Arif Ali
Statistics 154
02/14/14
Section 01

Homework 1

(Graphs and Code in Appendix)

Problem 1

Part 1

Based on the first few vectors, there is strong indication that the price value, BIDLO, ASKHI, and OPENPRC have the most effect with regards to the placement of the stock points in the various dimensions. However, the results for VOL and NUMTRD indicate a less meaningful impact. Daily Return, surprisingly, has a much lower impact then was expected. This indicates that Daily Returns will not be a major factor in the PCA is the stocks.

	PC1	PC2	PC3	PC4	PC5	PC6
BIDLO	0.5701116434	0.089078735	-0.0002062798	0.0218217762	7.643797e-01	0.2868539175
ASKHI	0.5700385209	0.089997837	-0.0002350854	0.0144604427	-6.306924e-01	0.5186274850
VOL	-0.1311507964	0.692890220	0.0013111857	0.7090065758	-3.244605e-03	0.0002001361
OPENPRC	0.5700748738	0.089714437	-0.0004380042	0.0173918177	-1.338769e-01	-0.8054441440
NUMTRD	-0.0885574273	0.704143999	0.0002837167	-0.7045008131	4.160755e-03	-0.0001516990
DailyReturns	0.0006983944	-0.001029459	0.9999989553	-0.0007142428	-4.615539e-05	-0.0001719136

Part 2

The data was subset to look only at the range of time that corresponded to the 2009 year. By doing this, I only looked at three companies, AMGEN INC, APARTMENT INVESTMENT & MGMT CO, and INVESCO LTD. Based on the plot of the hierarchical clustering; Amgen and Invesco are on the same level and Apartment Investment is several levels above the other two. This indicates that Apartment Investment is significantly different with regarding to the daily returns.

Problem 2

Part 1

The PCA indicates that there is one outlier at the top level of the PCA plot, which may indicate that one of the set gene expression, has a weaker link. The majority of the points are located in the lower left of a plot. This grouping indicates where the majority of the clusters are centered and thus where the most influence of the variance is.

With respect to the Kernel PCA, the differences are the magnitude of points between Kernel PCA and PCA. The Kernel PCA plot has a reduced number of points. This may be because the kernel PCA maps the multiple dimensions on a 2- dimension representation. The PCA requires that the components be brought down to the second dimension.

Part 2
The K- mediod's graph specifies that component1 and component2 explain 66.21% of the point's variability. The 14 mediods are mainly centered between -50 and 0 of component1 and -50 to 50 of component2.

Problem 3
Attempting to reproduce the simulation required converting xy Cartesian points with polar points in order to create the circle based points. The noise was added after the points were converted to polar coordinates.

The plotting between PCA and Kernel PCA differs because the PCA reflects the polar coordinates described in the first graph of 14.29; however, the Kernel PCA reflects a transition of points to a more uniform distributions.

Problem 4
The first scree plot (see appendix) is an example of a good scree plot because the variances are gradually decreasing. Ensuring the different columns in the data frame have increasing standard deviations did this.

The Second scree plot is an example of a bad scree plot because the differences between the eigenvalues are not significantly different indicating that there is no predictive value. This was done by ensuring vectors within the data frame will be the same.

Problem 5
The Simulation was constructed with a vector constructed with a normal distribution. The following four vectors, involve using the last 1000 of the first vector and a normal distribution for the remaining 9000. After the construction of the first five vectors, a sixth vector was added but with a normal distribution of a different standard deviation.

In order to ensure faithfulness, I compared the first vector after adding the sixth and seven vector. By comparing the values of the variances, the results were found to have differences that were negligible.

Problem 6
Making the variance of the fist 4 vectors created the simulation, four times the variance for the last 16 vectors resulted in the construction. Looking at two iterations of the simulation, the stability was determined by looking at the different similar to determining faithfulness. The simulation reinforces the value of variances regarding PCA and

Problem 7

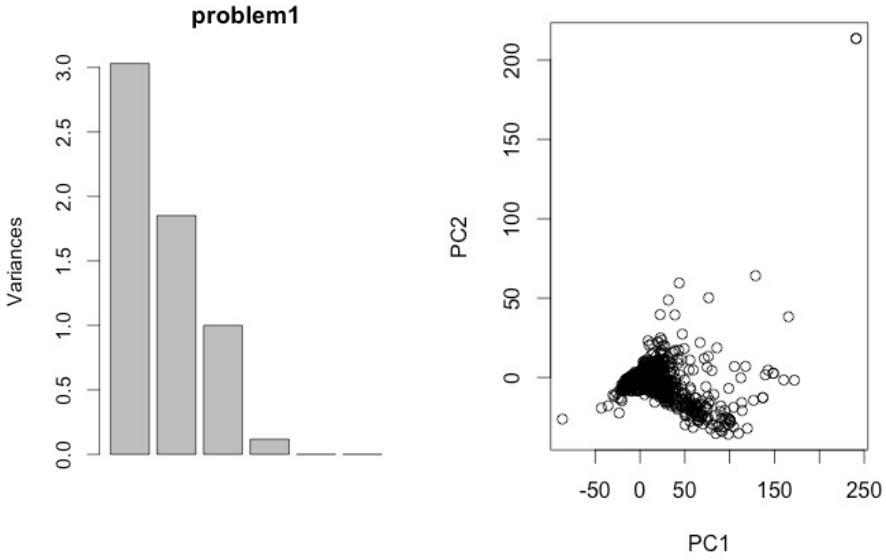
qnorm(0.75,0,1) - qnorm(0.25,0,1)= 1.3498

qnorm(0.75, $\boldsymbol{\mu}$, σ) - qnorm(0.25, $\boldsymbol{\mu}$, σ)= 1.3498 σ

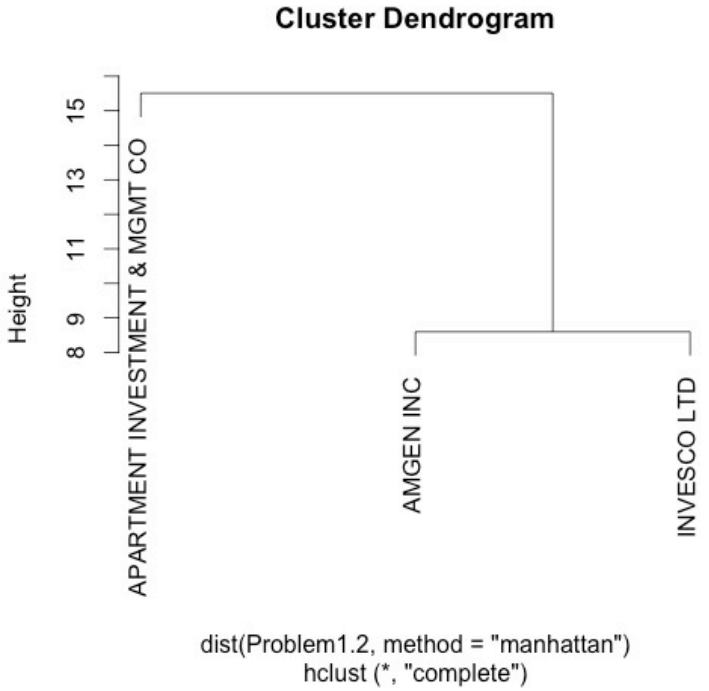
1-pnorm(1.5*1.349+qnorm(0.75)) = 0.003487979

APPENDIX

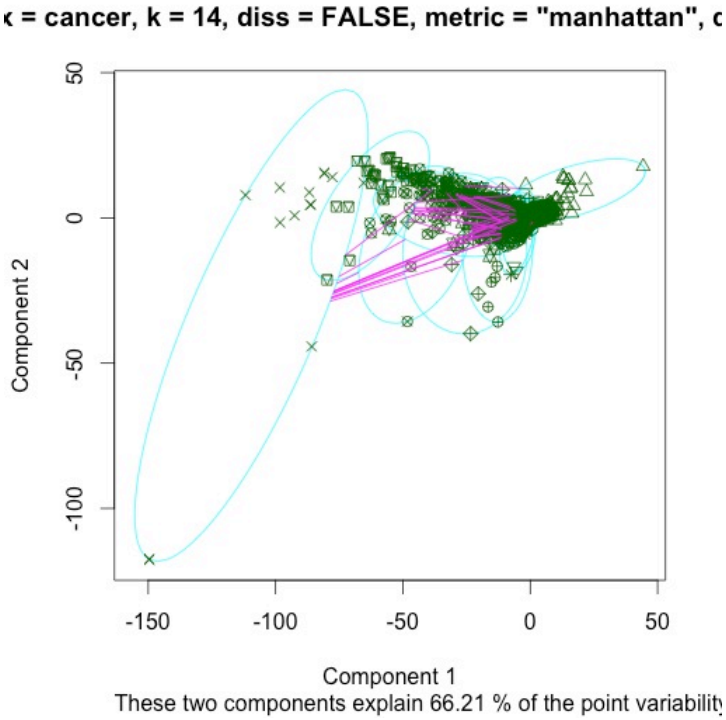
Graphs
Problem 1.1



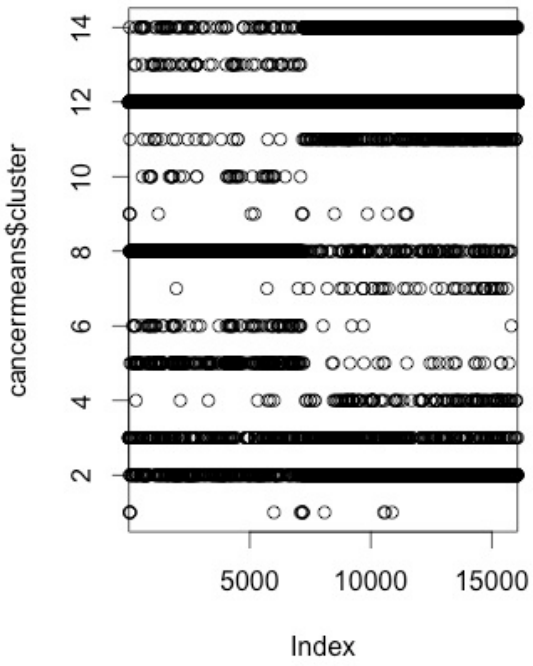
Problem 1.2



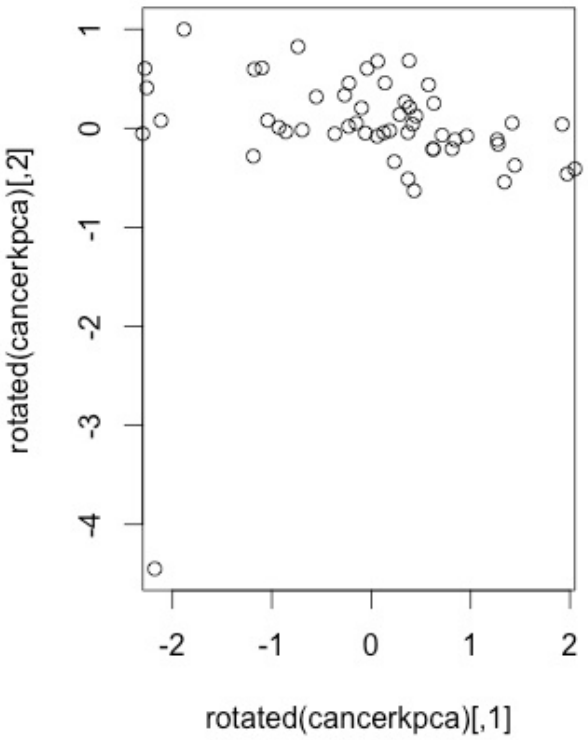
Problem 2.1



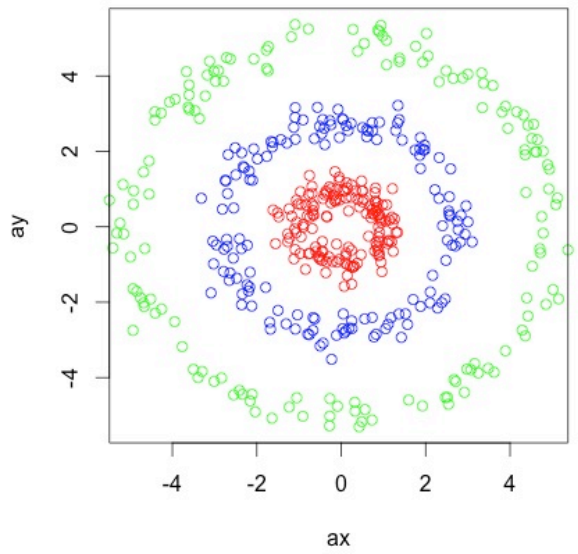
Problem 2.2.1



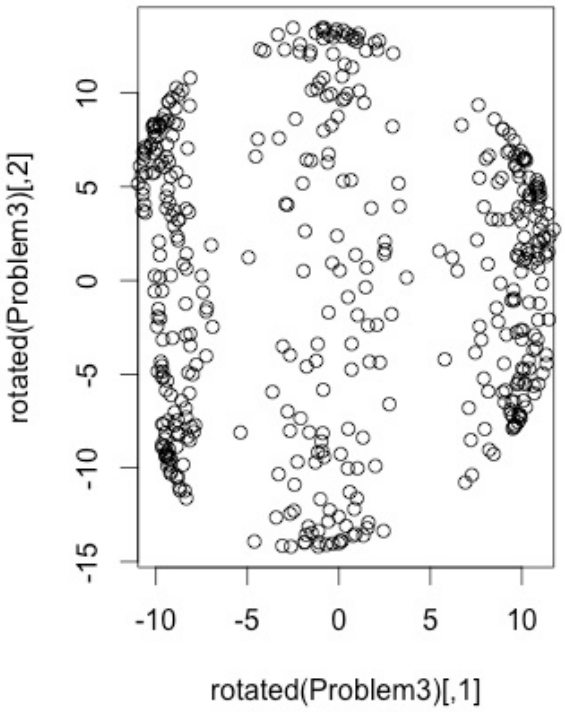
Problem 2.2.2



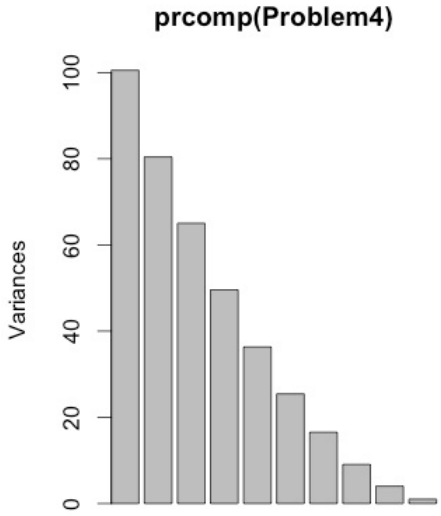
Problem 3.1



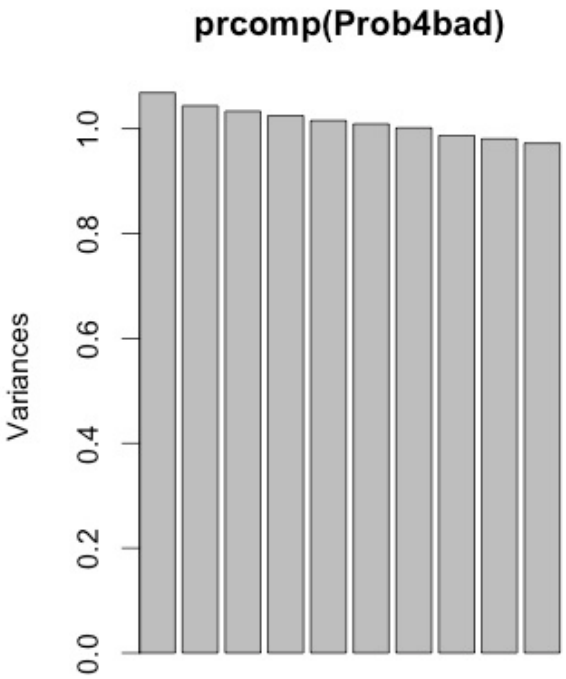
Problem 3.2



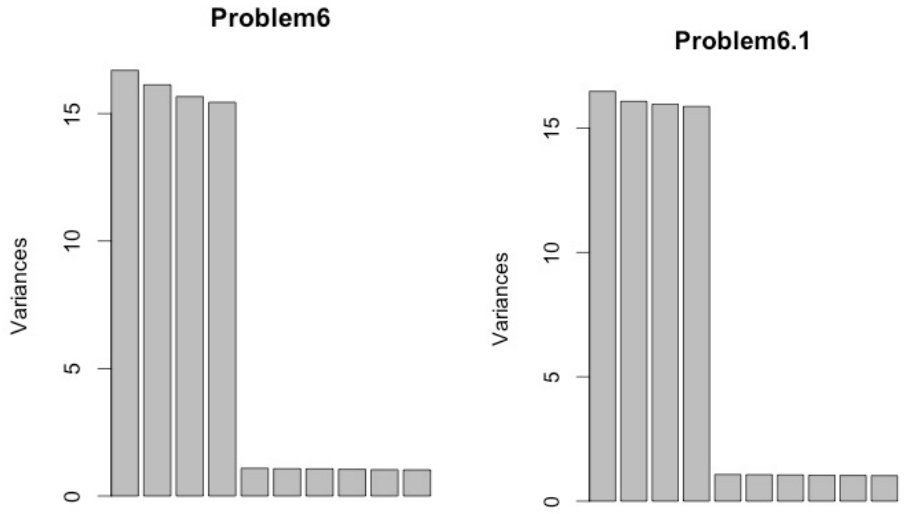
Problem 4.1



Problem 4.2



Problem 6



Code

```
####Problem 1.1
setwd("Dropbox/School/Statistics/Stat 154 Spring 2014/HW1/")
SP500 = read.csv("sp500Data.csv")
sp500Components = read.csv("bf4024f5c75fc062.csv")
#Rt =St/St11
DailyReturns = c()
tick = unique(sp500Components$TICKER)
for(i in 1:length(unique(sp500Components$TICKER))){
  Ticker = sp500Components[sp500Components$TICKER == tick[i], ]
  uniquediff = (Ticker$PRC[2:length(Ticker$PRC)]/Ticker$PRC[1:(length(Ticker$PRC)-1)])-1
  DailyReturns = c(DailyReturns, 0, uniquediff)
}
sp500Components$DailyReturns = DailyReturns

problem1 = sp500Components[, c("BIDLO", "ASKHI", "VOL", "OPENPRC", "NUMTRD", "DailyReturns")]
problem1 = na.omit(problem1)
problem1 = prcomp(problem1, scale=T)
#Because the Mean not equal to 0, scale=T
problem1$rotation
#                 PC1         PC2          PC3          PC4          PC5          PC6
# BIDLO        0.5701116434  0.089078735 -0.0002062798  0.0218217762  7.643797e-01  0.2868539175
# ASKHI        0.5700385209  0.089997837 -0.0002350854  0.0144604427 -6.306924e-01
0.5186274850
# VOL         -0.1311507964  0.692890220  0.0013111857  0.7090065758 -3.244605e-03  0.0002001361
# OPENPRC      0.5700748738  0.089714437 -0.0004380042  0.0173918177 -1.338769e-01 -
0.8054441440
# NUMTRD      -0.0885574273  0.704143999  0.0002837167 -0.7045008131  4.160755e-03 -
0.0001516990
# DailyReturns  0.0006983944 -0.001029459  0.9999989553 -0.0007142428 -4.615539e-05 -
0.0001719136

plot(problem1$x)
screeplot(problem1)
####Problem 1.2
###2009 starts at row 337 (20090102 )and ends at 588 (20091231) for the first time
range = sp500Components$date[337:588]
range.frame = sp500Components[sp500Components$date==sp500Components$date[337:588],]
company.numbers = unique(range.frame$COMNAM)
Problem1.2 = data.frame(range.frame[range.frame$COMNAM == company.numbers[1], "DailyReturns"])
for(i in 2:length(company.numbers)){
  Problem1.2[,i] = range.frame[range.frame$COMNAM == company.numbers[i], "DailyReturns"]
}
names(Problem1.2) = company.numbers
Problem1.2 = t(Problem1.2)
Problem1.2 = as.matrix(Problem1.2)
Problem1.2 = hclust(dist(Problem1.2, method = "manhattan"))
plot(Problem1.2)
###look at the hcluster
####Problem 2.1
Problem2 = prcomp(cancer, center = T)
Problem2$x[,c(1,2)]
plot(Problem2$x[,c(1,2)])
screeplot(Problem2)
```

```
library("kernlab")
cancerkpca = kpca(cancer)
plot(rotated(cancerkpca))
####Problem 2.2
library("cluster")
cancermediod=pam(cancer,14,diss=
FALSE,metric="manhattan",do.swap=FALSE)#,cluster.only=TRUE,do.swap=FALSE)
plot(cancermediod)
cancermeans=kmeans(cancer,14,nstart=10)
length(cancermeans$cluster)
plot(cancermeans$cluster)
####Problem 3
aa = runif(150, 0, 2*pi)
ax = 5*cos(aa) + rnorm(150, 0, 0.25)
ay = 5*sin(aa) + rnorm(150, 0, 0.25)

bb = runif(150, 0, 2*pi)
bx = 2.8*cos(bb) + rnorm(150, 0, 0.25)
by = 2.8*sin(bb) + rnorm(150, 0, 0.25)

cc =  runif(150, 0, 2*pi)
cx = cos(cc) + rnorm(150, 0, 0.25)
cy = sin(cc) + rnorm(150, 0, 0.25)
plot(x = ax, y = ay, col = "green")
points(x = bx, y = by, col = "blue")
points(x = cx, y = cy, col = "red")

combined.points = data.frame(c(ax, bx, cx), c(ay, by, cy))
combined.points = as.matrix(combined.points)
Prob3 = prcomp(combined.points)
Problem3 = kpca(combined.points, kernel = "rbfdot")
plot(rotated(Problem3))
####Problem 4
Problem4 = data.frame(rnorm(10000, 0, 1))
for(i in 2:10){
  Problem4[,i]= rnorm(10000, 0, i)
}
screeplot(prcomp(Problem4))
###good ones
Prob4bad = data.frame(rnorm(10000, 0, 1))
for(i in 2:10){
  Prob4bad[,i]= rnorm(10000, 0, 1)
}
screeplot(prcomp(Prob4bad))

####Problem 5
Problem5 = data.frame(V1=rnorm(10000, 0, 5))
for(i in 2:5){
  Problem5[, i] = c(Problem5[9001:10000, 1], rnorm(9000, 0, 5))
}
prcomp(Problem5)$rotation
Problem5[, 6] = rnorm(10000, 0, 1)
prcomp(Problem5)$rotation
Problem5[, 7] = rnorm(10000, 0, 2)
prcomp(Problem5)$rotation
```

```
####Problem 6
#par(mfrow=c(3,2))
#for(i in 1:6){
Problem6 = data.frame(rnorm(10000, 0, 4))
for(i in 2:4){
  Problem6[,i]=rnorm(10000, 0, 4)
}
for(i in 5:20){
  Problem6[,i]=rnorm(10000, 0, 1)
}
Problem6 = prcomp(Problem6)
Problem6
Problem6.1 = data.frame(rnorm(10000, 0, 4))
for(i in 2:4){
  Problem6.1[,i]=rnorm(10000, 0, 4)
}
for(i in 5:20){
  Problem6.1[,i]=rnorm(10000, 0, 1)
}
Problem6.1 = prcomp(Problem6.1)
Problem6$rotation[,1]
Problem6.1$rotation[,1]
plot(Problem6.1)
#}
####Problem 7
qnorm(0.75,0,1)-qnorm(0.25,0,1)
#1.3498
#given N(mu, sigma^2)
##mu - mu = 0
#1.3498*sigma
1-pnorm(1.5*1.349+qnorm(0.75))
```