install.packages("devtools")

library(devtools)

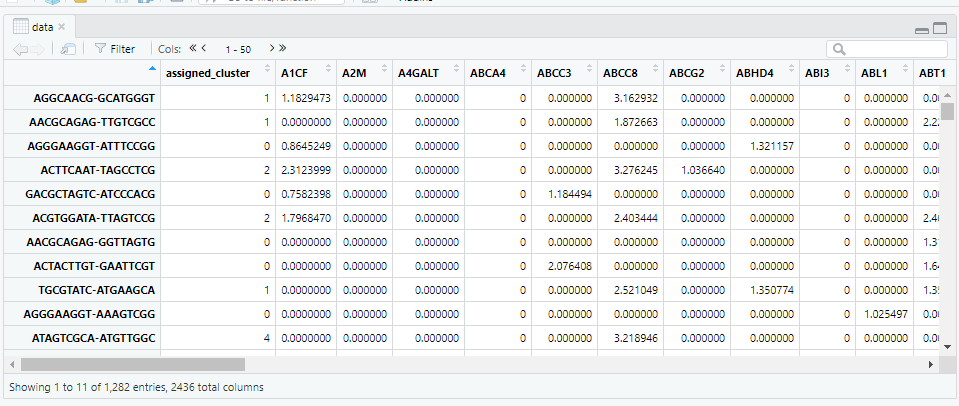
install\_github("mkarikom/RSoptSC")

library(RSoptSC)

data = read.csv("D:\\Masters\\Datasets\\Baron-Human Pancreas\\HumanD4NormWithColNamesRowNames.csv")

row.names(data) = data$X

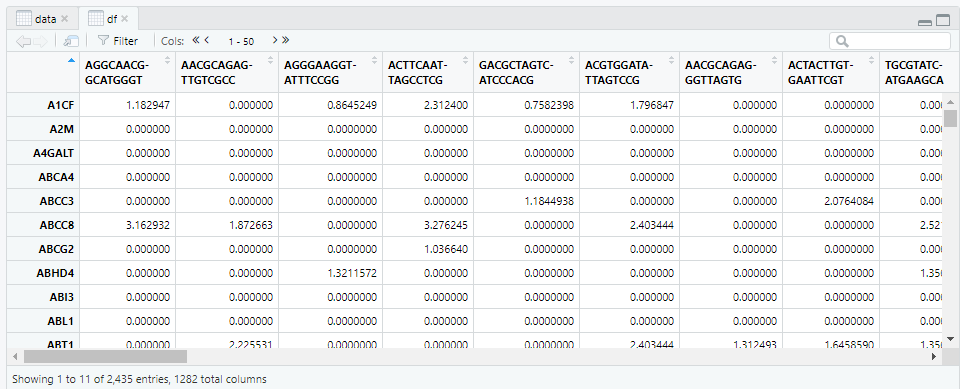
data = subset(data, select = -c(X))



df = subset(data, select = -c(assigned\_cluster))

df = abs(df)

df = t(df)



data\_m = data.matrix(df)

S <- SimilarityM(lambda = 0.05,

data = data\_m,

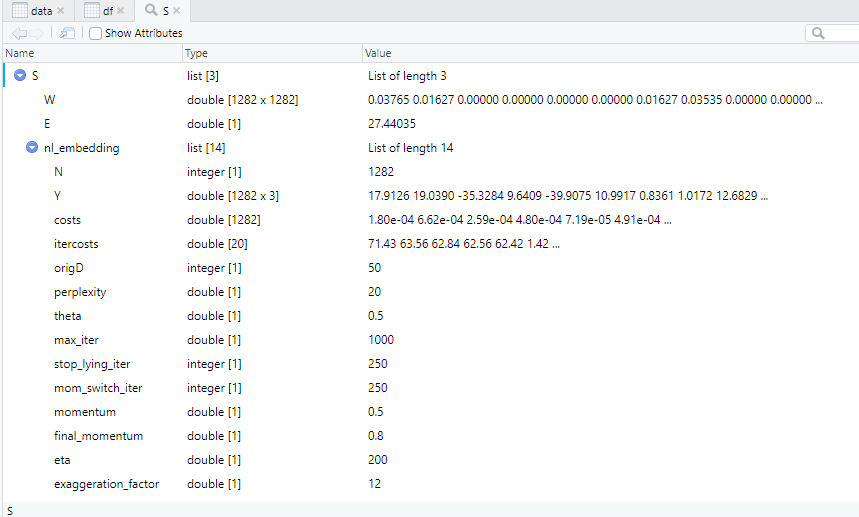
dims = 3,

pre\_embed\_method = 'tsne',

perplexity = 20,

pca\_center = TRUE,

pca\_scale = TRUE)



low\_dim\_mapping <- RepresentationMap(similarity\_matrix = S$W,

flat\_embedding\_method = 'tsne',

join\_components = TRUE,

perplexity = 35,

theta = 0.5,

normalize = FALSE,

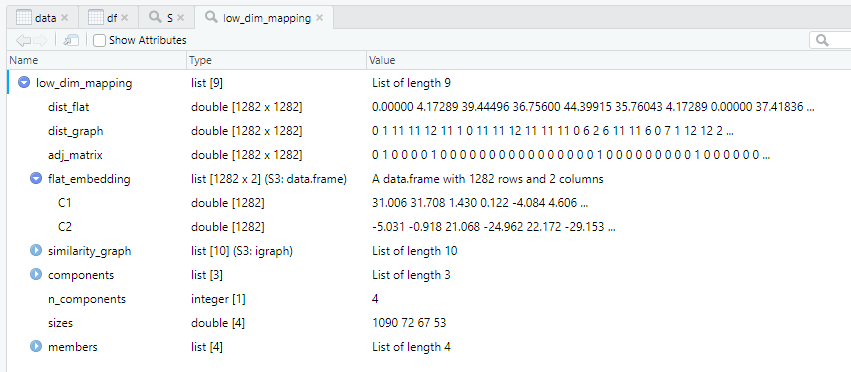
pca = TRUE,

pca\_center = TRUE,

pca\_scale = TRUE,

dims = 2,

initial\_dims = 2)

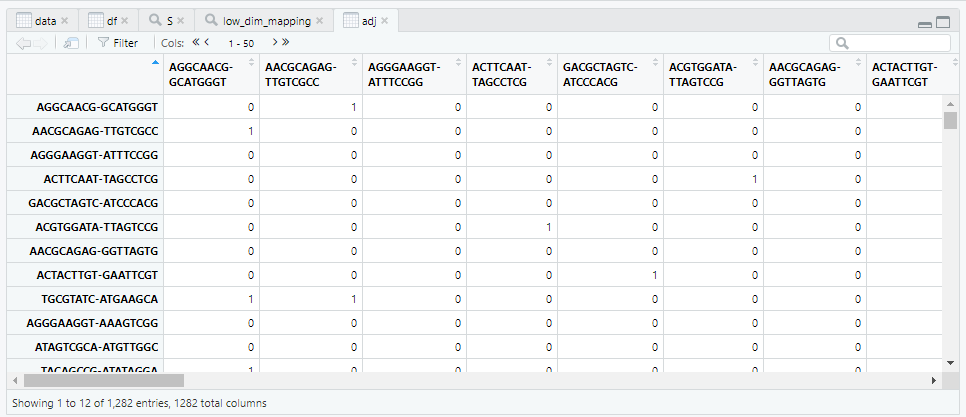


library(igraph)

adj = low\_dim\_mapping$adj\_matrix

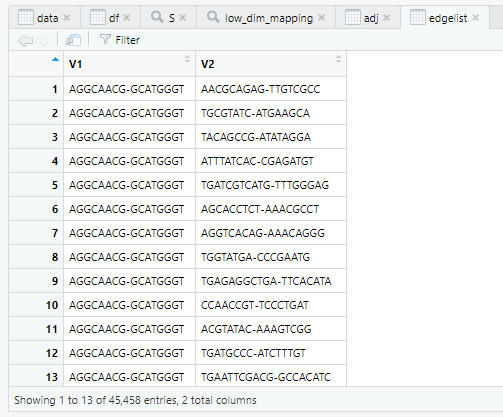
row.names(adj) = colnames(df)

colnames(adj) = colnames(df)



graph = graph\_from\_adjacency\_matrix(adj)

edgelist = get.edgelist(graph)



write.csv(edgelist,"D:\\Masters\\My\_thesis\\Code\\RSoptSC\\HumanD4\\edgelist\_HumanD4.csv")

write.csv(adj,"D:\\Masters\\My\_thesis\\Code\\RSoptSC\\HumanD4\\adjacencymatrix\_HumanD4.csv")

write.csv(colnames(adj),"D:\\Masters\\My\_thesis\\Code\\RSoptSC\\HumanD4\\cellnames\_HumanD4.csv")