QF301. Lecture 19 In-Class Assignment.

2021-11-08

I pledge on my honor that I have not given or received any unauthorized assistance on this assignment/examination. I further pledge that I have not copied any material from a book, article, the Internet or any other source except where I have expressly cited the source.

By filling out the following fields, you are signing this pledge. No assignment will get credit without being pledged.

Name: Siddharth Iyer

CWID: 10447455

Date: 11/8/2021

Question 1 (45pt)

Question 1.1

```
CWID = 10447455 #Place here your Campus wide ID number, this will personalize
#your results, but still maintain the reproduceable nature of using seeds.
#If you ever need to reset the seed in this assignment, use this as your seed
#Papers that use -1 as this CWID variable will earn 0's so make sure you change
#this value before you submit your work.
personal = CWID %% 10000
set.seed(personal)
```

Obtain the adjusted close prices and daily log returns for 20 different stocks from January 1, 2010 until December 31, 2019. Save these prices and returns into data frames. Print the first 6 lines of each data frame.

Solution:

```
stocks = c("MMM", "ABT", "AMD", "AAP", "AFL", "GOOG", "AMZN", "AAL", "AXP", "FDX", "AAPL", "T", "BBY",

library(quantmod)

for(stock in stocks){
  getSymbols(stock, from="2010-01-01", to="2019-12-31", src="yahoo")
}

price_df = data.frame(MMM$MMM.Adjusted, ABT$ABT.Adjusted, AMD$AMD.Adjusted,
                      AAP$AAP.Adjusted, AFL$AFL.Adjusted, GOOG$GOOG.Adjusted,
                      AMZN$AMZN.Adjusted, AAL$AAL.Adjusted, AXP$AXP.Adjusted,
                      FDX$FDX.Adjusted, AAPL$AAPL.Adjusted, T$T.Adjusted,
                      BBY$BBY.Adjusted, BLK$BLK.Adjusted, BK$BK.Adjusted,
                      COF$COF.Adjusted, CMG$CMG.Adjusted, C$C.Adjusted,
                      COST$COST.Adjusted, DPZ$DPZ.Adjusted)
colnames(price_df) <- stocks
```

```
rets_df = log(price_df[-1,]/price_df[-length(price_df),])

head(price_df)
```

```
##                 MMM      ABT  AMD      AAP      AFL     GOOG   AMZN      AAL
## 2010-01-04 60.37971 19.84859 9.70 38.66294 17.79271 312.2048 133.90 4.496876
## 2010-01-05 60.00151 19.68823 9.71 38.43313 18.30887 310.8299 134.69 5.005958
## 2010-01-06 60.85244 19.79757 9.57 38.76824 18.46971 302.9943 132.25 4.798554
## 2010-01-07 60.89609 19.96157 9.47 38.75868 18.66793 295.9407 130.00 4.939965
## 2010-01-08 61.32520 20.06363 9.43 38.91187 18.48092 299.8860 133.52 4.845692
## 2010-01-11 61.07793 20.16568 9.14 38.52889 18.96717 299.4326 130.31 4.751417
##                 AXP      FDX     AAPL        T      BBY      BLK       BK
## 2010-01-04 34.23098 75.44473 6.553026 14.38626 28.88457 175.8633 22.05763
## 2010-01-05 34.15570 76.43015 6.564356 14.31579 29.61764 176.6226 22.29154
## 2010-01-06 34.70783 75.79733 6.459940 14.10632 29.38766 172.9811 21.95626
## 2010-01-07 35.27073 74.97460 6.447998 13.94793 29.85482 174.8829 22.88411
## 2010-01-08 35.24553 76.83698 6.490867 13.84575 28.68333 176.1139 23.02445
## 2010-01-11 34.84222 78.88020 6.433607 13.77933 28.19460 178.9814 22.62681
##                 COF   CMG        C     COST      DPZ
## 2010-01-04 33.05940 87.84 29.26650 44.22290 7.331961
## 2010-01-05 34.36316 89.02 30.38550 44.08892 7.399538
## 2010-01-06 34.54940 87.32 31.33236 44.66208 7.610710
## 2010-01-07 36.25952 86.43 31.41844 44.44620 7.914802
## 2010-01-08 35.94628 91.89 30.90197 44.12614 7.914802
## 2010-01-11 35.30288 96.77 31.24627 44.17823 7.973932
```

```
head(rets_df)
```

```
##                      MMM          ABT          AMD          AAP          AFL
## 2010-01-05 -0.0062832764 -0.008112179  0.001030397 -0.005961644  0.028596880
## 2010-01-06  0.0140821220  0.005538208 -0.014523077  0.008681403  0.008746398
## 2010-01-07  0.0007170190  0.008250024 -0.010504298 -0.000246624  0.010675416
## 2010-01-08  0.0070218337  0.005099498 -0.004232811  0.003944743 -0.010068390
## 2010-01-11 -0.0040401146  0.005073576 -0.031235711 -0.009890995  0.025970635
## 2010-01-12  0.0008330145 -0.002896183 -0.055101065 -0.017548834 -0.005140894
##                     GOOG         AMZN          AAL          AXP          FDX
## 2010-01-05 -0.004413395  0.005882649  0.107245870 -0.0022015702  0.012977022
## 2010-01-06 -0.025531931 -0.018281786 -0.042314181  0.0160358862 -0.008314289
## 2010-01-07 -0.023554756 -0.017159620  0.029043624  0.0160881321 -0.010913586
## 2010-01-08  0.013243041  0.026716859 -0.019268183 -0.0007146153  0.024536658
## 2010-01-11 -0.001512745 -0.024335098 -0.019647173 -0.0115088132  0.026244210
## 2010-01-12 -0.017842125 -0.022977026  0.007905251  0.0131762578 -0.007708826
##                    AAPL           T          BBY          BLK           BK
## 2010-01-05  0.001727479 -0.0049107392  0.025062652  0.004307793  0.010548660
## 2010-01-06 -0.016034377 -0.0147403438 -0.007795474 -0.020832449 -0.015154849
## 2010-01-07 -0.001850335 -0.0112914585  0.015771343  0.010933973  0.041390221
## 2010-01-08  0.006626417 -0.0073527111 -0.040029910  0.007014526  0.006113999
## 2010-01-11 -0.008860767 -0.0048086824 -0.017185610  0.016150647 -0.017421297
## 2010-01-12 -0.011440303 -0.0003708414  0.000764464 -0.018035834  0.001721518
##                     COF         CMG           C         COST         DPZ
## 2010-01-05  0.038679027  0.01334410  0.03752207 -0.003034228 0.009174555
## 2010-01-06  0.005405269 -0.01928150  0.03068600  0.012916181 0.028138900
```

```
## 2010-01-07  0.048311695 -0.01024469  0.00274352 -0.004845283 0.039178211
## 2010-01-08 -0.008676449  0.06125736 -0.01657504 -0.007227051 0.000000000
## 2010-01-11 -0.018061112  0.05174480  0.01107991  0.001179761 0.007443044
## 2010-01-12  0.003829513 -0.01404882 -0.03077144 -0.005406316 0.049596682
```

Question 1.2

Cluster these stocks based on the adjusted close prices.
Use K-Means Clustering with 4 clusters and 20 attempts at clustering Print the clusters. Do these match with market sectors?

Solution:

```
km.equity = kmeans(t(price_df),4,nstart=20)
sort(km.equity$cluster)
```

```
## GOOG AMZN  MMM  AAP  FDX  BLK COST  DPZ  CMG  ABT  AMD  AFL  AAL  AXP AAPL    T
##    1    1    2    2    2    2    2    2    3    4    4    4    4    4    4    4
##  BBY   BK  COF    C
##    4    4    4    4
```

CMG: Consumer Discretionary

MMM: Industrial AAP: Consumer Discretionary FDX: Industrial BLK: Financials COST: Consumer Staples DPZ: Consumer Discretionary

ABT: Health Care AMD: IT AFL: Financials AAL: Industrial AXP: Financials AAPL: IT T: Communication Services BBY: Consumer Discretionary BK: Financials COF: Financials C: Financials

GOOG: Communication Services AMZN: Consumer Discretionary

There doesn't seem to be a clustering based on sector according to this dataset.

Question 1.3

Repeat Question 1.2 but normalize the adjusted close prices first. How do your clusters differ (if at all)?

Solution:

```
km.equity.scaled = kmeans(t(scale(price_df)),4,nstart=20)
sort(km.equity.scaled$cluster)
```

```
##  ABT  AFL GOOG AMZN  AXP AAPL  BBY    C COST  DPZ  MMM  AAL  FDX    T  BLK   BK
##    1    1    1    1    1    1    1    1    1    1    2    2    2    2    2    2
##  COF  AMD  AAP  CMG
##    2    3    4    4
```

1 - Consumer Discretionary 2 - Financials + Discretionary products (I'd say this is growth cluster) 3 - I'd say this is Industrial + large asset financials that are correlated with industrial

Question 1.4

Repeat Question 1.2 but with the log returns. How do your clusters differ (if at all)?

Solution:

```
km.logrets = kmeans(t(rets_df),4,nstart=20)
sort(km.logrets$cluster)
```

```
##  AAL  AMD  AXP  COF  MMM  ABT  AAP  AFL GOOG AMZN  FDX AAPL    T  BBY  BLK   BK
##    1    2    2    2    3    3    3    3    3    3    3    3    3    3    3    3
##  CMG    C COST  DPZ
##    3    3    3    4
```

The clusters appear to be even worse now... there is almost to sector classification. See above for ticker industries.

Question 1.5

Repeat Question 1.4 but normalize the the log returns first. How do your clusters differ (if at all)?

Solution:

```
km.logrets.scaled = kmeans(t(scale(rets_df)),4,nstart=20)
sort(km.logrets.scaled$cluster)
```

```
##  AFL  AAL  BBY    C  DPZ  MMM  ABT  AAP AAPL    T  BLK COST  AMD GOOG AMZN  AXP
##    1    1    1    1    1    2    2    2    2    2    2    2    3    3    3    3
##   BK  COF  FDX  CMG
##    3    3    4    4
```

Yes, the cluster has changed. 1 - No apparent cluster based on industry. 2 - No apparent cluster based on industry. 3 - No pure cluster based on indstury.

Question 1.6

Do any of the clusters considered with the K-Means Clustering algorithm seem to match your intuition best? Briefly (2-3 sentences) comment.

Solution:

Scaled price seems to do the best... it doesn't classify purely by industry, but it does somehow get the growth clusters vs industrial/large asset financial clusters. All of them are still incredibly bad and shouldn't be used in a real-world analysis.

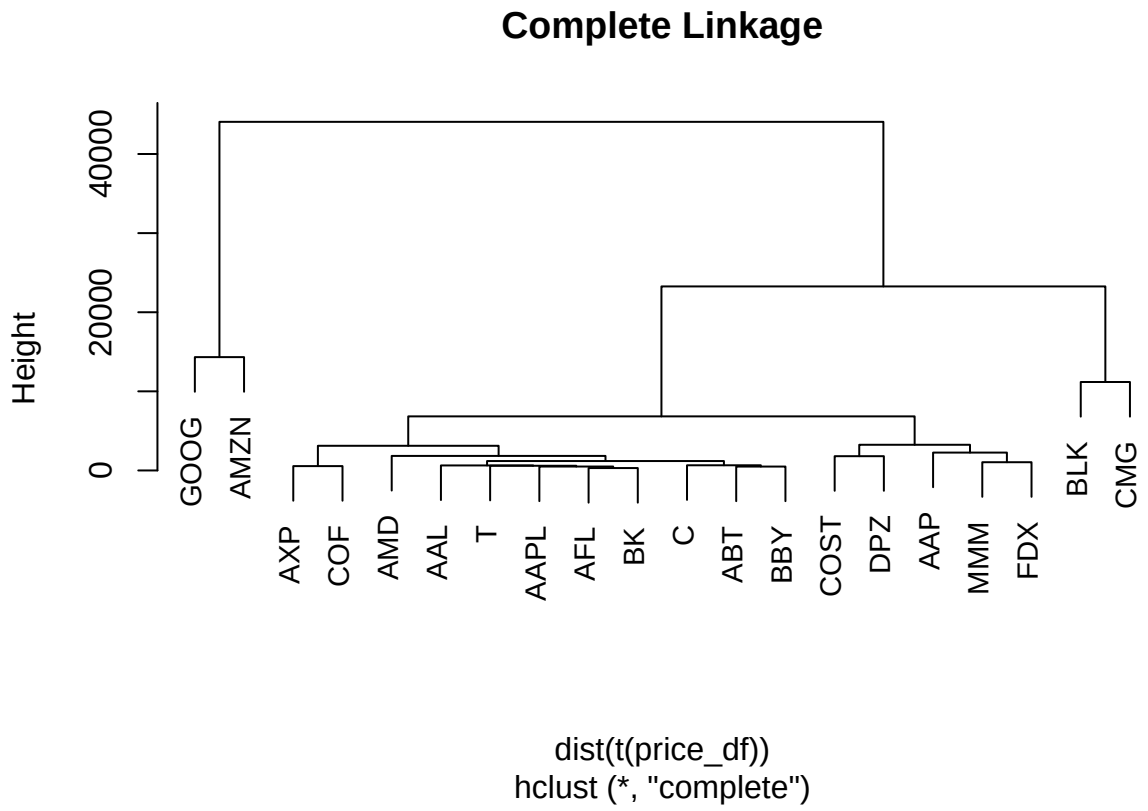
Question 2 (45 pt)

Question 2.1

Cluster these stocks based on the adjusted close prices.
Use Hierarchical Clustering with Complete Distance Metric. Print the dendrogram. Print the clusters when cut at 4 clusters. Do these match with market sectors?

Solution:

```
hc.complete=hclust(dist(t(price_df)),method="complete")
plot(hc.complete,main="Complete Linkage",cex=.9)
```

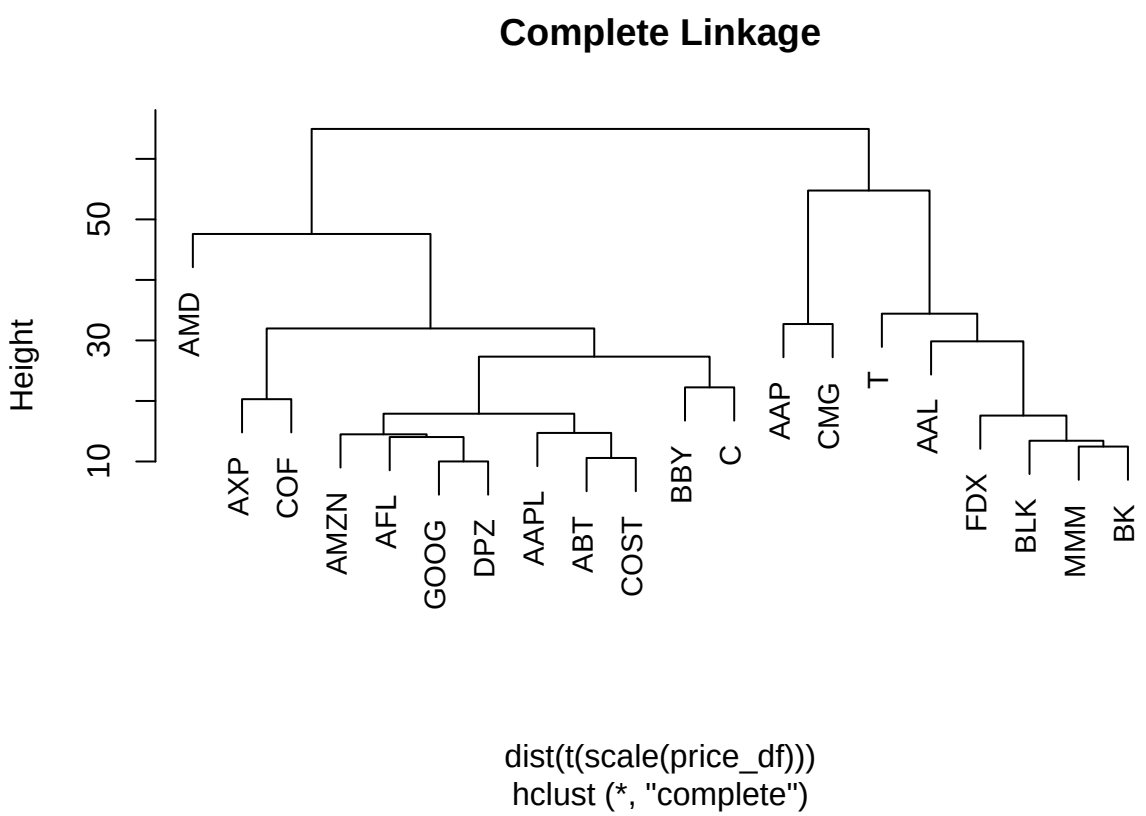


Question 2.2

Repeat Question 2.1 but normalize the adjusted close prices first. How do your clusters differ (if at all)?

Solution:

```
hc.complete.scaled=hclust(dist(t(scale(price_df))),method="complete")
plot(hc.complete.scaled,main="Complete Linkage",cex=.9)
```



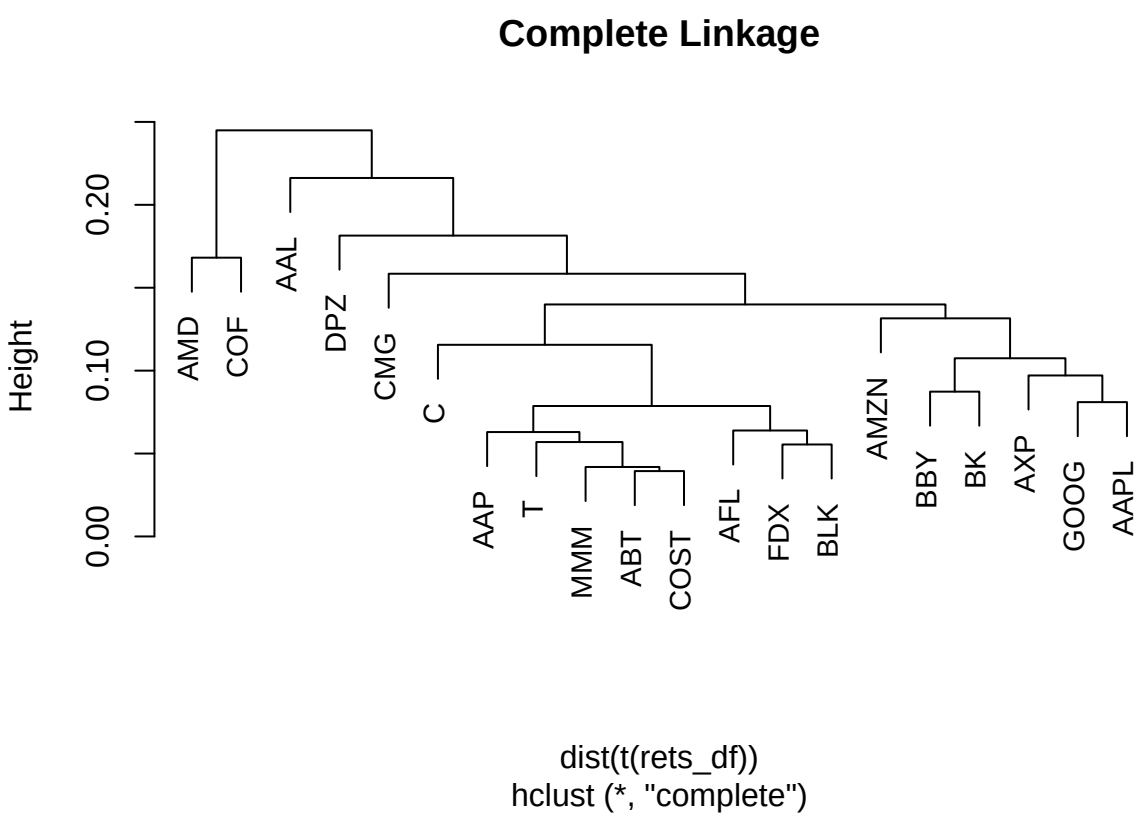
They are much more different. . . the scaled data eliminates some of the clusters previously identified. The financials, consumer discretionary, and IT are grouped together, which grow during expansion cycles. . . this might be good.

Question 2.3

Repeat Question 2.1 but with the log returns. How do your clusters differ (if at all)?

Solution:

```
hc.complete.rets=hclust(dist(t(rets_df)),method="complete")
plot(hc.complete.rets,main="Complete Linkage",cex=.9)
```



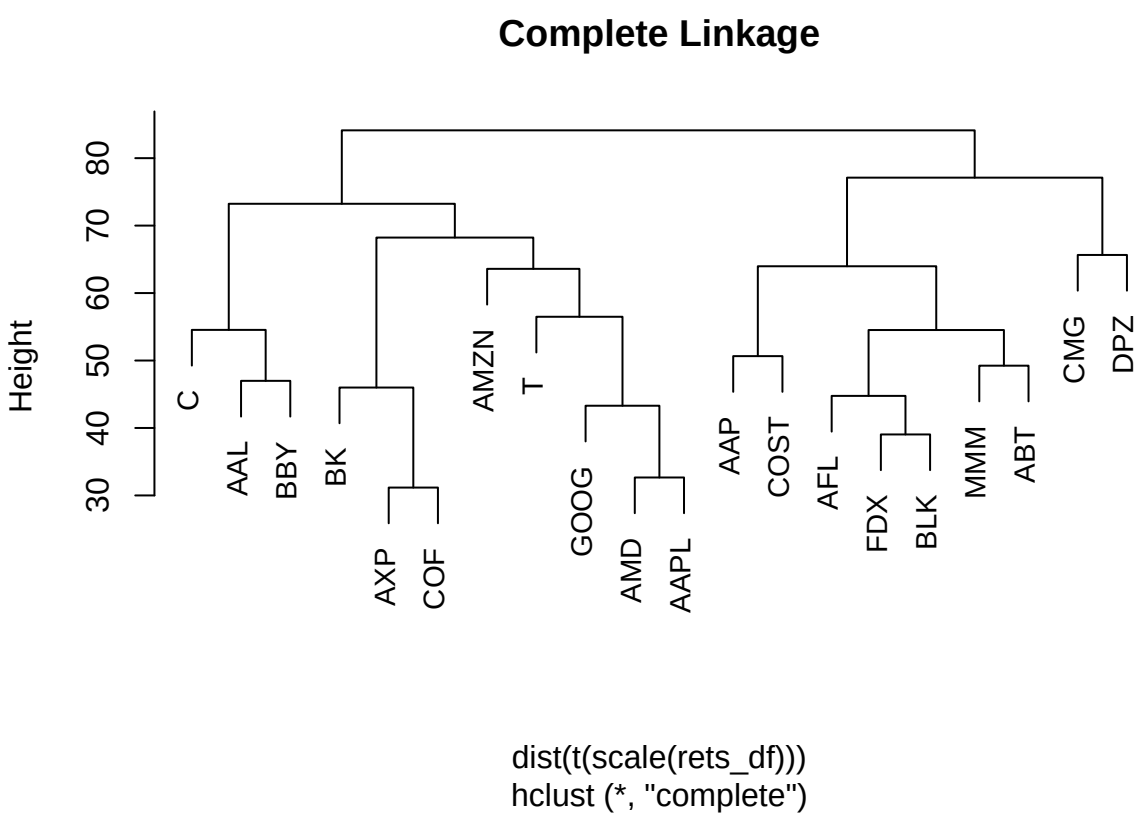
Heavily tiered, which means there is a lot of node impurity in this model.

Question 2.4

Repeat Question 2.3 but normalize the the log returns first. How do your clusters differ (if at all)?

Solution:

```
hc.complete.lognorm=hclust(dist(t(scale(rets_df))),method="complete")
plot(hc.complete.lognorm,main="Complete Linkage",cex=.9)
```



very good clustering of IT stocks..AMZN, GOOG, AMD, AAPL. Further, industrial and financial stocks are well grouped together, although they eventually fall into the same cluster, which is undesirable

Question 2.5

Do any of the clusters considered with the Hierarchical Clustering algorithm seem to match your intuition best? Briefly (2-3 sentences) comment.

Solution:

The last model, scaled returns, provides very good clustering of IT stocks..AMZN, GOOG, AMD, AAPL. Further, industrial and financial stocks are well grouped together, although they eventually fall into the same cluster, which is undesirable.

Question 3 (10pt)

Briefly (2-3 sentences) comment on if you prefer K-Means or Hierarchical clustering on your data.

Solution:

I strongly prefer Hierarchical clustering because it clearly shows intermediate classifications that contributed to the result. Also, this doesn't suffer from as much dimensionality issues as K-Means.