Networkx_KMeans_Clustering

Jinhang Jiang

2/14/2021

```
library(fpc)
```

```
## Warning: package 'fpc' was built under R version 3.6.3
```

```
set.seed(2021)
emb_df = read.csv("embedding1.csv")
head(emb_df)
```

```
##                 X         X0          X1         X2          X3         X4
## 1 Adriene Mishler -0.1477443 -0.04417613 0.06567242 -0.08306757 -0.2942839
## 2        BadBunny -0.1633349 -0.06643627 0.05284105 -0.11405229 -0.3198857
## 3     kerwinfrost -0.1685659 -0.03580442 0.09819931 -0.12992153 -0.3602397
## 4           naeun -0.1491481 -0.05937215 0.06643727 -0.08580997 -0.3000960
## 5       BlackPink -0.1593234 -0.04215098 0.08477107 -0.09913477 -0.3184721
## 6      James Bond -0.1805906 -0.03681745 0.05972180 -0.13149384 -0.3436686
##          X5        X6         X7        X8         X9         X10        X11
## 1 0.3429810 0.1849829 -0.2376676 0.2235412 0.08185298 -0.09295850 0.08174172
## 2 0.3597758 0.1732010 -0.2566546 0.2056699 0.07970025 -0.07796029 0.09566118
## 3 0.3463729 0.1887535 -0.2520032 0.1931427 0.09303840 -0.11016743 0.07104705
## 4 0.3548943 0.1691669 -0.2668381 0.2401966 0.08348589 -0.09188239 0.09916308
## 5 0.3627763 0.1765871 -0.2738244 0.2385865 0.09703717 -0.07636267 0.11553974
## 6 0.3640858 0.1609258 -0.2494683 0.1925222 0.10372300 -0.09347132 0.09355957
##         X12         X13         X14        X15        X16          X17       X18
## 1 0.1214787 -0.05708436 -0.05382956 -0.1616835 0.3793801 -0.005958968 0.4973411
## 2 0.1617988 -0.05224505 -0.06489857 -0.1604269 0.3855894  0.018726338 0.5047900
## 3 0.1416854 -0.04703205 -0.03559731 -0.1550070 0.3969614  0.019825980 0.5017759
## 4 0.1542110 -0.06280810 -0.06337177 -0.1801324 0.3627478 -0.004908913 0.5143443
## 5 0.1348774 -0.05406377 -0.08551746 -0.1561645 0.3779071  0.022926740 0.5116010
## 6 0.1682413 -0.04500875 -0.03184752 -0.1633230 0.3815782  0.017134823 0.5189145
##          X19        X20        X21        X22        X23        X24
## 1 -0.4271076 -0.2202427 0.02367071 -0.2788570 -0.6747245 0.12224557
## 2 -0.4371284 -0.2115554 0.04778819 -0.2757116 -0.6606291 0.13680339
## 3 -0.4680668 -0.2824263 0.05803127 -0.3106933 -0.7495399 0.11420732
## 4 -0.4367349 -0.2440492 0.04128515 -0.2676063 -0.6687455 0.11743054
## 5 -0.4181175 -0.2075909 0.06378071 -0.2871105 -0.6503714 0.13403615
## 6 -0.4742091 -0.2720678 0.06608412 -0.2798534 -0.7314249 0.09062206
```

Fit a Model

We fit K-means here and take a look of the model:

```r
fit <- kmeans(emb_df[,-1], 5,)
# Cluster
fit$cluster
```

```
##  [1] 5 1 2 5 4 2 1 4 3 3 1 4
```

```r
# Summary
summary(fit)
```

```
##              Length Class  Mode
## cluster       12    -none- numeric
## centers      125    -none- numeric
## totss          1    -none- numeric
## withinss       5    -none- numeric
## tot.withinss   1    -none- numeric
## betweenss      1    -none- numeric
## size           5    -none- numeric
## iter           1    -none- numeric
## ifault         1    -none- numeric
```

```r
# Names
names(fit)
```

```
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

```r
#tot.withinss
fit$tot.withinss
```

```
## [1] 0.02301187
```

SubSample

We run a quick loop here to sub sample the clusters for potential later use:

```r
subsample <- list()
for(i in 1:5){
  subsample[[i]]<- emb_df[fit$cluster==i,-1]
}
```

Let's veryfiy those centers given the cluster labels:

```r
fit$centers[1:20]
```

```
##  [1] -0.15769295 -0.17457822 -0.13376568 -0.15499543 -0.14844621 -0.06808316
##  [7] -0.03631094 -0.05532296 -0.05850993 -0.05177414  0.05672679  0.07896056
## [13]  0.08057165  0.07056261  0.06605485 -0.09623007 -0.13070769 -0.11672436
## [19] -0.09795589 -0.08443877
```

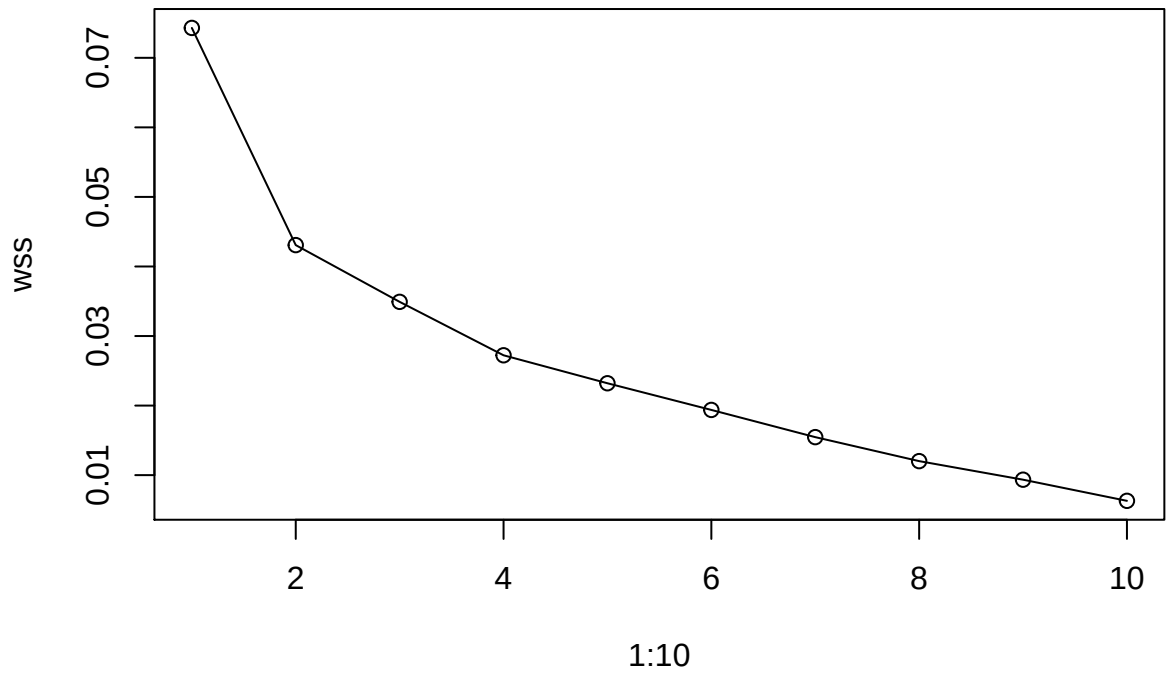
```r
apply(subsample[[1]], 2, mean)
```

```
##           X0           X1           X2           X3           X4           X5
## -0.157692953 -0.068083161  0.056726787 -0.096230070 -0.311057090  0.354578053
##           X6           X7           X8           X9          X10          X11
##  0.168170547 -0.256425803  0.206463060  0.091204687 -0.089015779  0.105819843
##          X12          X13          X14          X15          X16          X17
##  0.144602290 -0.060515177 -0.062432888 -0.170880833  0.382958777  0.004104897
##          X18          X19          X20          X21          X22          X23
##  0.503299207 -0.422734257 -0.224859520  0.059138627 -0.284777113 -0.663934950
##          X24
##  0.137470297
```

We can run a quick check for the choice of K here:

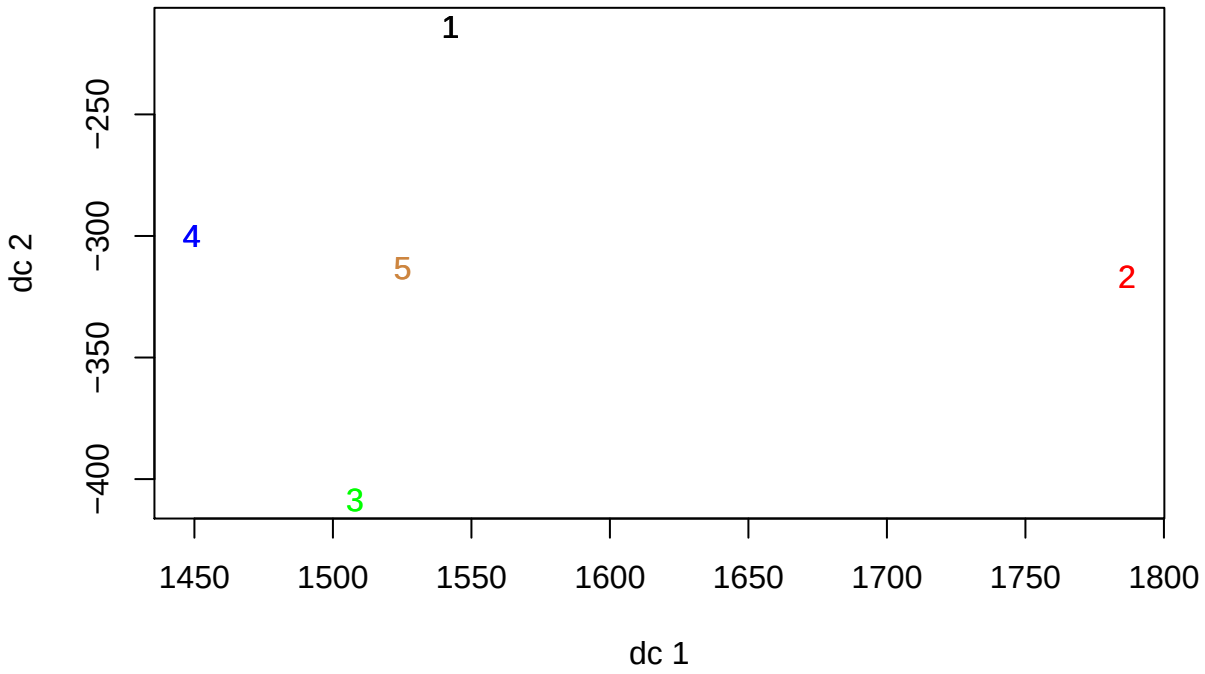
```r
########## choice of K
wss<- NULL
for (i in 1:10){
  fit1=kmeans(emb_df[,-1],centers = i)
  wss=c(wss, fit1$tot.withinss)
}

plot(1:10, wss, type = "o")
```

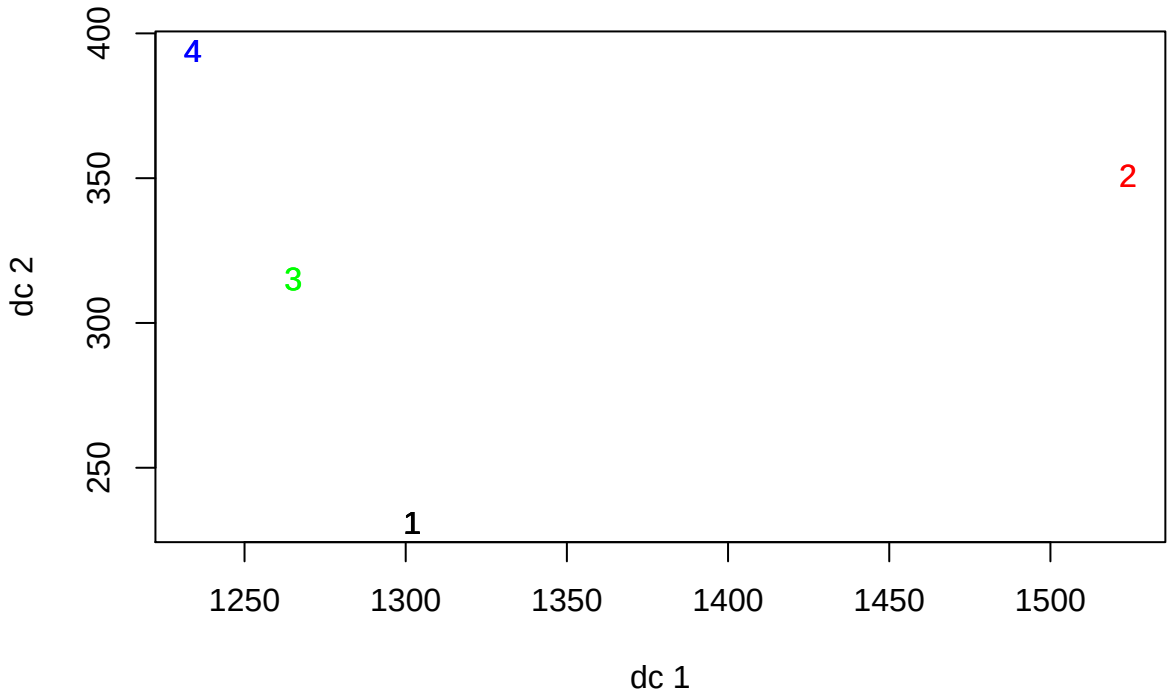


```r
fit2 <- kmeans(emb_df[,-1], 4,)

plotcluster(emb_df[,-1],fit$cluster)
```

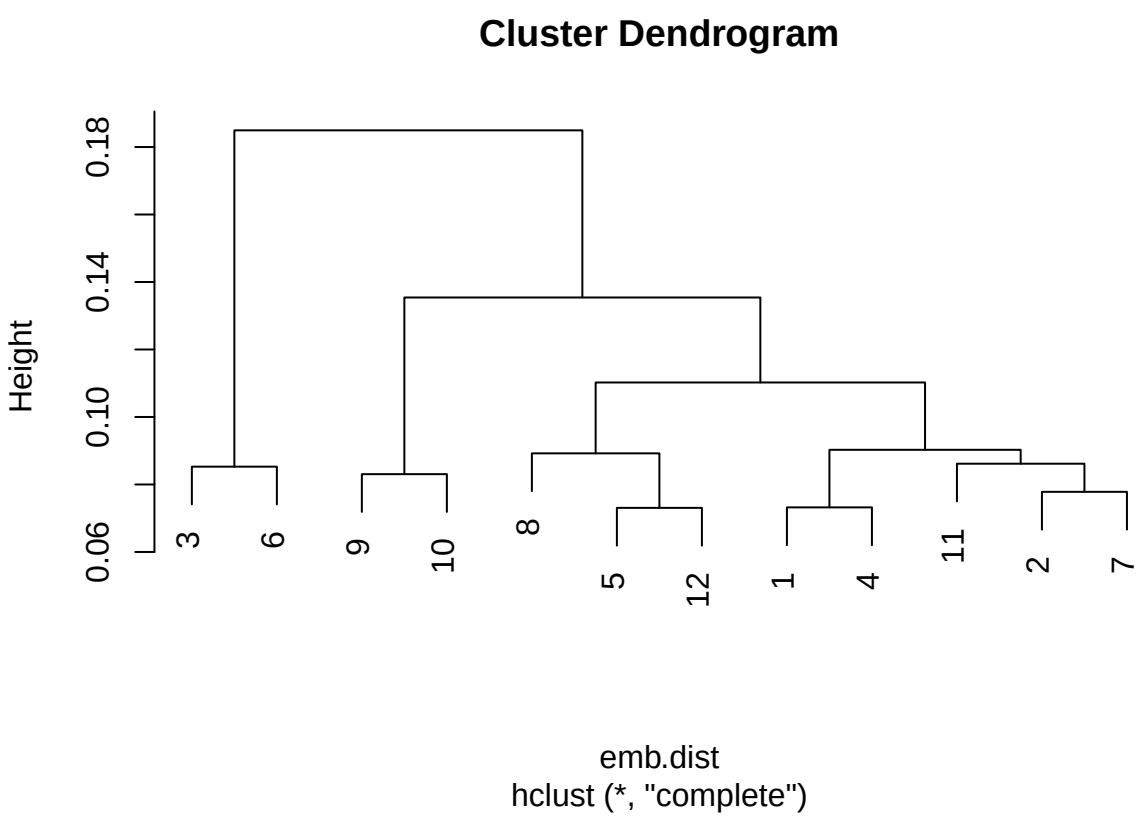



```r
plotcluster(emb_df[,-1],fit2$cluster)
```



Performe hierarchical clustering:

```
## calculate the distance matrix
emb.dist<- dist(emb_df[,-1])
#obtain clusters
emb.hcluster<- hclust(emb.dist)
plot(emb.hcluster)
```


Cluster Dendrogram
Height
0.06
0.10
0.14
0.18
3
6
9
10
8
5
12
1
4
11
2
7
emb.dist
hclust (*, "complete")