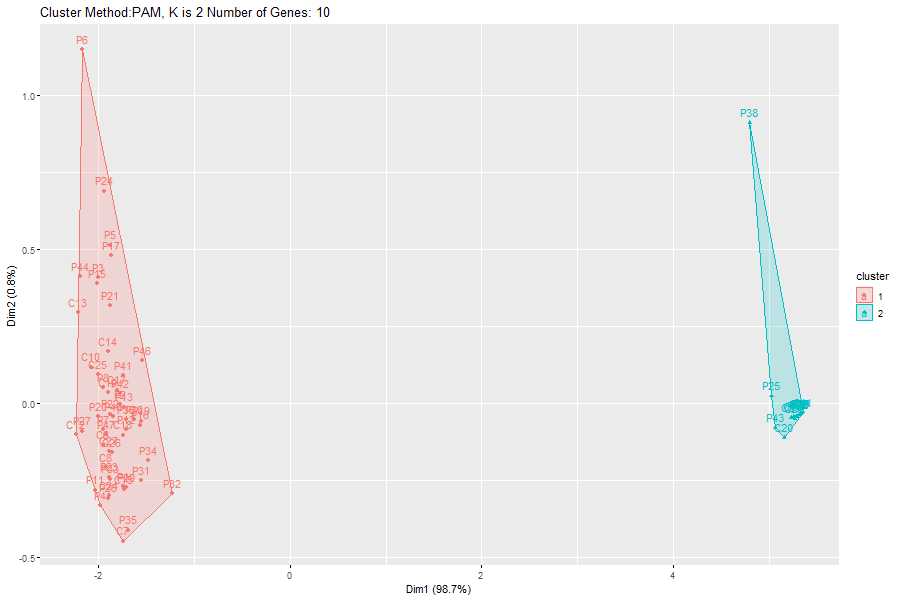
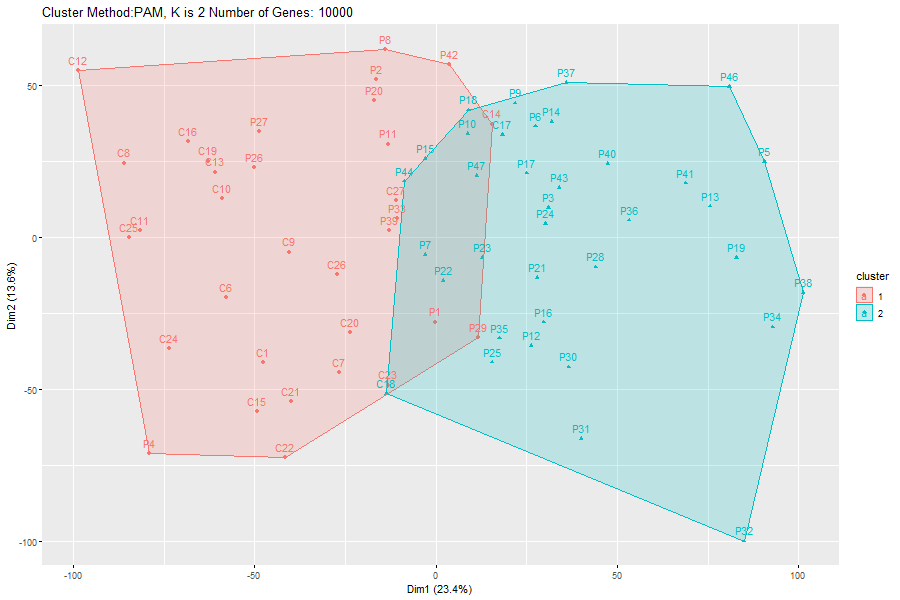
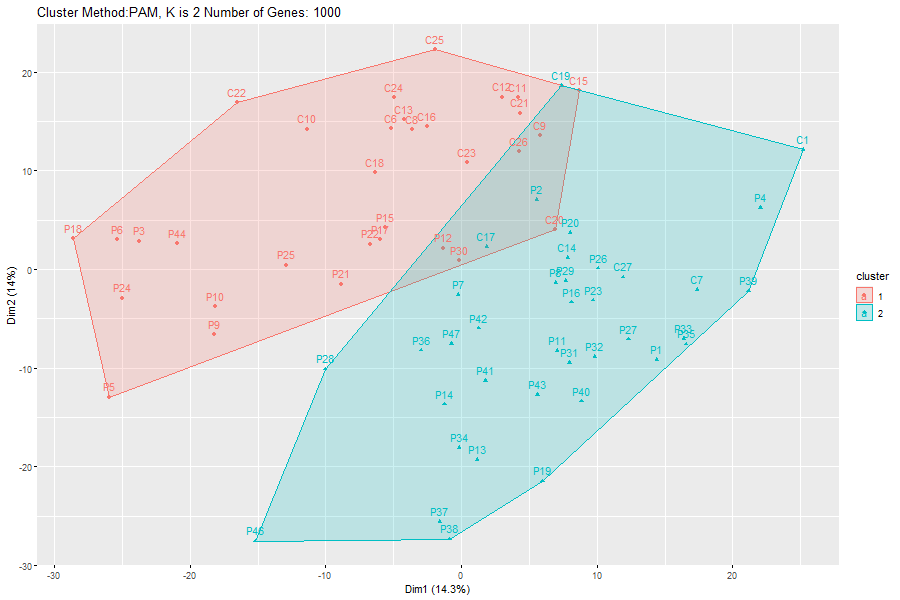
When looking at the alluvial plot, we see that with 10 and 100 genes the two clusters contain approximately the same samples. However, when using more than 100 genes, the clusters are not similar, as samples separate differently between the two clusters.

**HC:**  Here it is demonstrated again that HC does not work well with our data. In contrast to KM, when increasing the number of genes, the algorithm creates one big cluster. It seems like it can't handle many features in our data.

This also stands out when looking at the alluvial plot. One can also notice that clustering with 10 and 100 genes gives the same result.

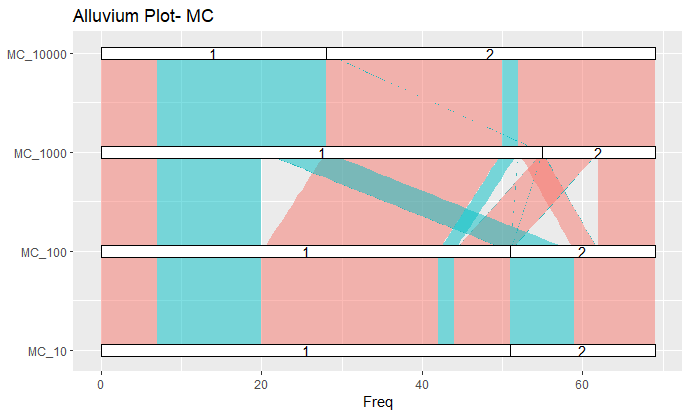
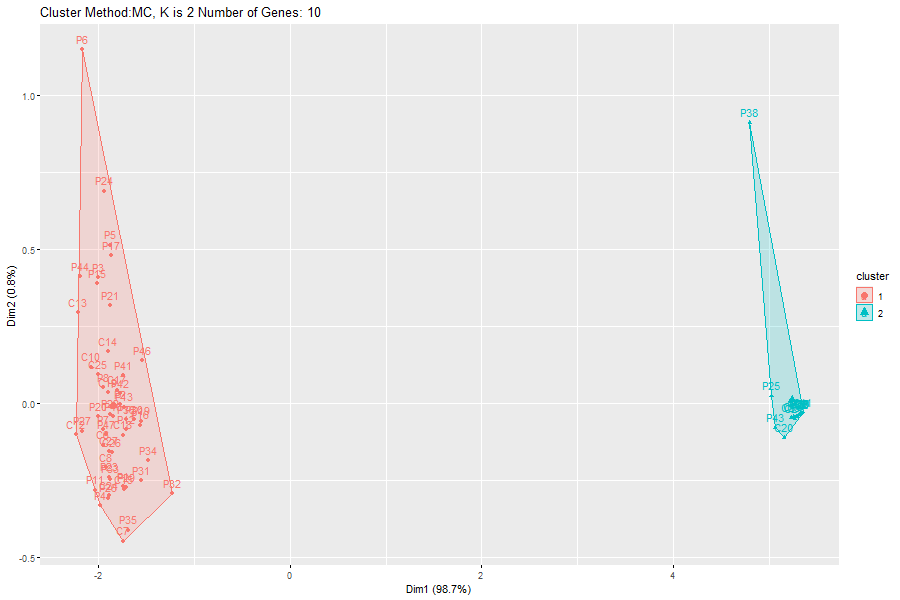
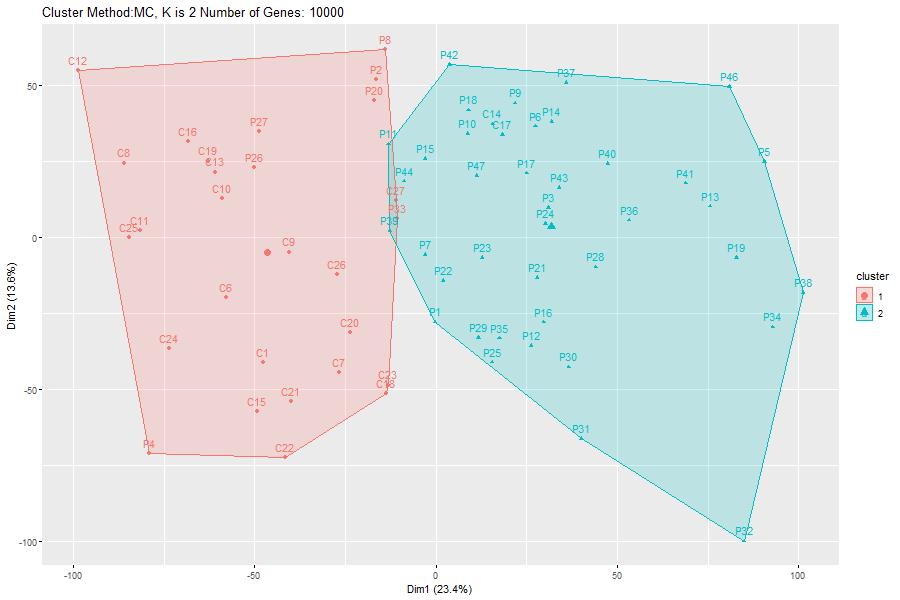
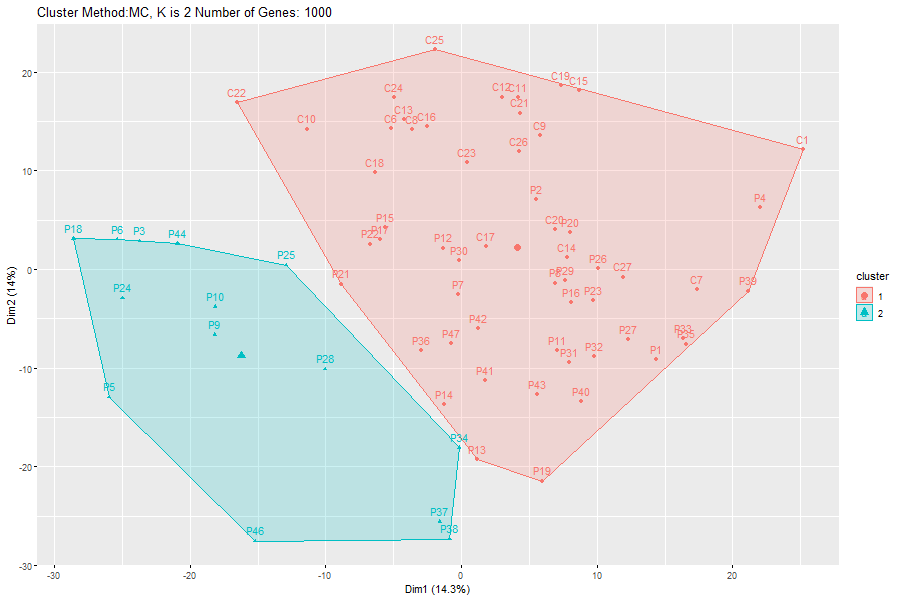
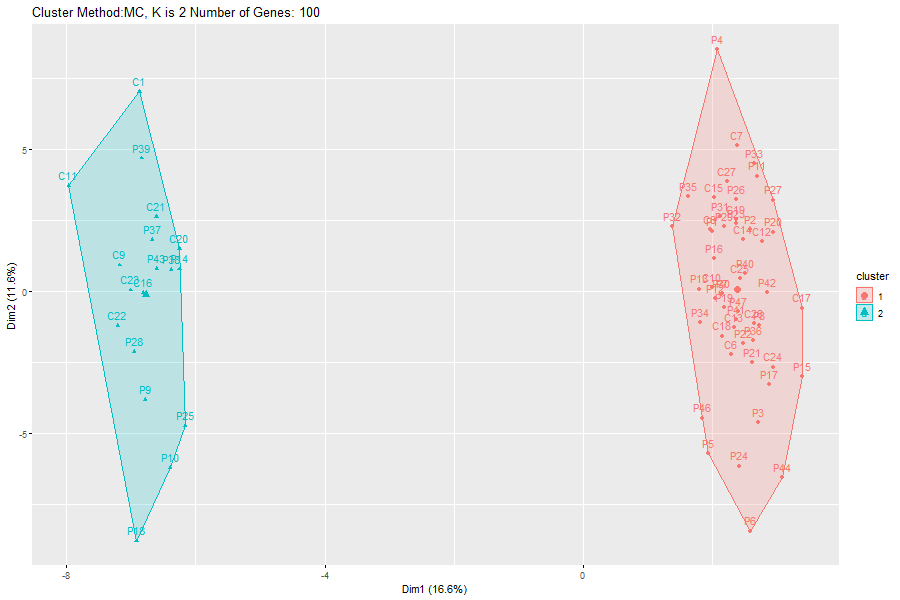


**PAM:**  By looking at the alluvial plot, we see that with 10 and 100 genes we get the same clusters. Also, we notice that for 1000 and 10000 genes, we notice a decrease in the accuracy of the groups do to overlaps between sample groups.



**Gaussian Mixture Models:** By looking at the alluvial plot, we can see that the clusters created using 10 and 100 genes are the same and that the difference between the clusters created using 1000 and 10000 genes is that many samples moved from cluster number 1 to cluster number 2. The algorithm shows that as the number of samples provided increases, the accuracy of the graph improves too.

**Heatmaps and Dendrograms**



We created a Heatmap using the 5000 most varied genes, we chose k=4.

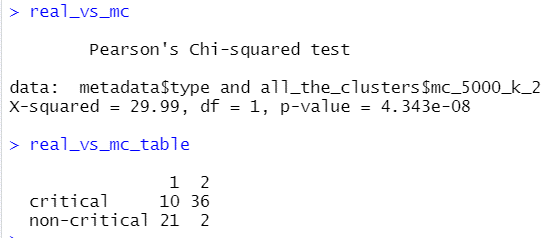
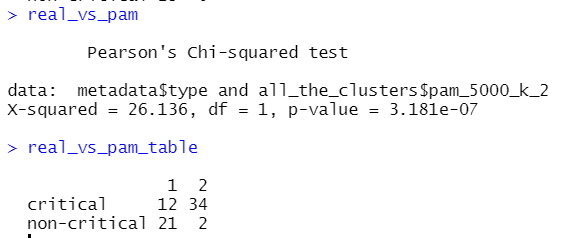
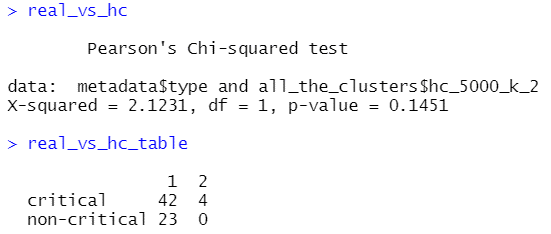
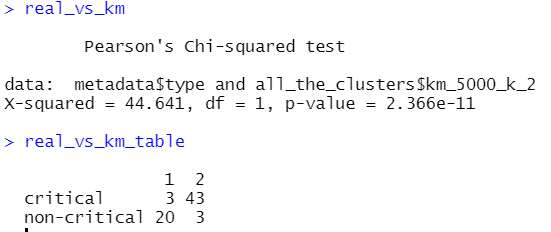
A screenshot of a computer

Description automatically generated with low confidenceIn the graph, we can see the heatmap of the genes by the samples and for each sample, we can see the cluster it belongs in each algorithm.

**Statistics**

1. First, we compared the clusters of each algorithm to our real 2 groups (critical and non-critical patients).

We ran the chi-squared test of each “cluster-real” pair, these are the results:



One can notice that all the algorithms, but hclust, have a very low p-value. That strengthens our earlier thoughts about hclust, as we saw it didn’t work well with our data and created irrelevant clusters. We see that its result could happen by chance.

In addition, one can look at the X-squared results and see that KM has the largest score, then MC and PAM, and the smallest one is HC. It fits the tables that show that KM indeed has the closest clusters to the real groups and we can infer that the KM algorithm result and the patients’ conditions are dependent and we can see that the hclust algorithm result and the patients’ conditions are more likely to be independent.

(The bigger the X-squared score the less likely the variables are independent)

1. (+c+d) We took the real groups (critical and non-critical patients), every result of the following algorithms: K-Means, hclust, PAM, and Gaussian Mixture Models, with each of these numbers of genes: 10, 100, 1000, 10000, with k=2 and made all the possible pairs to perform the chi-squared test (210 pairs).

Table

Description automatically generatedThis is the head of the table:

We calculated the p-value and the X-squared score with the chi-squared test and then calculated the adjusted p-value.

1. Chart

   Description automatically generatedWe entered the table into the “ggpair” command and got the enrichment analysis.