STATS 419 Survey of Multivariate Analysis

Week 03 Assignment

Lindsey Kornowske
(lindsey.kornowske@wsu.edu)
[]

Instructor: Monte J. Shaffer

21 September 2020

1 Matrix

Create the "rotate matrix" functions as described in lectures. Apply to "myMatrix", called "mat" in my code.

```
source( paste0(local.path,"WEEK-03/functions/functions-matrix.R"), local=T);

mat <- matrix(c(1, 0, 2,0, 3, 0,4, 0, 5), nrow = 3, ncol = 3,byrow = T);
rotateMatrix90(mat)
```

```
##      [,1] [,2] [,3]
## [1,]    4    0    1
## [2,]    0    3    0
## [3,]    5    0    2
```

```
rotateMatrix180(mat)
```

```
##      [,1] [,2] [,3]
## [1,]    5    0    4
## [2,]    0    3    0
## [3,]    2    0    1
```

```
rotateMatrix270(mat)
```

```
##      [,1] [,2] [,3]
## [1,]    2    0    5
## [2,]    0    3    0
## [3,]    1    0    4
```

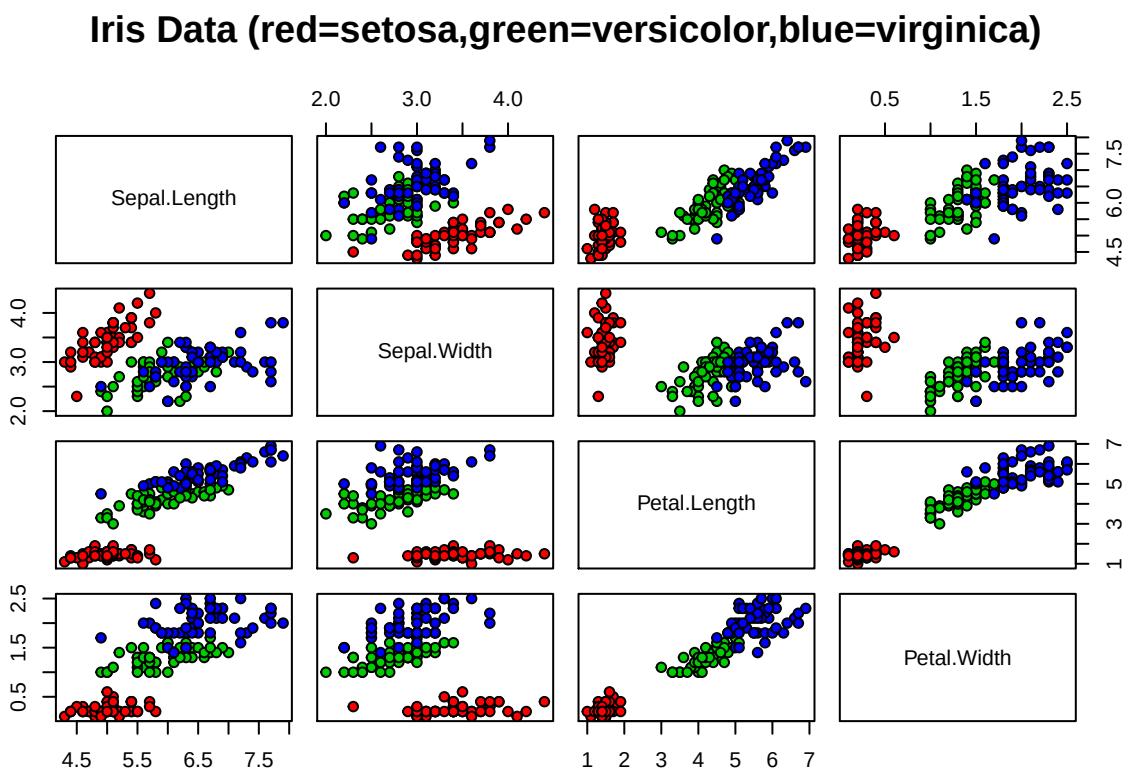
2 IRIS

Recreate the graphic for the IRIS Data Set using R. Same titles, same scales, same colors.

```
data("iris")

i<- iris

pairs(~Sepal.Length+Sepal.Width+Petal.Length+Petal.Width,
      data = i,
      pch = 21,
      bg = c("red", "green3", "blue")[unclass(i$Species)],
      main = "Iris Data (red=setosa,green=versicolor,blue=virginica)",
      cex.labels=1,
      verOdd=F,
      horOdd=F)
```



3. Write 2-3 sentences concisely definiing the IRIS dataset.

The IRIS data set is comprised of width and length measurements for the petals and sepals of three Iris species that were collected by Dr. Edgar Anderson and analyzed by Sir Ronal Aylmer Fisher for the characterization of these species (Cui). This multivariate dataset is a fixture in the data science community for the exemplification of concepts in data science and machine learning due to its combined completeness, simplicity, and inherent applications for classification (Dua and Graff 2019, Wikipedia Contributors 2020).

```
d <- read.csv("/Users/lindseykornowske/.git/STAT419/WEEK-03/functions/personality-raw.txt", header = T,
d <- d[,-3]
d<- d %>% separate(date_test, c("date","time"),sep = " ")
d$newdate <- strptime(as.character(d$date), "%m/%d/%Y")
d$newdate <- format(d$newdate, "%Y-%m-%d")
```

```
d<- d[,c(1,64,2:63)]
d <- d %>% separate(date, c("Month","Day","Year"), sep = "/")
#d$Year <- numeric(d$Year)
d$week <- strftime(d$newdate, format = "%V")
d <- d[,c(1:5,67,6:66)]
d<- d[with(d, order(Year, week, decreasing = T)), ]
d<- d[!duplicated(d$md5_email), ]

#write.csv(d, "personality-clean.txt", sep = "|")
```

Raw File: 838 observations Clean File: 678 observations

3 Summary

```
source( paste0(local.path,"WEEK-03/functions/functions-summary.R"), local=T);
sum <- doSummary(d,8,"var2","b62c73cdaf59e0a13de495b84030734e");
sum
```

```
##                    V1          V2          V3          V4          V5
## length   678.00000000 678.0000000 678.0000000 678.0000000 678.0000000
## NA Count   0.00000000   0.0000000   0.0000000   0.0000000   0.0000000
## mean       3.42005900   3.5864307   2.4359882   4.0430678   4.1775811
## median     3.40000000   3.4000000   2.6000000   4.2000000   4.2000000
## mode       3.40000000   4.2000000   1.8000000   4.2000000   4.2000000
## sd         1.02001187   1.0814641   1.0959317   0.8614014   0.7609168
## variance   1.04042422   1.1695645   1.2010663   0.7420124   0.5789944
## user       3.40000000   4.2000000   2.6000000   4.2000000   2.6000000
## z-score   -0.01966545   0.5673506   0.1496551   0.1821824  -2.0732635
##                    V6          V7          V8          V9         V10         V11
## length   678.0000000 678.0000000 678.0000000 678.0000000 678.0000000 678.000000
## NA Count   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.000000
## mean       3.8165192   3.8814159   3.1640118   4.3215339   3.8578171   2.997640
## median     4.2000000   4.2000000   3.4000000   4.2000000   4.2000000   3.400000
## mode       4.2000000   4.2000000   3.4000000   4.2000000   4.2000000   3.400000
## sd         0.9512605   0.8987648   0.9766566   0.6952738   0.8707062   1.057739
## variance   0.9048966   0.8077782   0.9538580   0.4834056   0.7581293   1.118813
## user       2.6000000   4.2000000   2.6000000   3.4000000   4.2000000   4.200000
## z-score   -1.2788496   0.3544688  -0.5774925  -1.3254260   0.3929946   1.136726
##                   V12         V13         V14         V15         V16
## length   678.0000000 678.0000000 678.0000000 678.0000000 678.0000000
## NA Count   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000
## mean       3.2831858   3.6725664   3.8306785   3.8991150   4.0890855
## median     3.4000000   3.4000000   4.2000000   4.2000000   4.2000000
## mode       3.4000000   4.2000000   4.2000000   4.2000000   4.2000000
## sd         0.9686583   0.9235710   1.0356153   0.8256410   0.7537393
## variance   0.9382988   0.8529834   1.0724991   0.6816831   0.5681229
## user       3.4000000   3.4000000   4.2000000   5.0000000   3.4000000
## z-score    0.1205938  -0.2951223   0.3566204   1.3333700  -0.9142226
##                   V17         V18         V19         V20         V21
## length   678.0000000 678.0000000 678.0000000 678.0000000 678.0000000
## NA Count   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000
## mean       4.3038348   4.0749263   3.4849558   2.3415929   3.0247788
```

```
## median     4.2000000   4.2000000   3.4000000   1.8000000   3.4000000
## mode       4.2000000   4.2000000   3.4000000   1.8000000   3.4000000
## sd         0.7175064   0.7762670   0.8806357   0.9646079   1.1199703
## variance   0.5148154   0.6025905   0.7755193   0.9304684   1.2543334
## user       5.0000000   3.4000000   1.8000000   2.6000000   2.6000000
## z-score    0.9702564  -0.8694512  -1.9133402   0.2678882  -0.3792768
##                 V22          V23         V24         V25         V26
## length   678.0000000 678.00000000 678.0000000 678.000000 678.0000000
## NA Count   0.0000000   0.00000000   0.0000000   0.000000   0.0000000
## mean       3.8436578   4.17286136   3.7917404   2.414749   4.1398230
## median     4.2000000   4.20000000   4.2000000   2.600000   4.2000000
## mode       4.2000000   4.20000000   4.2000000   1.800000   4.2000000
## sd         0.9087038   0.79835563   0.8747515   1.133310   0.8504971
## variance   0.8257426   0.63737171   0.7651902   1.284391   0.7233453
## user       2.6000000   4.20000000   3.4000000   5.000000   2.6000000
## z-score   -1.3686064   0.03399318  -0.4478305   2.281151  -1.8104977
##                 V27         V28         V29         V30         V31
## length   678.0000000 678.0000000 678.0000000 678.0000000 678.0000000
## NA Count   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000
## mean       3.6230088   3.5557522   2.9539823   3.8082596   3.8707965
## median     3.4000000   3.4000000   2.6000000   4.2000000   4.2000000
## mode       4.2000000   4.2000000   2.6000000   4.2000000   4.2000000
## sd         0.8621716   1.0133762   1.0700738   0.7943242   0.9586747
## variance   0.7433398   1.0269314   1.1450580   0.6309509   0.9190573
## user       4.2000000   3.4000000   2.6000000   2.6000000   4.2000000
## z-score    0.6692301  -0.1536963  -0.3308018  -1.5211165   0.3433944
##                 V32         V33         V34         V35          V36
## length   678.0000000 678.0000000 678.0000000 678.0000000 678.00000000
## NA Count   0.0000000   0.0000000   0.0000000   0.0000000   0.00000000
## mean       3.6123894   3.9994100   3.8070796   3.5014749   4.18938053
## median     3.4000000   4.2000000   4.2000000   3.4000000   4.20000000
## mode       4.2000000   4.2000000   4.2000000   3.4000000   4.20000000
## sd         0.9964012   0.8660337   0.9024225   0.9996591   0.74231020
## variance   0.9928153   0.7500144   0.8143663   0.9993184   0.55102443
## user       1.8000000   3.4000000   4.2000000   4.2000000   4.20000000
## z-score   -1.8189355  -0.6921324   0.4354062   0.6987633   0.01430597
##                 V37          V38         V39         V40         V41
## length   678.0000000 678.00000000 678.0000000 678.0000000 678.0000000
## NA Count   0.0000000   0.00000000   0.0000000   0.0000000   0.0000000
## mean       3.1227139   4.14218289   4.0336283   3.9439528   3.7174041
## median     3.4000000   4.20000000   4.2000000   4.2000000   4.2000000
## mode       3.4000000   4.20000000   4.2000000   4.2000000   4.2000000
## sd         1.1160374   0.84059974   0.8661987   0.9014269   1.0473111
## variance   1.2455394   0.70660793   0.7503003   0.8125705   1.0968606
## user       2.6000000   4.20000000   2.6000000   4.2000000   4.2000000
## z-score   -0.4683659   0.06878078  -1.6550801   0.2840465   0.4607951
##                 V42         V43         V44         V45         V46
## length   678.0000000 678.0000000 678.0000000 678.0000000 678.0000000
## NA Count   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000
## mean       3.8200590   3.7185841   3.7067847   3.8613569   3.9522124
## median     4.2000000   3.4000000   3.4000000   4.2000000   4.2000000
## mode       4.2000000   4.2000000   4.2000000   4.2000000   4.2000000
## sd         0.9290703   0.8677258   0.9448209   0.8484651   0.9218597
## variance   0.8631716   0.7529481   0.8926865   0.7198930   0.8498252
```

```
## user       4.2000000   4.2000000   2.6000000   4.2000000   4.2000000
## z-score    0.4089475   0.5548019  -1.1714227   0.3991243   0.2687910
##                  V47         V48         V49         V50         V51         V52
## length   678.0000000 678.0000000 678.0000000 678.0000000 678.000000 678.0000000
## NA Count   0.0000000   0.0000000   0.0000000   0.0000000   0.000000   0.0000000
## mean       3.9451327   3.5486726   3.7610619   3.3551622   3.568732   3.9687316
## median     4.2000000   3.4000000   4.2000000   3.4000000   3.400000   4.2000000
## mode       4.2000000   4.2000000   4.2000000   3.4000000   3.400000   4.2000000
## sd         0.8757010   1.0356523   0.9515998   1.0289150   1.032112   0.8649221
## variance   0.7668522   1.0725758   0.9055421   1.0586661   1.065254   0.7480903
## user       2.6000000   3.4000000   2.6000000   4.2000000   1.800000   4.2000000
## z-score   -1.5360640  -0.1435545  -1.2201158   0.8210958  -1.713702   0.2673864
##                  V53         V54         V55         V56         V57         V58
## length   678.0000000 678.0000000 678.0000000 678.0000000 678.000000 678.0000000
## NA Count   0.0000000   0.0000000   0.0000000   0.0000000   0.000000   0.0000000
## mean       4.2814159   4.1032448   3.3032448   3.8283186   3.783481   3.9262537
## median     4.2000000   4.2000000   3.4000000   4.2000000   4.200000   4.2000000
## mode       5.0000000   4.2000000   3.4000000   4.2000000   4.200000   4.2000000
## sd         0.8088011   0.8375713   0.9868058   0.9329171   0.864852   0.9227093
## variance   0.6541593   0.7015256   0.9737856   0.8703342   0.747969   0.8513924
## user       2.6000000   3.4000000   4.2000000   4.2000000   1.800000   4.2000000
## z-score   -2.0788990  -0.8396239   0.9087454   0.3984078  -2.293434   0.2966767
##                  V59         V60
## length   678.0000000 678.0000000
## NA Count   0.0000000   0.0000000
## mean       3.7327434   3.6879056
## median     4.2000000   3.4000000
## mode       4.2000000   4.2000000
## sd         0.9871053   0.9868566
## variance   0.9743768   0.9738860
## user       2.6000000   4.2000000
## z-score   -1.1475406   0.5189147
```

3.1 Variance

3.1.1 Naive

```
doSampleVariance(d[,21], "naive")
```

```
## [1] 1.072499
```

3.1.2 Two Pass

```
doSampleVariance(d[,21], "2pass")
```

```
## [1] 1.072499
```

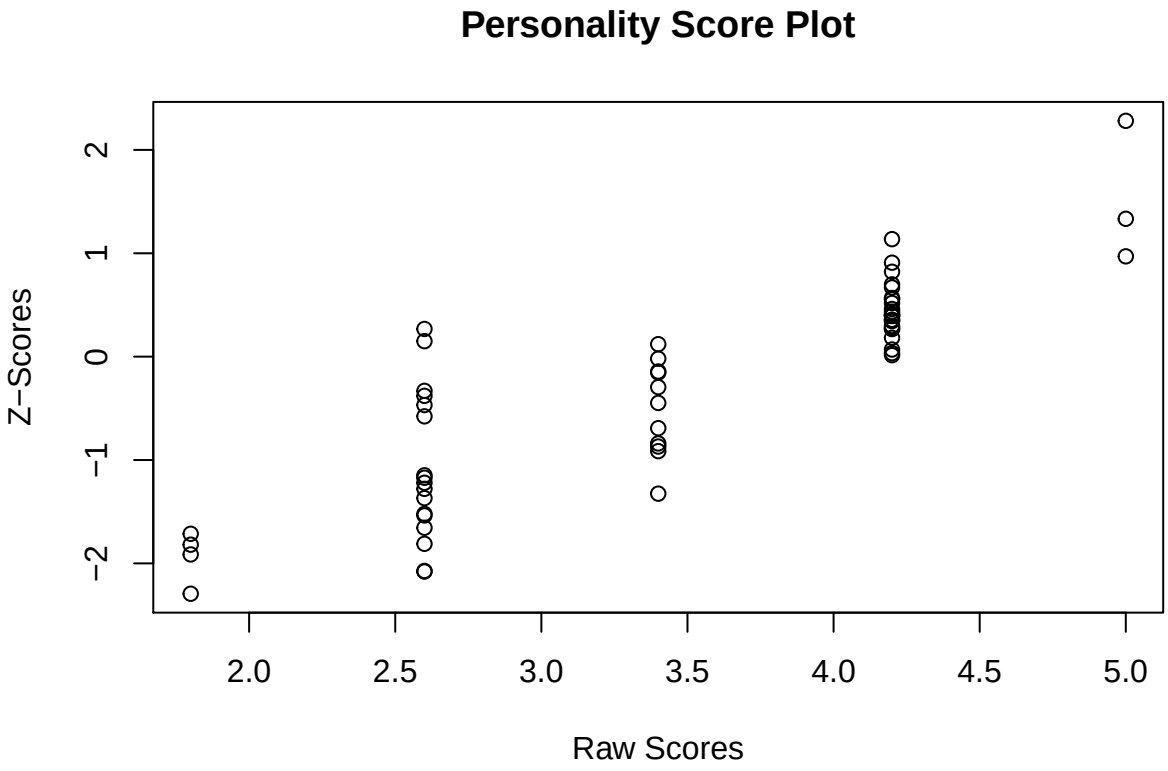
3.2 Mode

```
doMode(d[,21])
```

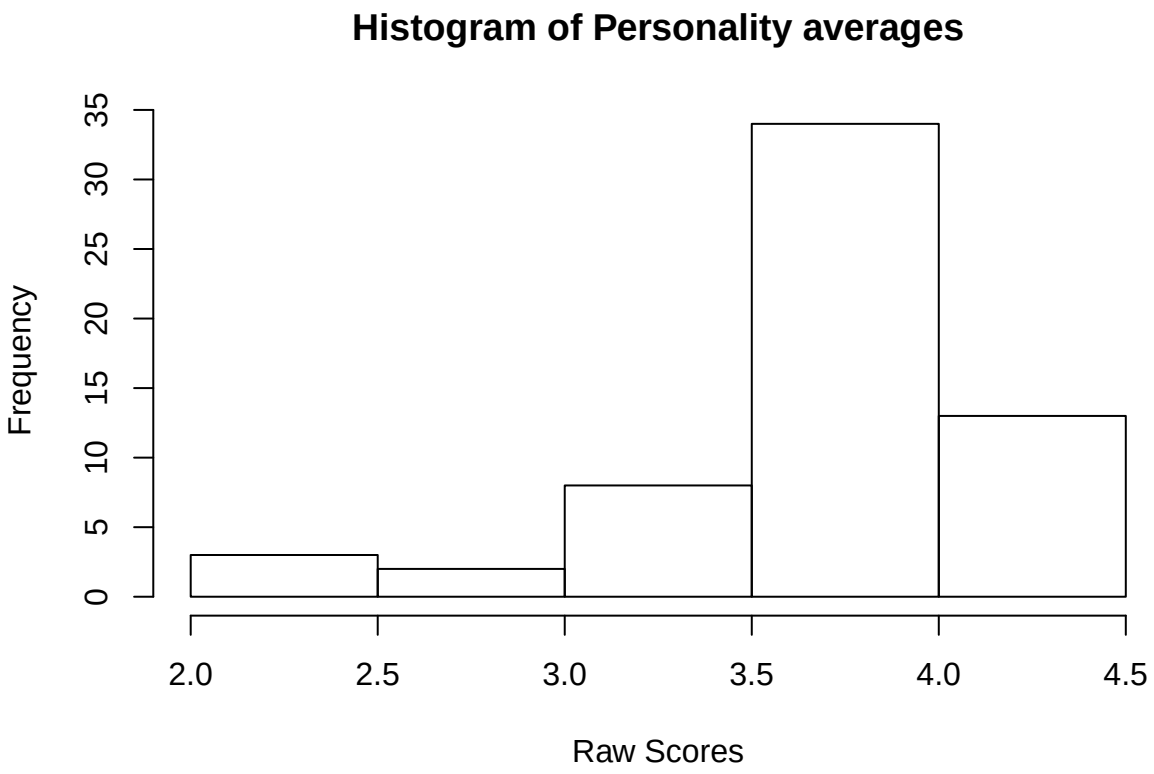
```
## [1] 4.2
```

3.3 Visualize

```
sum <- t(sum)
plot(sum[,8], sum[,9], main = "Personality Score Plot", xlab = "Raw Scores", ylab= "Z-Scores")
```




```
hist(sum[,3], main = "Histogram of Personality averages", xlab = "Raw Scores")
```



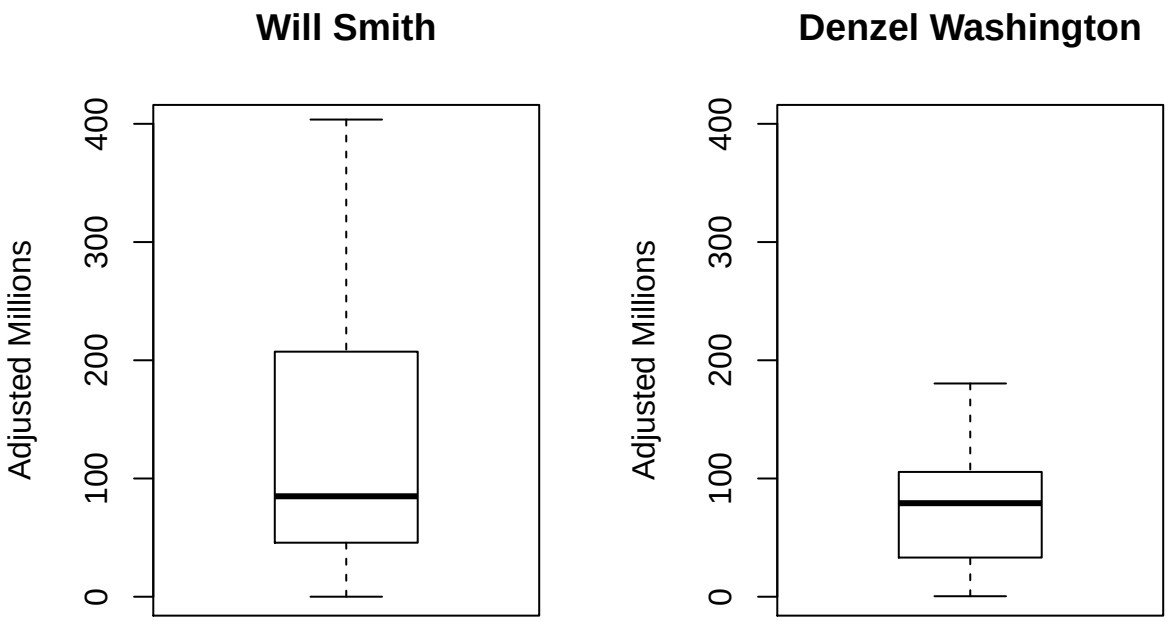
The histogram of the scoring averages for the entire personality dataset shows the highest frequency score falls around 4.0. This follows a pattern I have also observed in my sensory research where panelists have central tendencies regarding their use of scale because the extremes feel too definite. The plot comparing the z-scores and raw scores for the monte.shaffer@gmail.com data shows a distribution around each of the scores, which is more or less expected because there will be variation in the score an individual selects relative to the population. However, as we might also expect given the histogram averages, the z-scores for the raw scores near 4.5 have the greatest density of points near 0, or the center of the distribution.

4 Denzel vs. Will

6. Compare Will Smith and Denzel Washington. [See 03_n greater 1-v2.txt for the necessary functions and will-vs-denzel.txt for some sample code and in DROPBOX:You will have to create a new variable millions.2000 that converts each movie's millions based on the $year of the movie, so all dollars are in the same time frame. You will need inflation data from about 1980-2020 to make this work.

4.1 Movie Earnings Adjusted for Inflation

```
will = grabFilmsForPerson("nm0000226");
denzel = grabFilmsForPerson("nm0000243");
will.adjust <- inflationAdjust(will)
denzel.adjust <- inflationAdjust(denzel)
par(mfrow=c(1,2));
boxplot(will.adjust, main=will$name, ylim=c(0,400), ylab="Adjusted Millions" );
boxplot(denzel.adjust, main=denzel$name, ylim=c(0,400), ylab="Adjusted Millions" );
```

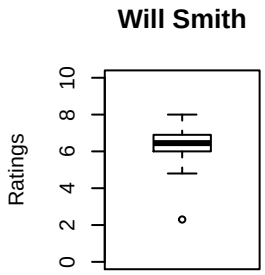


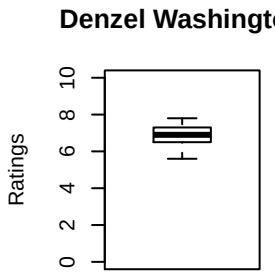
4.2 Additional Metrics Comparison

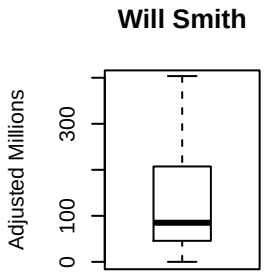
7. Build side-by-side box plots on several of the variables (including #6) to compare the two movie stars. After each box plot, write 2+ sentence describing what you are seeing, and what conclusions you can logically make. You will need to review what the box plot is showing with the box portion, the divider in the box, and the whiskers.

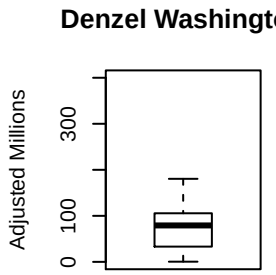
The median rating is higher for Denzel's movies than Will's movies. Will's movies have higher maximum and lower minimum ratings. The median adjusted millions are similar for both actors, however the highest earning Will Smith movie earned nearly double the highest earning Denzel movie and the 3rd quartile skews higher for the Will Smith movies as well. The median duration of each actor's movies is also similar, with the exception of one outlier by Denzel lasting nearly 200 minutes. The median production year for Denzel's movies falls before 2000, almost a decade before the median year for Will Smith's movies. This same pattern exists for the earliest movies Denzel and Will made. From these data, we can see that Denzel has had a longer career than Will Smith, but Will Smith's movies have made more money.

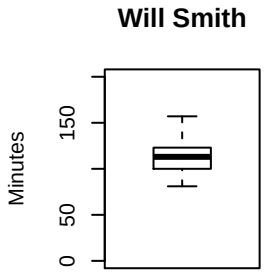
```
par(mfrow=c(2,4));
boxplot(will$movies.50$ratings, main=will$name, ylim = c(0,10), ylab="Ratings");
boxplot(denzel$movies.50$ratings, main=denzel$name, ylim = c(0,10), ylab="Ratings");
boxplot(will.adjust, main=will$name, ylim=c(0,400), ylab="Adjusted Millions" );
boxplot(denzel.adjust, main=denzel$name, ylim=c(0,400), ylab="Adjusted Millions" );
boxplot(will$movies.50$minutes, main=will$name, ylim = c(0,200), ylab="Minutes");
boxplot(denzel$movies.50$minutes, main=denzel$name, ylim = c(0,200), ylab="Minutes");
boxplot(will$movies.50$year, main=will$name, ylim = c(1980,2022), ylab="Year");
boxplot(denzel$movies.50$year, main=denzel$name, ylim = c(1980,2022), ylab="Year");
```

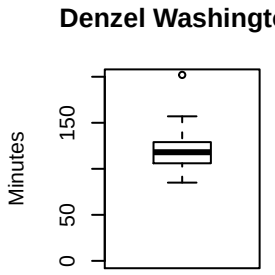

Will Smith
Ratings
0
2
4
6
8
10

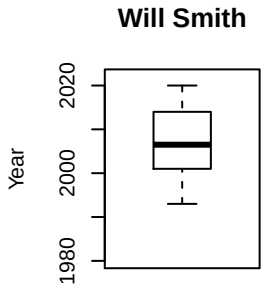

Denzel Washington
Ratings
0
2
4
6
8
10

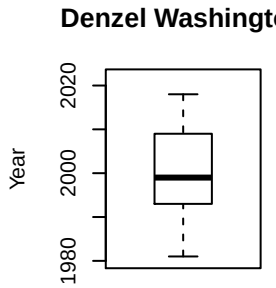

Will Smith
Adjusted Millions
0
100
300


Denzel Washington
Adjusted Millions
0
100
300


Will Smith
Minutes
0
50
150


Denzel Washington
Minutes
0
50
150


Will Smith
Year
1980
2000
2020


Denzel Washington
Year
1980
2000
2020