ST557: HOMEWORK 3

Brian Cervantes Alvarez
November 13, 2023

Question 1

Part A

No, we reject the null hypothesis that the average temperature in Corvallis has stayed the same. It is highly suggested to go with the alternative where the average temperature has changed over the span of 50 years or 5 decades.

```
library(MASS)
# Read the data
tempData <- read.csv("TempData.csv")

# Extract the relevant columns
decadesData <- tempData[, c('X1950s', 'X1960s', 'X1970s', 'X1980s', 'X1990s', 'X2000s')]

# Calculate mean vector and covariance matrix
meanVec <- colMeans(decadesData)
covMat <- cov(decadesData)

n <- nrow(decadesData)
p <- ncol(decadesData)

# Calculate Hotelling's T2 statistic
T2 <- n * t(meanVec) %*% solve(covMat) %*% meanVec

# Degrees of freedom
df1 <- p
df2 <- n - p

# Calculate the p-value using F-distribution
pval <- pf(T2, df1, df2, lower.tail = FALSE)

# Results
print(paste0("Hotelling's T2 statistic: ", T2))
```

```
[1] "Hotelling's T2 statistic: 2691.95664981641"
```

```
print(paste0("Degrees of freedom: ", df1, ", ", df2))
```

```
[1] "Degrees of freedom: 6, 14"
```

```
print(paste0("P-value: ", pval))
```

```
[1] "P-value: 1.31332194600578e-20"
```

```r
# Check if the p-value is less than the significance level (e.g., 0.05)
if (pval < 0.05) {
  print("Reject the null hypothesis. Not all means are equal.")
} else {
  print("Fail to reject the null hypothesis. Means are equal.")
}
```

```
[1] "Reject the null hypothesis. Not all means are equal."
```

Question 1

Part B

```r
alpha <- 0.05
# Number of comparisons
m <- 5
bonferroniFactor <- qt(1 - alpha / (2 * m), df = df2)

# Confidence Intervals
confIntervals <- matrix(NA, nrow = m, ncol = 2)
for (i in 2:(m + 1)) {
  diffMean <- meanVec[i] - meanVec[1]
  se <- sqrt(covMat[i, i] / n + covMat[1, 1] / n - 2 * covMat[i, 1] / n)
  marginOfError <- bonferroniFactor * se
  confIntervals[i - 1, ] <- c(diffMean - marginOfError, diffMean + marginOfError)
}

# Display Confidence Intervals
for (i in 2:(m + 1)) {
  print(paste0("95% Bonferroni CI for (mu", i, " - mu1): [", confIntervals[i - 1, 1], ", ",
}
```

```
[1] "95% Bonferroni CI for (mu2 - mu1): [0.0887633994080454, 0.232436600591956]"
[1] "95% Bonferroni CI for (mu3 - mu1): [-0.0736528186511561, 0.121052818651159]"
[1] "95% Bonferroni CI for (mu4 - mu1): [0.163693378223059, 0.44040662177694]"
[1] "95% Bonferroni CI for (mu5 - mu1): [0.577861662145883, 0.882338337854114]"
[1] "95% Bonferroni CI for (mu6 - mu1): [0.422515459538185, 0.798084540461816]"
```

```r
# Check if intervals include 0
zeroIncluded <- any(confIntervals[, 1] <= 0 & confIntervals[, 2] >= 0)

# Conclusion
if (zeroIncluded) {
  print("At least one interval includes 0. Conclude that the corresponding means are not sig
} else {
  print("None of the intervals include 0. Conclude that all corresponding means are signifi
}
```

```
[1] "At least one interval includes 0. Conclude that the corresponding means are not significa
```

Question 2

Part A

Upon visual inspection, there are variations in the values across the matrices, indicating changes in the relationships between variables over time.

```
skullData <- read.csv("SkullData.csv")

names(skullData)
```

```
[1] "Year" "MB"   "BH"   "BL"   "NH"
```

```
# Split the data by Year
splitData <- split(skullData[, 2:5], skullData[, 1])

# Compute covariance matrices for each Year
covMatrices <- lapply(splitData, cov)
covMatrices
```

```
$`-4000`
          MB          BH          BL          NH
MB 26.309195  4.1517241  0.4540230  7.2459770
BH  4.151724 19.9724138 -0.7931034  0.3931034
BL  0.454023 -0.7931034 34.6264368 -1.9195402
NH  7.245977  0.3931034 -1.9195402  7.6367816

$`-3300`
          MB         BH          BL        NH
MB 23.136782  1.010345  4.7678161 1.8425287
BH  1.010345 21.596552  3.3655172 5.6241379
BL  4.767816  3.365517 18.8919540 0.1908046
NH  1.842529  5.624138  0.1908046 8.7367816

$`-1850`
           MB           BH          BL           NH
MB 12.1195402  0.78620690 -0.7747126  0.89885057
BH  0.7862069 24.78620690  3.5931034 -0.08965517
BL -0.7747126  3.59310345 20.7229885  1.67011494
NH  0.8988506 -0.08965517  1.6701149 12.59885057

$`-200`
          MB         BH         BL       NH
MB 15.362069 -5.534483 -2.172414 2.051724
BH -5.534483 26.355172  8.110345 6.148276
BL -2.172414  8.110345 21.085057 5.328736
NH  2.051724  6.148276  5.328736 7.964368

$`150`
           MB          BH          BL          NH
MB 28.6264368 -0.2298851 -1.8793103 -1.9942529
```

```
BH -0.2298851 24.7126437 11.7241379  2.1494253
BL -1.8793103 11.7241379 25.5689655  0.3965517
NH -1.9942529  2.1494253  0.3965517 13.8264368
```

Question 2

Part B

The small p-value suggests notable variations in skull sizes across different years. While the MANOVA test strongly supports rejecting the null hypothesis in favor of the alternative, the researcher now needs to provide a coherent narrative explaining why interbreeding is considered the primary factor causing these differences in skull sizes.

```
# Fit the MANOVA model
manovaModel <- manova(cbind(MB, BH, BL, NH) ~ Year, data = skullData)
summary(manovaModel, test = "Wilks")
```

```
           Df   Wilks approx F num Df den Df   Pr(>F)
Year        1 0.70431   15.219      4    145 2.06e-10 ***
Residuals 148
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Question 2

Part C

For each skull measurement (MB, BH, BL, NH), the univariate tests indicate that the "Year" variable has a statistically significant effect, signifying variations in these skull features over time. These findings provide statistical support for the idea that skull measurements change across different years.

If you reject $H_0 : \mu_1 = \mu_2 = \mu_3 = \mu_4 = \mu_5$ whenever any of the univariate ANOVAs are significant at level α, the overall probability of a Type I error is not guaranteed to be controlled at level α. This is because the Bonferroni correction (α) is a conservative adjustment. The actual overall significance level might be lower than α, which reduces the chance of a Type I error but also reduces the power of the test!

In this case, all $H_0 : \mu_1 = \mu_2 = \mu_3 = \mu_4 = \mu_5$ are $\leq \alpha$, hence the overall probability of a Type 1 error is controlled at level α

```
# Univariate ANOVA for each variable
mbAnova <- aov(MB ~ Year, data = skullData)
bhAnova <- aov(BH ~ Year, data = skullData)
blAnova <- aov(BL ~ Year, data = skullData)
nhAnova <- aov(NH ~ Year, data = skullData)

# Check the p-values for each univariate ANOVA => Pr(>F)
summary(mbAnova)
```

```
             Df Sum Sq Mean Sq F value  Pr(>F)
Year          1  491.3   491.3   23.66 2.9e-06 ***
Residuals   148 3072.6    20.8
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(bhAnova)
```

```
             Df Sum Sq Mean Sq F value Pr(>F)
Year          1    120  119.56   5.033 0.0263 *
Residuals   148   3516   23.75
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(blAnova)
```

```
             Df Sum Sq Mean Sq F value   Pr(>F)
Year          1    780   780.0   32.71 5.73e-08 ***
Residuals   148   3529    23.8
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(nhAnova)
```

```
             Df Sum Sq Mean Sq F value Pr(>F)
Year          1   44.1   44.11   4.384  0.038 *
Residuals   148 1489.2   10.06
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Question 3

Part A

For NO_2, there is no evidence to suggest that SolarRad has a significant effect. For O_3, there is some marginal evidence ($p = 0.0526$) suggesting that SolarRad might have a significant effect, but it's not strong evidence.

```r
pollutionData <- read.csv("PollutionData.csv")

# Fit the multivariate multiple regression model
model <- lm(cbind(NO2, O3) ~ Wind + SolarRad, data = pollutionData)

summary(model)
```

```
Response NO2 :

Call:
lm(formula = NO2 ~ Wind + SolarRad, data = pollutionData)

Residuals:
    Min      1Q  Median      3Q     Max
-5.7521 -2.2053 -0.5917  1.6852 10.4623

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 10.11454    3.62607   2.789  0.00813 **
Wind        -0.21129    0.33917  -0.623  0.53694
SolarRad     0.02055    0.03094   0.664  0.51042
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.416 on 39 degrees of freedom
Multiple R-squared:  0.02311,   Adjusted R-squared:  -0.02698
F-statistic: 0.4614 on 2 and 39 DF,  p-value: 0.6338


Response O3 :

Call:
lm(formula = O3 ~ Wind + SolarRad, data = pollutionData)

Residuals:
    Min      1Q  Median      3Q     Max
-7.9527 -3.5053 -0.2998  1.4703 14.7123

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  8.27619    5.58044   1.483   0.1461
Wind        -0.78682    0.52198  -1.507   0.1398
SolarRad     0.09518    0.04761   1.999   0.0526 .
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5.257 on 39 degrees of freedom
Multiple R-squared:  0.1513,    Adjusted R-squared:  0.1078
F-statistic: 3.476 on 2 and 39 DF,  p-value: 0.04082
```

Question 3

Part B

In both cases (NO_2 and O_3), the p-values for the Wind variable are greater than 0.05, indicating that there is not enough evidence to reject the null hypothesis. The conclusion is that, based on these tests, there is no significant effect of the Wind variable on either NO_2 or O_3 in the models.

Question 3

Part C

The F-tests assess the joint significance of Wind and SolarRad predictors in the model, with the following results:

- For NO2: F-statistic = 0.4614, p-value = 0.6338 (Fail to reject $ _1 = _2 = 0 $)
- For O3: F-statistic = 3.476, p-value = 0.04082 (Reject $ _1 = _2 = 0 $)

Hence, the joint test aligns with individual tests. For O3, there's a significant relationship, consistent with SolarRad evidence in part (a) and lack of Wind evidence in part (b). For NO2, both joint and individual tests suggest no significant relationship with predictors. In other words, it's inconclusive whether $\beta_1 = \beta_2 = 0$.

Question 4

Part A

It is recommended to use the correlation matrix rather than the covariance matrix. The reason for this is that PCA is sensitive to the scale of the variables, and using the correlation matrix standardizes the variables to have mean 0 and standard deviation 1.

```
trackData <- read.csv("TrackData.csv")

# Extract the columns containing distances
distanceColumns <- c("X100m.s", "X200m.s", "X400m.s", "X800m.m", "X1500m.m", "X5000m.m", "X
distanceData <- trackData[, c("Country", distanceColumns)]

# Covariance matrix S
covMatrix <- cov(distanceData[, -1])
covMatrix
```

```
              X100m.s    X200m.s    X400m.s     X800m.m   X1500m.m   X5000m.m
X100m.s    0.12350249 0.20902182 0.43069956 0.016920438 0.03836684 0.17441020
X200m.s    0.20902182 0.41557024 0.79905603 0.033115455 0.07788771 0.35913859
X400m.s    0.43069956 0.79905603 2.12290020 0.080743131 0.18974209 0.90887976
X800m.m    0.01692044 0.03311545 0.08074313 0.004055758 0.00911532 0.04406209
X1500m.m   0.03836684 0.07788771 0.18974209 0.009115320 0.02430774 0.11592929
X5000m.m   0.17441020 0.35913859 0.90887976 0.044062088 0.11592929 0.64185811
X10000m.m  0.40184545 0.81171145 2.07341549 0.100049327 0.26343721 1.41154798
Marathon.m 1.68601222 3.54620963 9.47785704 0.473903333 1.24516296 6.89104852
            X10000m.m Marathon.m
X100m.s     0.4018455  1.6860122
X200m.s     0.8117114  3.5462096
X400m.s     2.0734155  9.4778570
X800m.m     0.1000493  0.4739033
X1500m.m    0.2634372  1.2451630
X5000m.m    1.4115480  6.8910485
X10000m.m   3.2678936 15.7321815
Marathon.m 15.7321815 85.1381467
```

```
# Correlation matrix R
corMatrix <- cor(distanceData[, -1])
corMatrix
```

```
              X100m.s   X200m.s   X400m.s   X800m.m  X1500m.m  X5000m.m
X100m.s    1.0000000 0.9226384 0.8411468 0.7560278 0.7002382 0.6194618
X200m.s    0.9226384 1.0000000 0.8507270 0.8066265 0.7749513 0.6953770
X400m.s    0.8411468 0.8507270 1.0000000 0.8701714 0.8352694 0.7786139
X800m.m    0.7560278 0.8066265 0.8701714 1.0000000 0.9180442 0.8635939
X1500m.m   0.7002382 0.7749513 0.8352694 0.9180442 1.0000000 0.9281140
X5000m.m   0.6194618 0.6953770 0.7786139 0.8635939 0.9281140 1.0000000
X10000m.m  0.6325389 0.6965391 0.7872045 0.8690489 0.9346970 0.9746354
Marathon.m 0.5199490 0.5961837 0.7049905 0.8064764 0.8655492 0.9321884
           X10000m.m Marathon.m
X100m.s    0.6325389  0.5199490
```

```
X200m.s     0.6965391  0.5961837
X400m.s     0.7872045  0.7049905
X800m.m     0.8690489  0.8064764
X1500m.m    0.9346970  0.8655492
X5000m.m    0.9746354  0.9321884
X10000m.m   1.0000000  0.9431763
Marathon.m 0.9431763  1.0000000
```

Question 4

Part B

```
eigenS <- eigen(covMatrix)
eigenS
```

```
eigen() decomposition
$values
[1] 8.991362e+01 1.412626e+00 2.598442e-01 1.094203e-01 2.730060e-02
[6] 1.273280e-02 2.243554e-03 4.455645e-04

$vectors
             [,1]        [,2]         [,3]         [,4]         [,5]
[1,] -0.019865407 -0.21068958 -0.029041979 -0.358784470  0.190181784
[2,] -0.041554499 -0.35892579 -0.018390126 -0.833534544 -0.048582165
[3,] -0.110631838 -0.82786251 -0.377669011  0.396041212 -0.012020033
[4,] -0.005487699 -0.02317490  0.005341591 -0.009568087 -0.011107487
[5,] -0.014386822 -0.04465255  0.050004337 -0.015981502 -0.043222520
[6,] -0.079308444 -0.12996134  0.336448522  0.018873808 -0.909186992
[7,] -0.181098994 -0.29885393  0.848722695  0.134662690  0.364239482
[8,] -0.972787446  0.18080736 -0.141872114 -0.028425488  0.006575083
             [,6]         [,7]          [,8]
[1,]  0.886865894 -0.052444908 -0.0139585779
[2,] -0.409969944  0.062270182 -0.0037828046
[3,] -0.047663812  0.020389912 -0.0094695712
[4,] -0.007204523 -0.261227847  0.9648302746
[5,] -0.067333230 -0.959092660 -0.2622644611
[6,]  0.184076191  0.052548542 -0.0001130819
[7,] -0.068113893  0.045771467  0.0045055042
[8,]  0.003532208 -0.001055127 -0.0008700758
```

Question 4

Part C

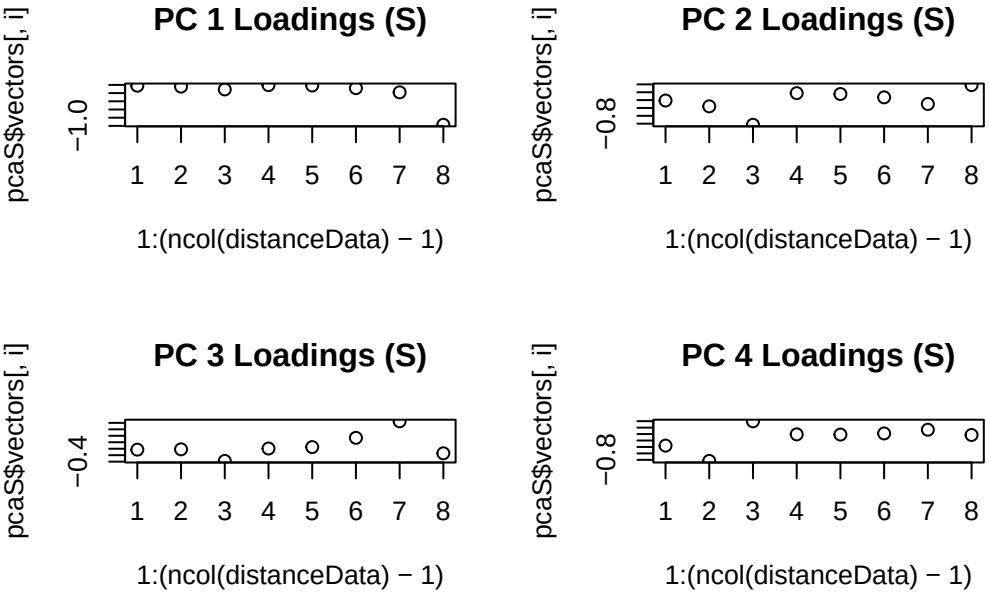
```
eigenR <- eigen(corMatrix)
eigenR
```

```
eigen() decomposition
$values
[1] 6.62214613 0.87761829 0.15932114 0.12404939 0.07988027 0.06796515 0.04641953
[8] 0.02260010

$vectors
           [,1]        [,2]       [,3]        [,4]       [,5]       [,6]
[1,] -0.3175565 -0.56687750  0.3322620 -0.12762827  0.2625555  0.5937042
[2,] -0.3369792 -0.46162589  0.3606567  0.25911576 -0.1539571 -0.6561367
[3,] -0.3556454 -0.24827331 -0.5604674 -0.65234077 -0.2183229 -0.1566252
[4,] -0.3686841 -0.01242993 -0.5324823  0.47999895  0.5400528  0.0146918
[5,] -0.3728099  0.13979665 -0.1534427  0.40451039 -0.4877151  0.1578430
[6,] -0.3643741  0.31203045  0.1897643 -0.02958755 -0.2539792  0.1412987
[7,] -0.3667726  0.30685985  0.1817517 -0.08006862 -0.1331764  0.2190168
[8,] -0.3419261  0.43896267  0.2632087 -0.29951213  0.4979283 -0.3152849
             [,7]          [,8]
[1,]  0.136241260 -0.1055416752
[2,] -0.112639528  0.0960543222
[3,] -0.002853707  0.0001272032
[4,] -0.238016094  0.0381651151
[5,]  0.610011482 -0.1392909844
[6,] -0.591298850 -0.5466969221
[7,] -0.176871021  0.7967952190
[8,]  0.398822209 -0.1581638575
```

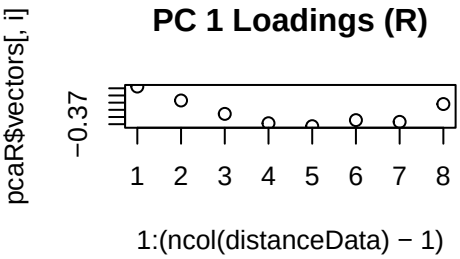
Question 4

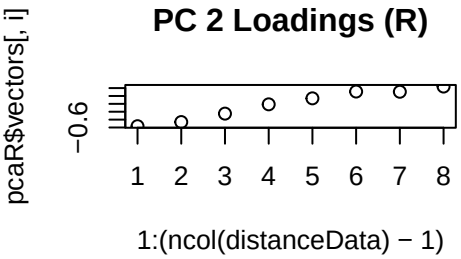
Part D

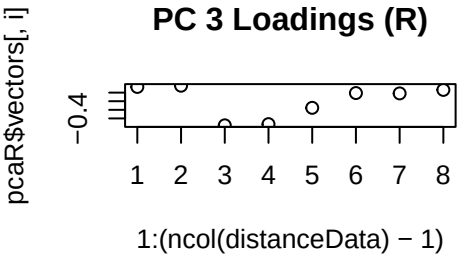
```r
# PCA using covariance matrix
pcaS <- eigenS

# PCA using correlation matrix
pcaR <- eigenR

# Plots
par(mfrow = c(2, 2))  # Setting up a 2x2 grid for subplots

# Plotting loadings for the first four principal components of S
for (i in 1:4) {
  plot(1:(ncol(distanceData) - 1), pcaS$vectors[, i],
       main = paste("PC", i, "Loadings (S)"))
}
```

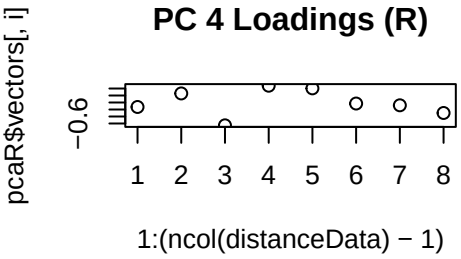



```r
# Plotting loadings for the first four principal components of R
for (i in 1:4) {
  plot(1:(ncol(distanceData) - 1), pcaR$vectors[, i],
       main = paste("PC", i, "Loadings (R)"))
}
```


PC 1 Loadings (R)
pcaR$vectors[, i]
−0.37
1 2 3 4 5 6 7 8
1:(ncol(distanceData) − 1)


PC 2 Loadings (R)
pcaR$vectors[, i]
−0.6
1 2 3 4 5 6 7 8
1:(ncol(distanceData) − 1)


PC 3 Loadings (R)
pcaR$vectors[, i]
−0.4
1 2 3 4 5 6 7 8
1:(ncol(distanceData) − 1)


PC 4 Loadings (R)
pcaR$vectors[, i]
−0.6
1 2 3 4 5 6 7 8
1:(ncol(distanceData) − 1)

Question 4

Part E

The loadings for PCA 1 for Matrix S shows a strong, positive influence on the principal component.

Question 4

Part F

So, if the first principal component's loadings form a parabola for the distance records, it means that the way these races impact the data is not just a simple straight line. There could be some tricky and not-so-straight patterns going on. This suggests we should take a closer look at how performances in these races affect the overall patterns in our data, especially when it comes to the first principal component.

Question 4

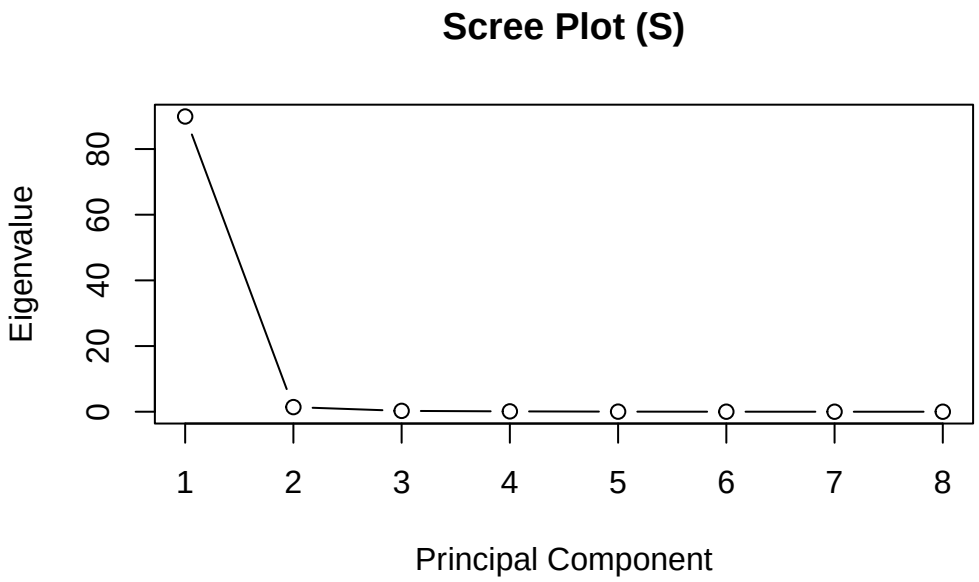
Part G

The straight line pattern in the loadings of the second principal component helps us understand how various distance records affect the overall trends we see in the data. This line gives us a clearer picture of how each type of race contributes to the patterns we're studying. It lets us look more closely at how performances differ across these distances from the dataset.

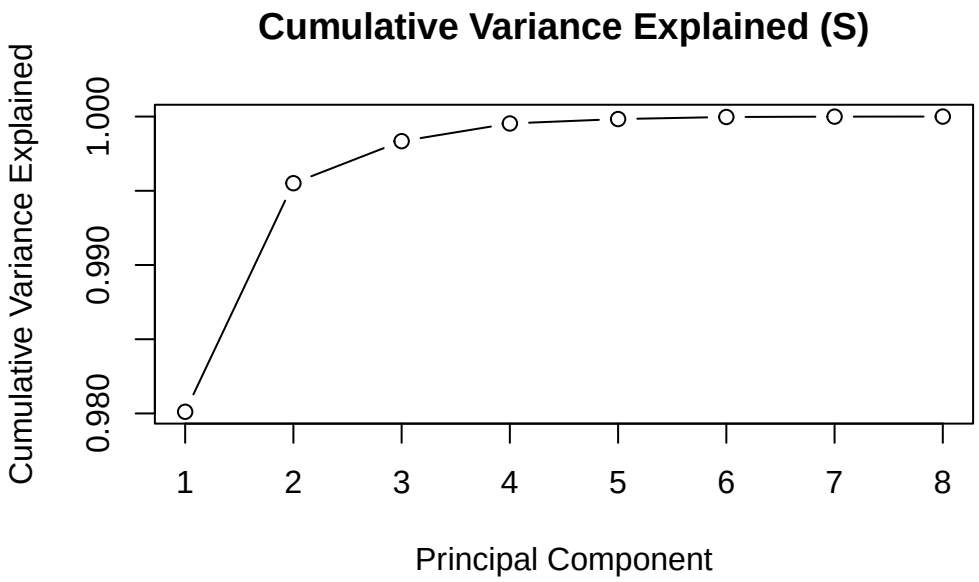
Question 4

Part H

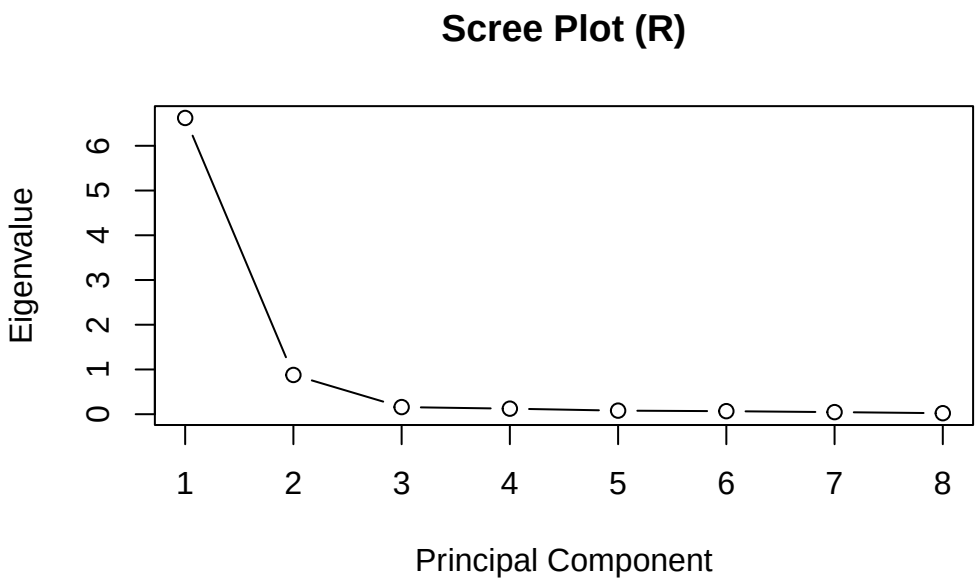
```
# Calculate cumulative variance explained
cumVarianceS <- cumsum(pcaS$values) / sum(pcaS$values)
cumVarianceR <- cumsum(pcaR$values) / sum(pcaR$values)

# Scree plot for S
plot(1:length(pcaS$values), pcaS$values, type = 'b',
     main = "Scree Plot (S)", xlab = "Principal Component",
     ylab = "Eigenvalue")
```



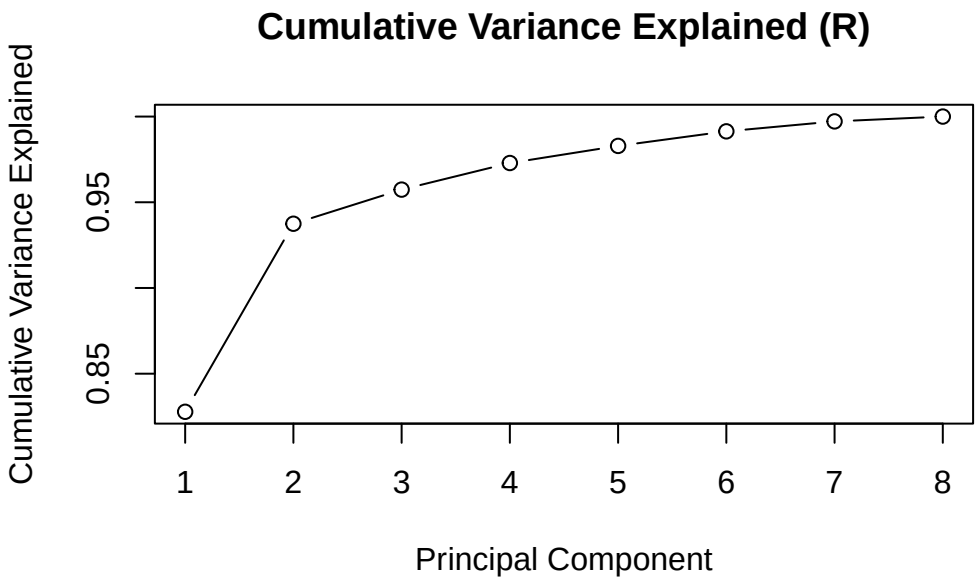

```
# Cumulative variance explained plot for S
plot(1:length(pcaS$values), cumVarianceS, type = 'b',
     main = "Cumulative Variance Explained (S)",
     xlab = "Principal Component", ylab = "Cumulative Variance Explained")
```




```r
# Scree plot for R
plot(1:length(pcaR$values), pcaR$values, type = 'b',
     main = "Scree Plot (R)", xlab = "Principal Component",
     ylab = "Eigenvalue")
```




```r
# Cumulative variance explained plot for R
plot(1:length(pcaR$values), cumVarianceR, type = 'b',
     main = "Cumulative Variance Explained (R)",
     xlab = "Principal Component", ylab = "Cumulative Variance Explained")
```


Cumulative Variance Explained (R)
Cumulative Variance Explained
0.85
0.95
1
2
3
4
5
6
7
8
Principal Component

Question 4

Part I

Checking out the scree plot, I'd stick with the first 3 components because that's where the variance stops increasing much. But, when I consider the total variance, I usually stop at the point where the components add up to around 80%. Funny enough, the first 1 component get the job done with about 98% variance. I would still pick the first 3 components as it levels off right after, which means that they are not as significant as the other components.

Question 5

Part A

```
nyseData <- read.csv("NYSEData.csv")

# Construct the sample covariance matrix S and find the sample principal components
covMatrix <- cov(nyseData)
print(covMatrix)
```

```
                   JPMorgan     Citibank   WellsFargo RoyalDutchShell
JPMorgan        4.332695e-04 0.0002756679 1.590265e-04    6.411929e-05
Citibank        2.756679e-04 0.0004387172 1.799737e-04    1.814512e-04
WellsFargo      1.590265e-04 0.0001799737 2.239722e-04    7.341348e-05
RoyalDutchShell 6.411929e-05 0.0001814512 7.341348e-05    7.224964e-04
ExxonMobil      8.896616e-05 0.0001232623 6.054612e-05    5.082772e-04
                  ExxonMobil
JPMorgan        8.896616e-05
Citibank        1.232623e-04
WellsFargo      6.054612e-05
RoyalDutchShell 5.082772e-04
ExxonMobil      7.656742e-04
```

```
pca <- princomp(covMatrix)
print(pca)
```

```
Call:
princomp(x = covMatrix)

Standard deviations:
      Comp.1       Comp.2       Comp.3       Comp.4       Comp.5
3.932325e-04 1.147326e-04 1.008194e-04 6.200912e-05 7.337776e-13

 5  variables and  5 observations.
```

Question 5

Part B

```
# Calculate the proportion of total sample variance explained by the first three principal c
propVarExplained <- cumsum(pca$sdev^2) / sum(pca$sdev^2)
print(propVarExplained[3])
```

```
   Comp.3
0.9788503
```

Question 5

Part C

- In PCA 1, JPMorgan and Citibank display positive loadings, indicating a strong positive association, while RoyalDutchShell and ExxonMobil show negative loadings, suggesting an inverse relationship with financial stocks. WellsFargo exhibits a smaller positive loading, implying a positive association with less influence compared to JPMorgan and Citibank.
- In PCA 2, Citibank and ExxonMobil have negative and positive loadings, respectively, indicating an inverse relationship between these financial and energy stocks.
- In PCA 3, JPMorgan and Citibank exhibit positive loadings, indicating a positive association, while WellsFargo has a negative loading, suggesting a contrasting movement compared to JPMorgan and Citibank. RoyalDutchShell and ExxonMobil show small loadings, indicating less influence on this component.

```
# Extract loadings for the first three components
loadings <- pca$loadings[, 1:3]
print(loadings)
```

```
                  Comp.1      Comp.2      Comp.3
JPMorgan       0.2766971  0.11679657  0.71053427
Citibank       0.1730433 -0.44557801  0.57372704
WellsFargo     0.1478641 -0.02165175 -0.20642721
RoyalDutchShell -0.6325384 -0.68636428  0.04079836
ExxonMobil     -0.6866776  0.56236471  0.34885761
```