Project Report on
“Clustering Performance on gene expression data”

Summary of research questions and results:

1. Validate the results clustering using PCA on the gene expression data

In cases where we have a large amount of data, PCA can be useful for dimension reduction. The principal components obtained subsequently help in generating meaningful insights. Clustering is useful for cases where we have unlabelled data, and it helps in observing patterns in data. My objective was to visually compare the two techniques and validate them against each other.

Motivation and background:

While experimenting on a huge dataset containing the gene expression values of hundreds of genes, we can use PCA to reduce the dimension and then proceed with further analysis or use clustering to observe an underlying relationship between the different genes. These methods are generally used for any kind of data that has a large number of features.

The motivation of this project is to validate the results of one technique with another i.e. validate the results clustering using PCA

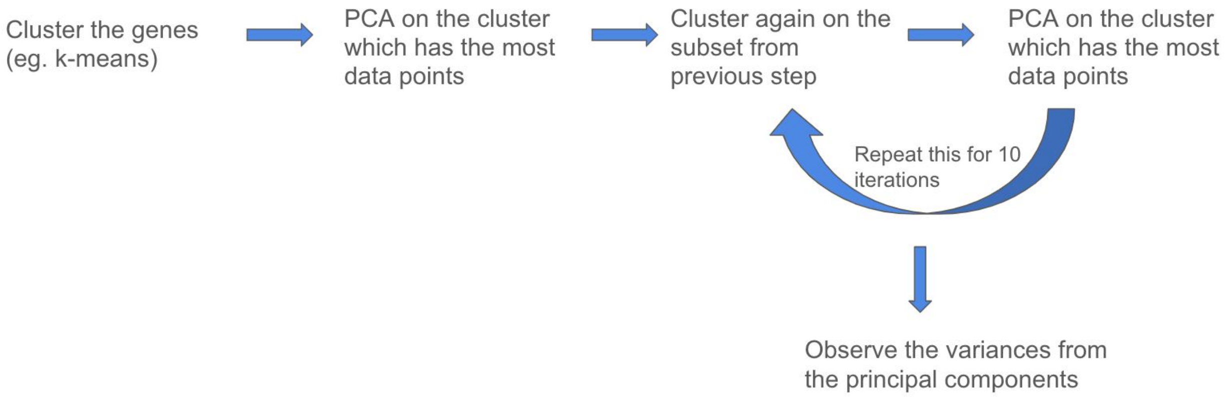
Dataset:

The dataset used is the PANCAN dataset made available via SYNAPSE. This is a publically available dataset but the user needs to register via their email address before downloading. I have made a connection from the code to the URL using my personal credentials which I have added to the code and I give the permission to use for the project purpose, but the user can use their personal id/password if there exists one.
URL- https://www.synapse.org/#!Synapse:syn4303551

This collection of data is part of the RNA-Seq (HiSeq) PANCAN data set, it is a random extraction of gene expressions of patients having different types of tumor: BRCA, KIRC, COAD, LUAD, and PRAD.
Samples (instances) are stored row-wise. Variables (attributes) of each sample are RNA-Seq gene expression levels measured by illumina HiSeq platform. There are 820 samples and 20, 502 genes.

Methodology:

Overview of the steps followed:


Cluster the genes (eg. k-means)
PCA on the cluster which has the most data points
Cluster again on the subset from previous step
PCA on the cluster which has the most data points
Repeat this for 10 iterations
Observe the variances from the principal components

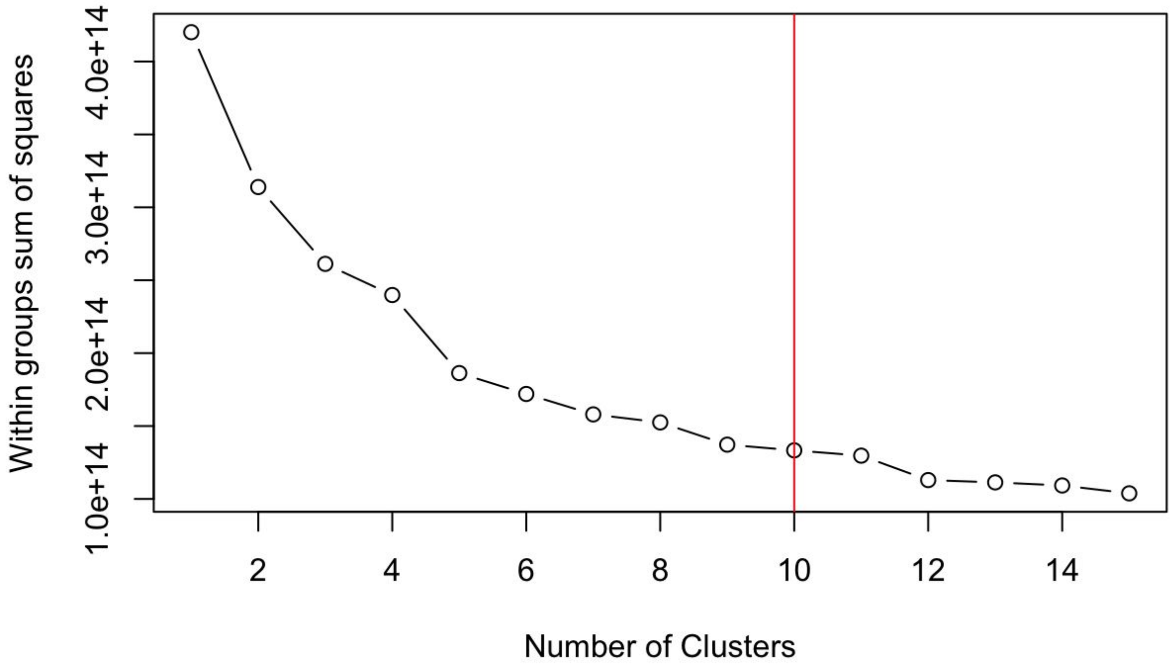
1. Read the data from https://www.synapse.org/#!Synapse:syn4303551 by either downloading or directly from the url
2. Data Manipulation
    a. Drop the first 29 rows because gene id's are not legible
    b. Transpose the remaining data, because ~20,000 genes are currently in rows and the columns are the experiments. We want the data such that each row represents a row and the columns represent the genes
3. Apply k-means clustering on the data and subset the data, keeping only the cluster that has the maximum data points
4. Apply PCA on the subset data and plot the graph showing the percentage variation by 10 principal components
5. Repeat steps 3 and 4 ten times and plot the 10 graphs side by side obtained in step 4

Results

I tried hierarchical clustering on the data but faced runtime issues (due to a large number of features in the data). I also tried density-based clustering but the separation was similar to k-means clustering, so decided to continue with k-means clustering.

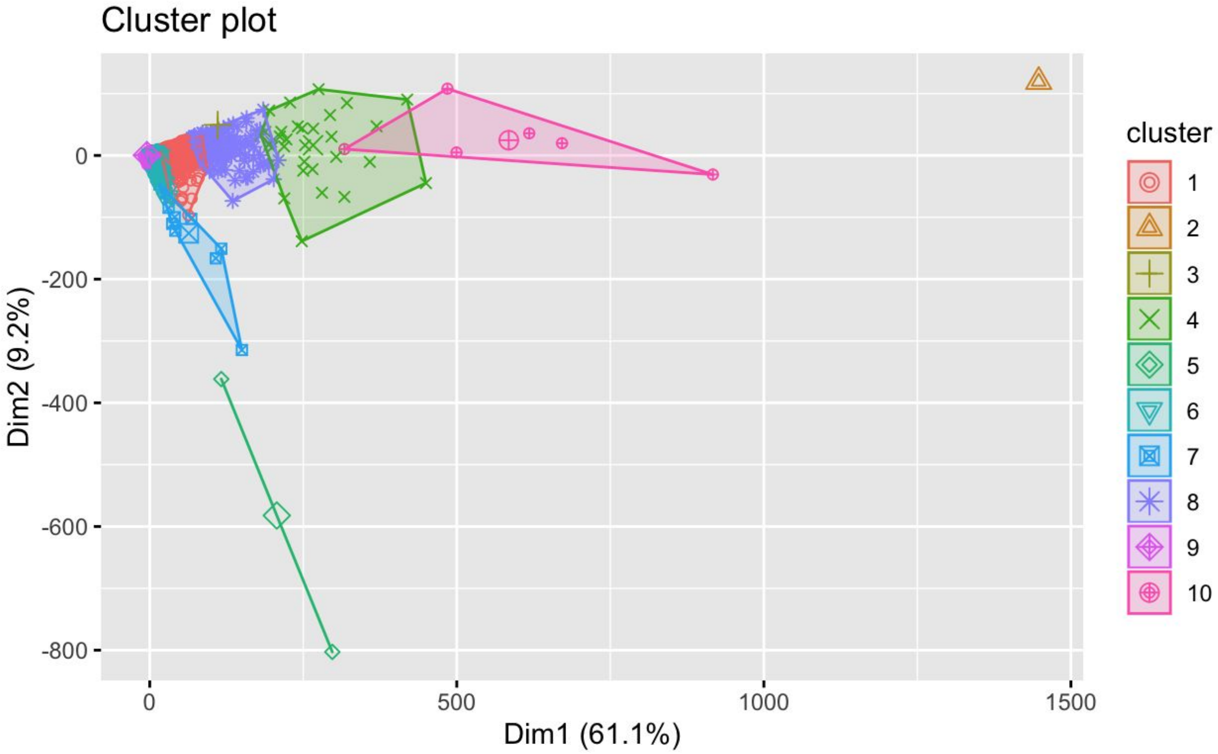
Elbow curve-

In order to determine the number of optimum clusters in k-means clustering, I used the elbow curve to decide the cutoff as shown below:

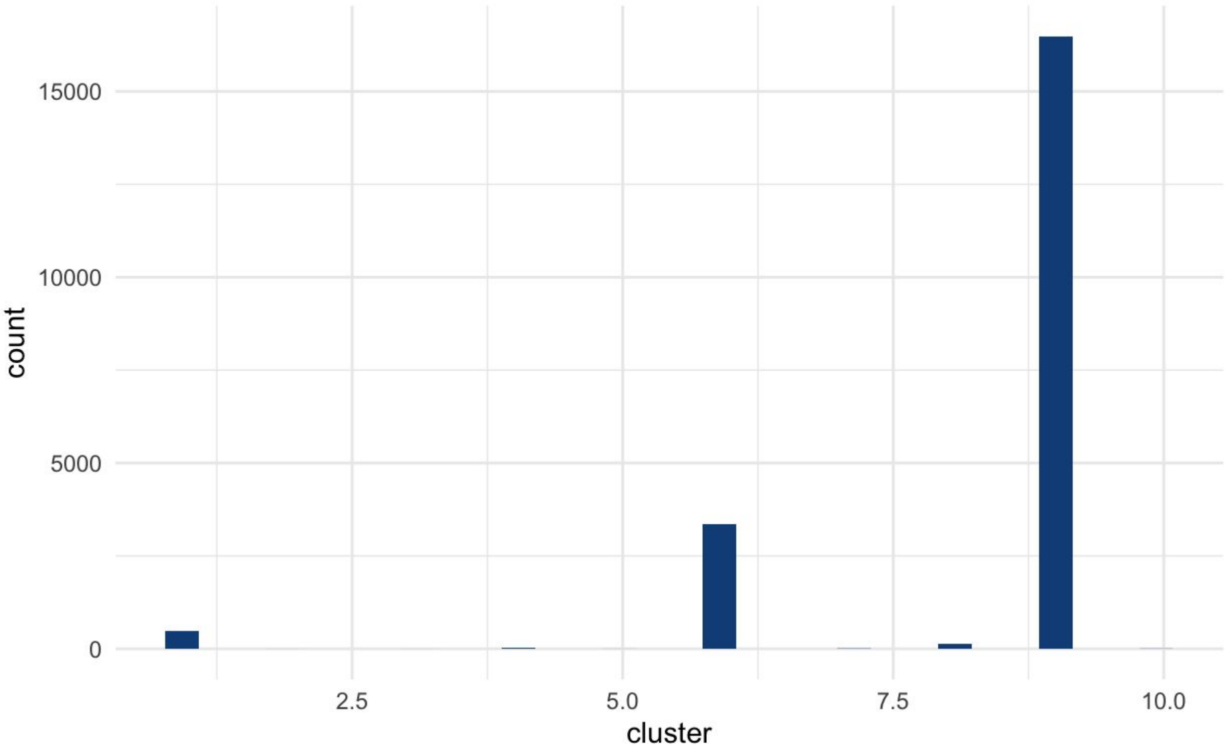


It can be observed from the plot that the WSS is very high if we segment the data in a few clusters. It could be possibly due to the fact that the data has ~20k genes and they would effectively need to be clustered into more than 2k groups (but I couldn’t test this due to runtime issues).

K-means clustering with clusters=10:
In order to continue with the k-means clustering, I chose the number of clusters = 10 based on the elbow curve(cutoff is decided where there is a drop in the error) and obtained the following separation:

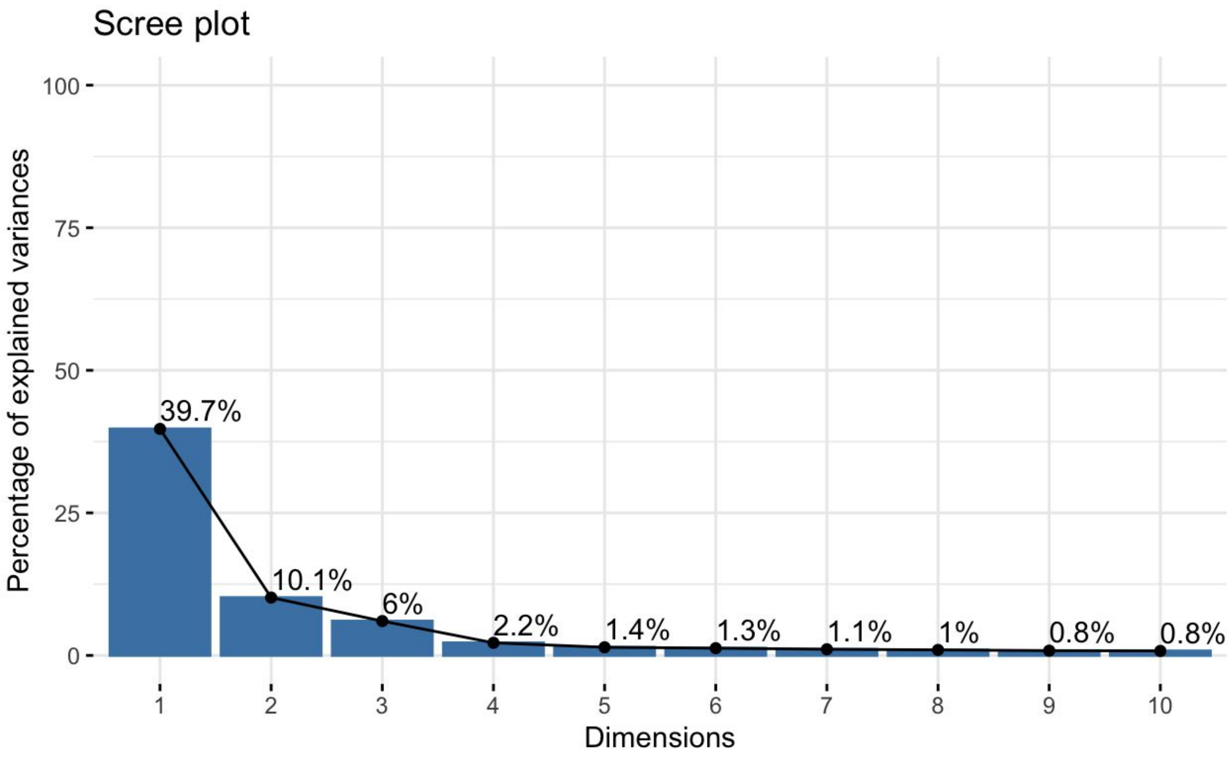


Histogram showing the counts in each cluster(total clusters= 10) is shown below:



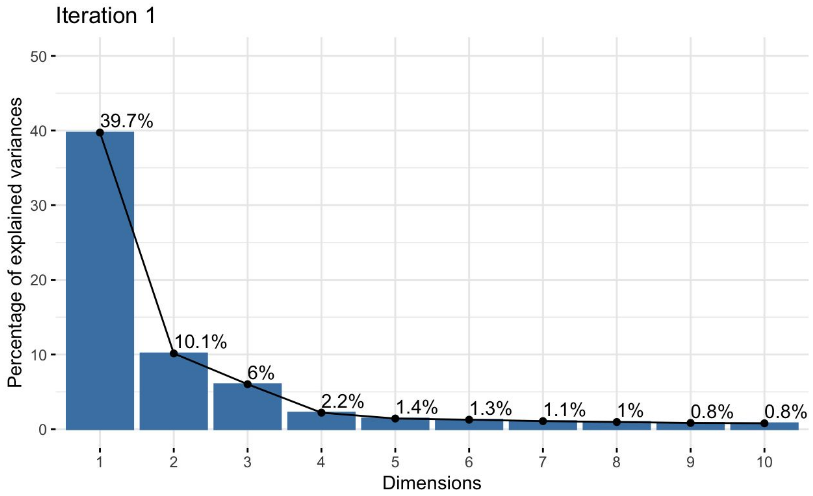
It can be observed from the graphs above that cluster 9 has the highest number of data points(~16k), and all the other clusters are closeby, except cluster 5 and cluster 2 (which seem like an outlier)
It can be observed that just using k-means clustering on such a data is not useful and we can't make any meaningful interpretation. In such cases, PCA helps to reduce the data into a few principal components which explain the maximum variation in the data.

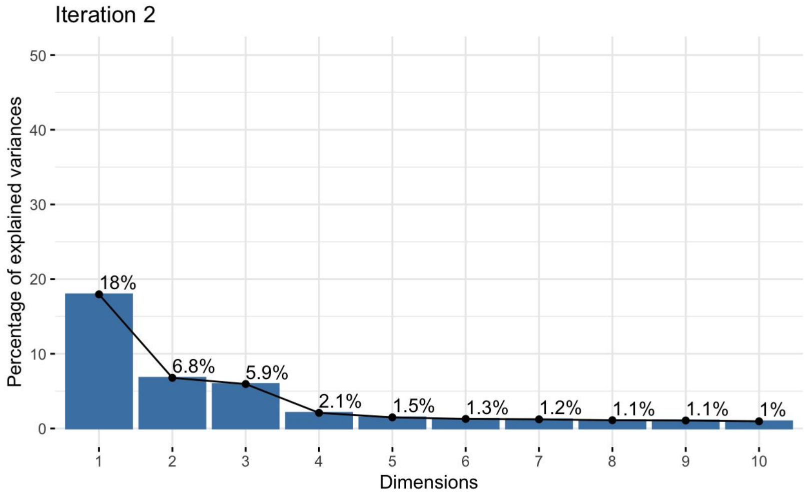
PCA:
Applying PCA on the entire dataset will give the plot below and it can be seen that the first two principal components explain ~70% variation in the data:

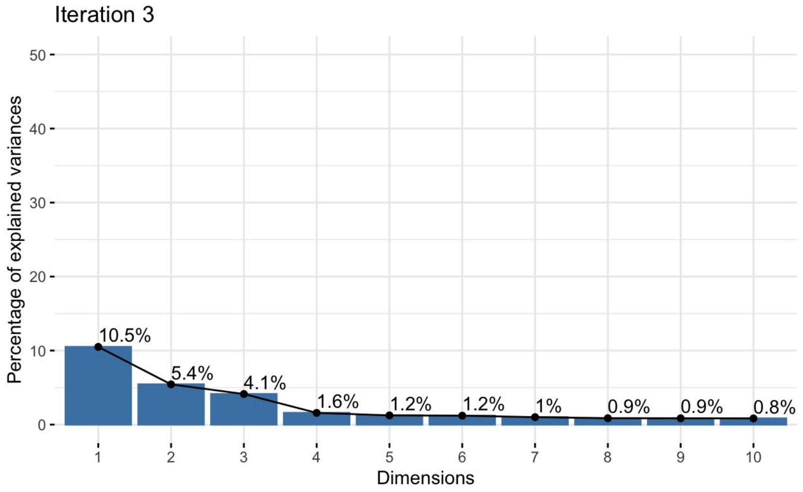


Scree plots obtained for the 10 iterations:
The following graphs show that variation explained the top 10 principal components of the data keeps on decreasing when we apply PCA on the data that has the most similar data points i.e. they belong to the same cluster obtained from k-means clustering

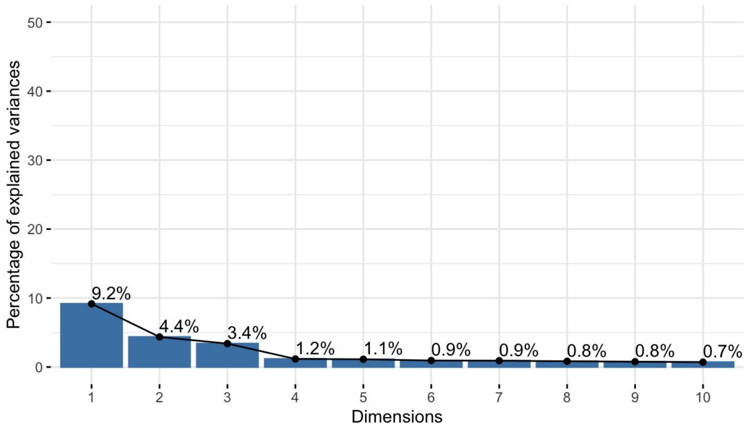
One of the possible explanations could be that since more similar data points are clustered together when we try reducing the dimension those data points using PCA, the top 10 principal components are not able to explain a lot of variation in the data, which would mean that we need to take all those data points when performing any further analysis. From iteration 7 onwards the number of genes in the data is in the range 800-300 instead, while in the first iteration the PCA is applied on a dataset having ~16k genes


Iteration 1
Percentage of explained variances
0
10
20
30
40
50
39.7%
10.1%
6%
2.2%
1.4%
1.3%
1.1%
1%
0.8%
0.8%
1
2
3
4
5
6
7
8
9
10
Dimensions

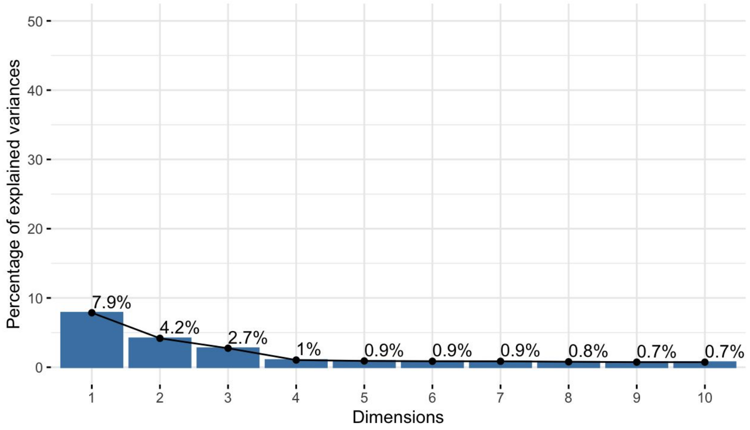

Iteration 2
Percentage of explained variances
0
10
20
30
40
50
18%
6.8%
5.9%
2.1%
1.5%
1.3%
1.2%
1.1%
1.1%
1%
1
2
3
4
5
6
7
8
9
10
Dimensions


Iteration 3
Percentage of explained variances
0
10
20
30
40
50
10.5%
5.4%
4.1%
1.6%
1.2%
1.2%
1%
0.9%
0.9%
0.8%
1
2
3
4
5
6
7
8
9
10
Dimensions

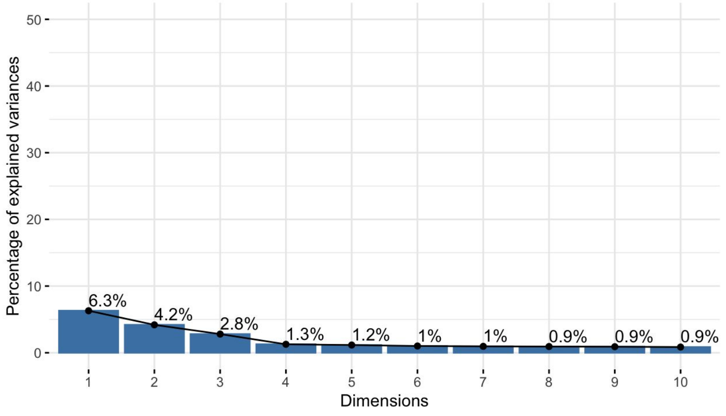
Iteration 4



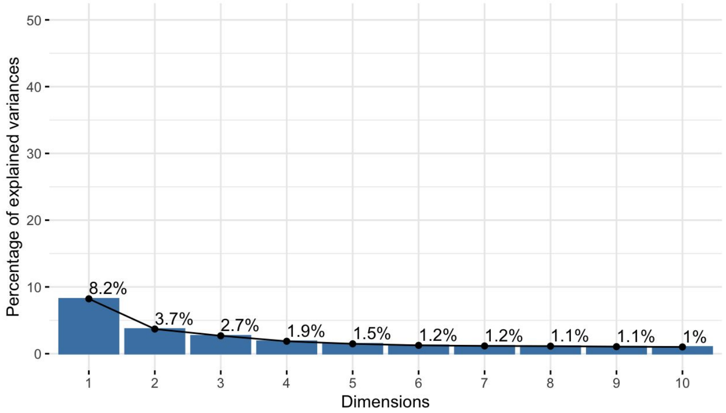
Iteration 5



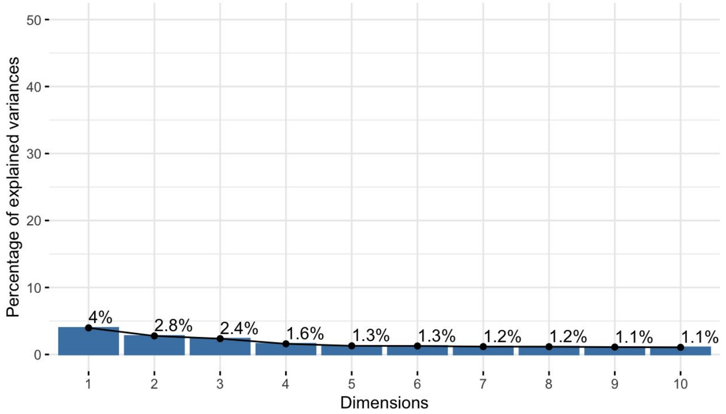
Iteration 6



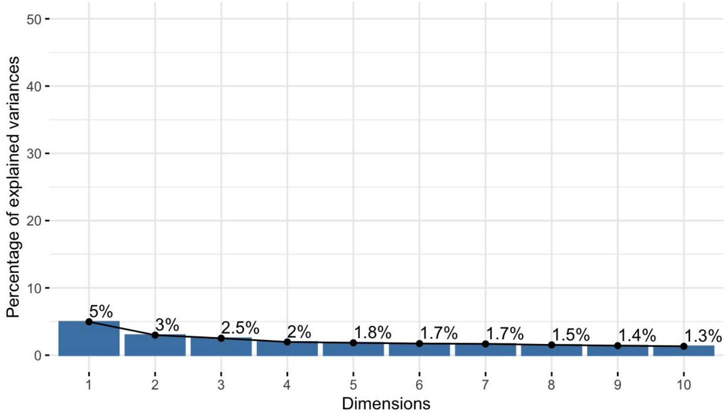
Iteration 7

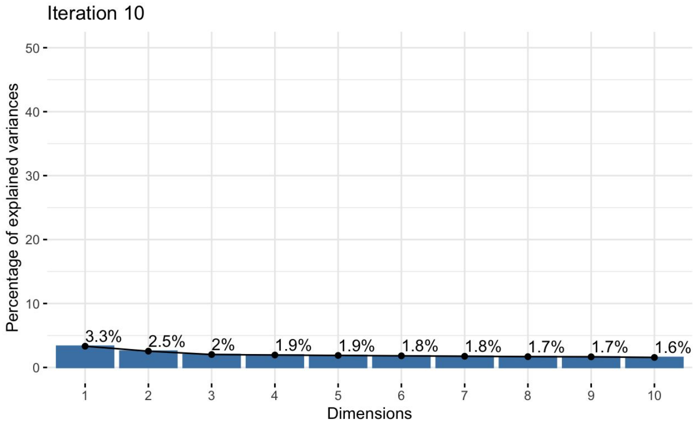


Iteration 8



Iteration 9





Reproducing the results:

I have placed the final R code my github repository here:
https://github.com/sofitiwari/Project/tree/master/code

In order to replicate the results, please install the following packages using install.packages("package-name") command in the R console (can use R-Studio):

1. "data.table"
2. "ggplot2"
3. "factoextra"
4. "dbscan"
5. "factoMineR"

Install the package synapser by typing the following in the console

1. install.packages("synapser", repos=c("http://ran.synapse.org", "http://cran.fhcrc.org"))

I have also placed a pdf document "Project_code_executed" here https://github.com/sofitiwari/Project/tree/master/code that shows the code along with the expected output. This can be used as a reference when executing the code Project.R

In order to execute the code, just run the whole R code

Reflection:

I had some difficulty framing a project proposal related to bioinformatics, but it became easier after we had a class on "Data sources" and I wish I had known that earlier.
I also had thought of comparing my clustering results with a protein-protein interaction graph, but it did not work out eventually because they are both different things(PPI graphs are built using different techniques). So my estimate was wrong here and I spent a lot of time figuring out how to do it. To summarize I would say, having a strong bioinformatics background(with respect to the different kinds of data available, knowing the common graphs and networks) will help in doing the project in lesser time because students will waste lesser time things that are difficult to implement in a time constraint which will lead to better planning of the project. I liked the project

guide that was made available to us. It was pretty detailed and helped me not miss any part or deadline.