### Centrality-Correlation Based Complex Network Similarity

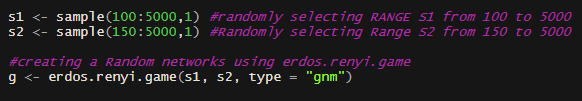
##### The Proposed Approach is the following

###### Given a complex network of size n, and a set of k centrality measures C = {C1, ,Ck}, We

compute for each centrality Ci the induced ranking vectors of the nodes σni. For each couple of centrality measures Ci, Cj we can compute the raking correlation factor cor(σni, σnj). A network Ni can then be represented as a vector in the k\*(k−1)/ 2 dimensional space. Similarity between two networks Ni, Nj can then be measured as the inverse of the distance separating both networks is the new representation space. The goal of this project is to implement the proposed network similarity measure and to evaluate this approach by using this similarity measure to cluster networks generated using different network generators.

##### Creating a Random Networks

Generating random graph networks by using a function called **erdos.renyi.game** from **igraph package**.



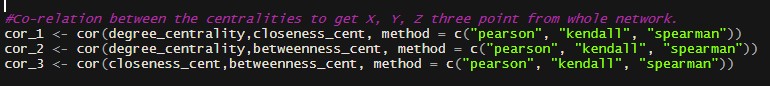
##### Calculating Centralities

##### Calculating Centralities (**Degree, Closeness, and Betweenness**) for each randomly generated graph by using **igraph package**.

###### 

**Calculating Centrality Correlation**

Calculating Centrality Correlation between the (**Degree, closeness, and Betweenness**) centralities using the function cor().



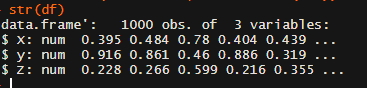
##### creating a DataFrame

Creating a DataFrame from the Points occurred from **cor\_1, cor\_2 and cor\_3**. And removing unwanted columns.

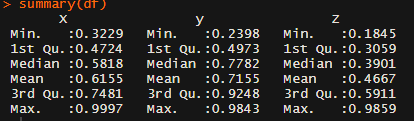
### 

##### Results of DataFrame(df)

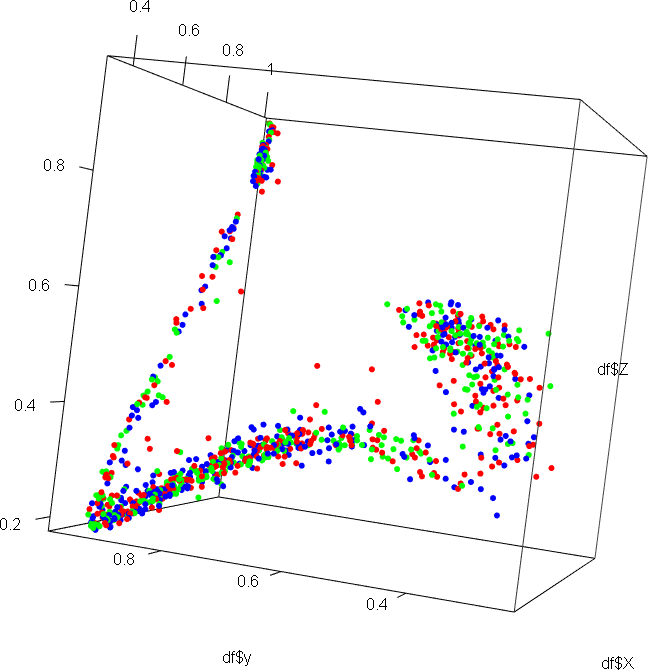
###### Here some of the results like Structure, Summary, Dimensions and Names in DataFrame.



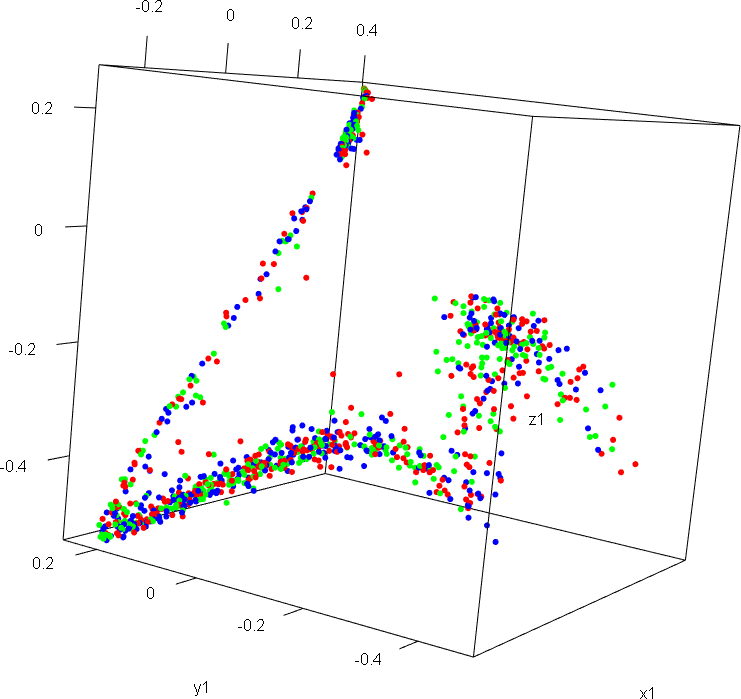


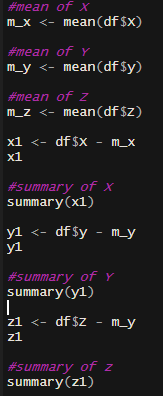


##### Plotting the DataFrame

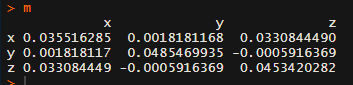
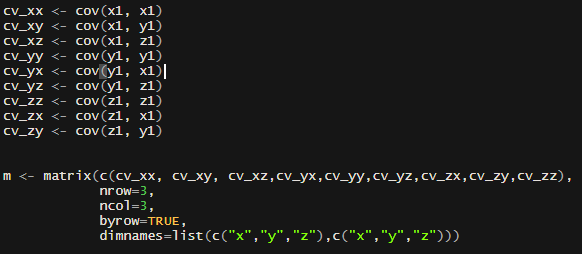


**Standardization, where the dimensions now have a mean of zero**





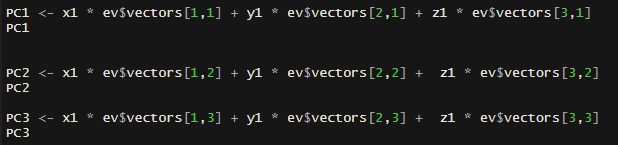
##### Covariance matrix

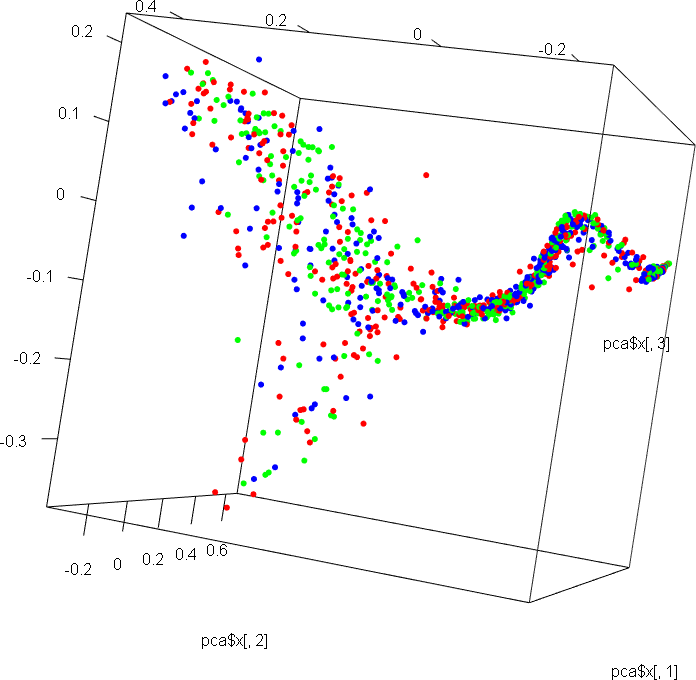
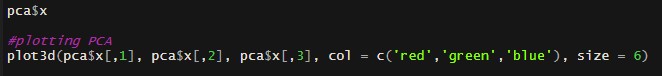
Covariance measures how dimensions vary with respect to each other and the covariance matrix contains all covariance measures between all dimensions and Names in DataFrame and eigenvalues of the covariance matrix. An eigenvector is a direction and an eigenvalue is a number that indicates how much variance is in the data in that direction.

### 

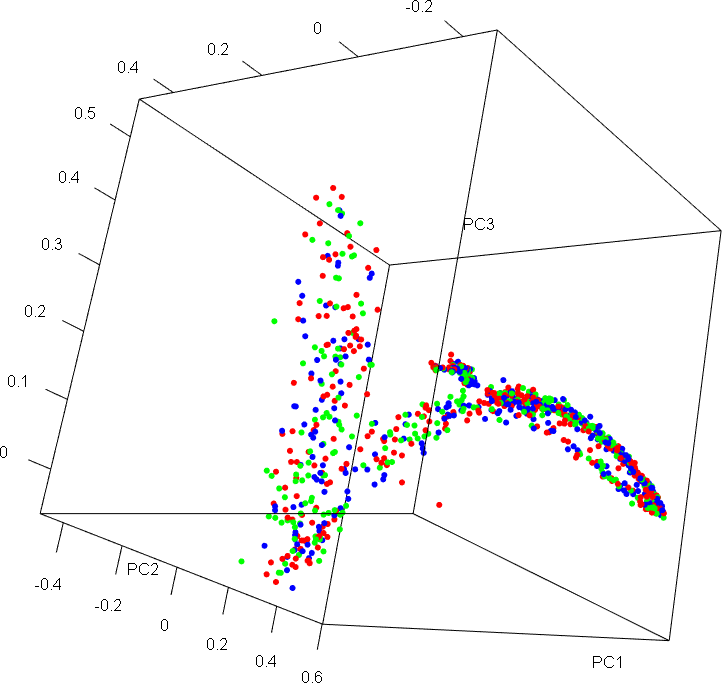
##### Principle Component Analysis

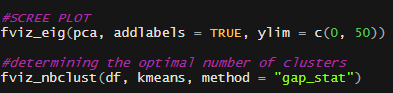
###### The largest eigenvalue is the first principal component. We multiply the standardized values to the first eigenvector, which is stored in ev$vectors[,1].

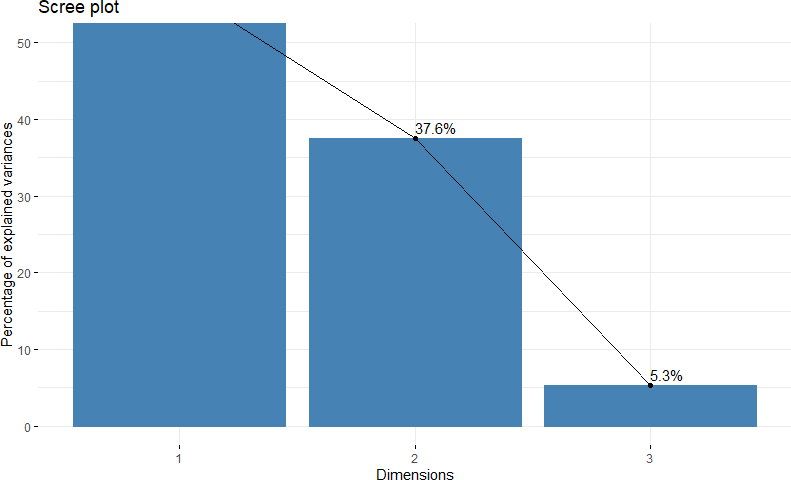
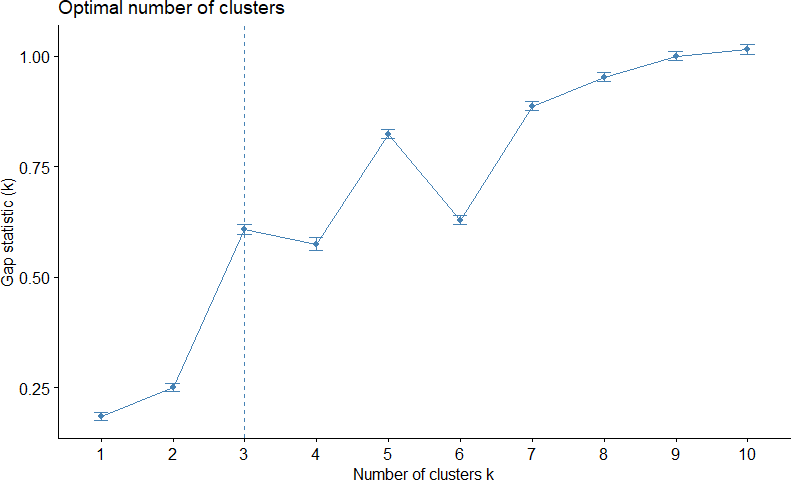




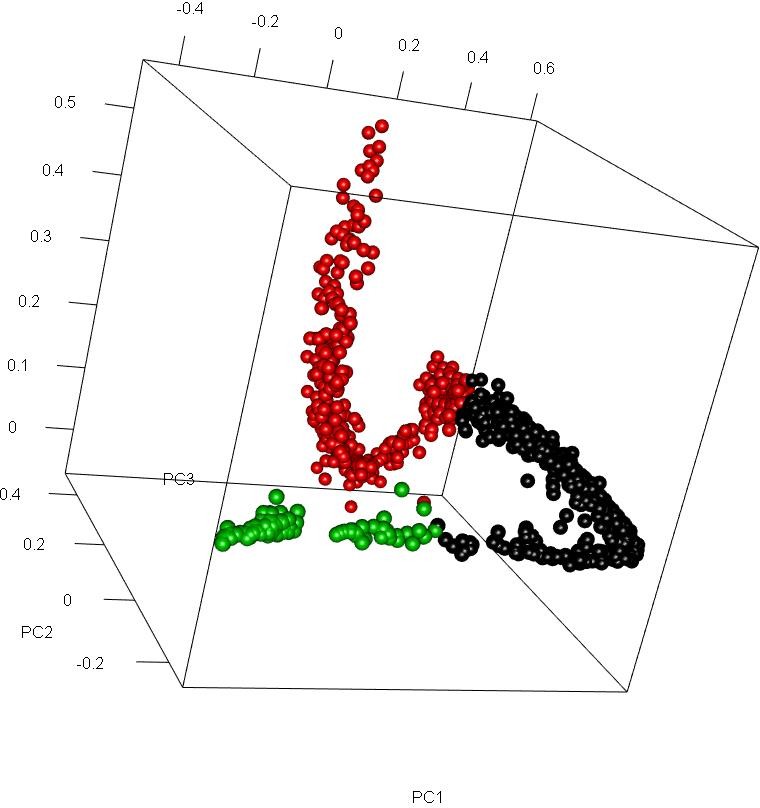


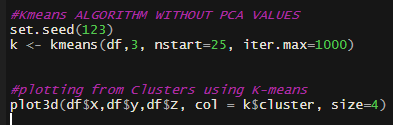
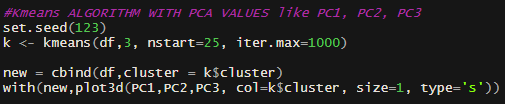


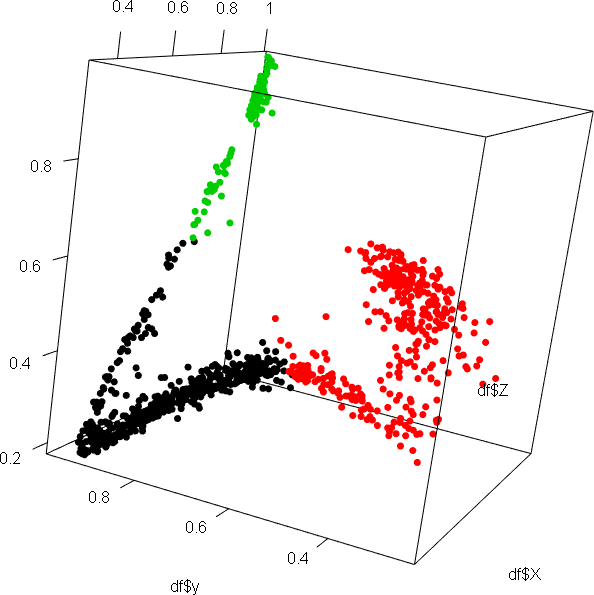


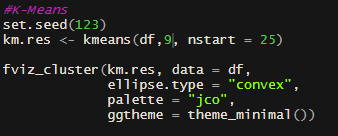
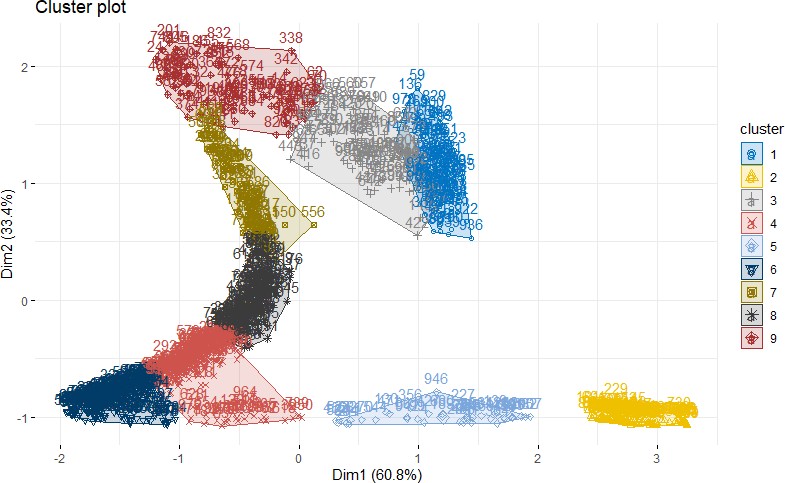
##### Clustering using K-Means Algorithm

 **K-means with PCA values K-means without PCA values**





###### 

##### Clustering using Hierarchical Algorithm

###### 

