HW1

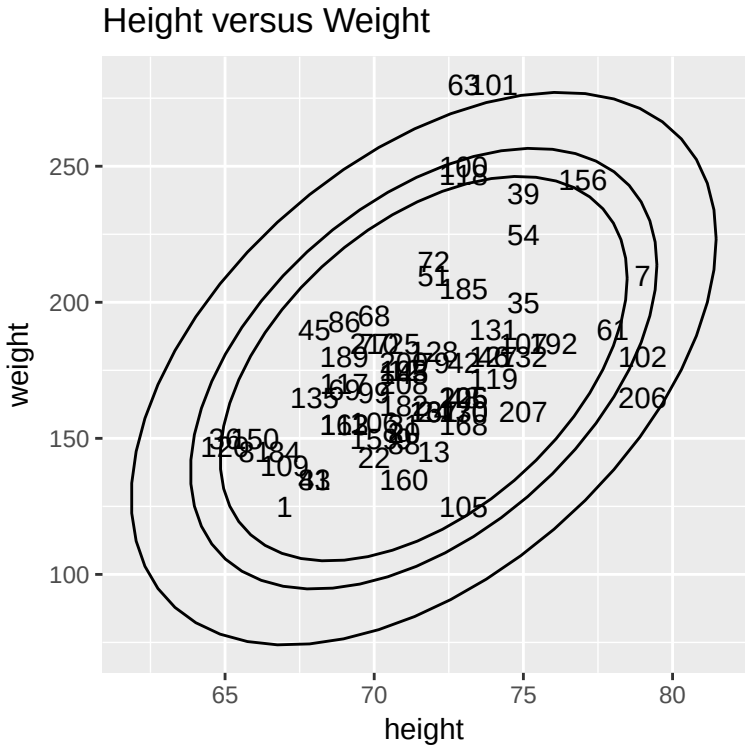
Due: 1/25/2018

Naomi Giertych

Question 3

Part a

I fit a 2-dimensional Gaussian on the male data contained in the height and weight dataset. Below is a scatterplot of the male data with a 99%, a 95%, and a 90% confidence level ellipse of the 2-dim Guassian superimposed. The 95% confidence level does well in fitting the data; only missing a few of the points. It is interesting to note that individuals 63 and 101 (index is from the original dataset) are outliers in the entire male dataset with near average heights and way above average weights. These two individuals have heights of 73 and 74 inches, respectively, and both weigh 280 lbs.

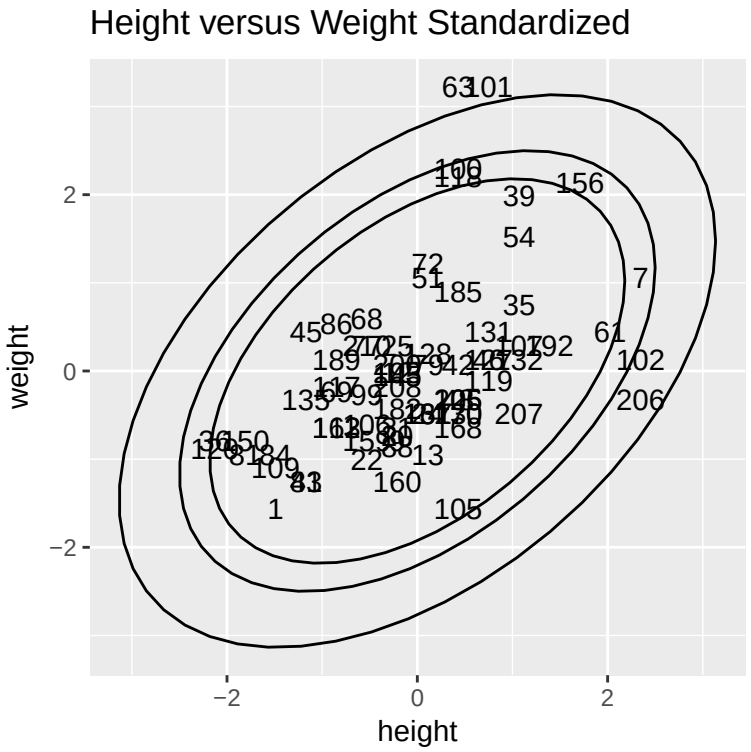



```
##     height weight
## 63      73    280
## 101     74    280
```

Part b

I standardize the male only height and weight data by centering each by their means and then divided by their standard deviations. Below is a scatterplot of the standardized male data with a 99%, a 95%, and a 90% confidence level ellipse of the 2-dim Guassian (of the standardized data) superimposed. It looks pretty much the same as the non-standardizd graph except now it is centered at zero. It is easier to determine relative distance of individuals after centering and standardizing the data; however the units no longer make

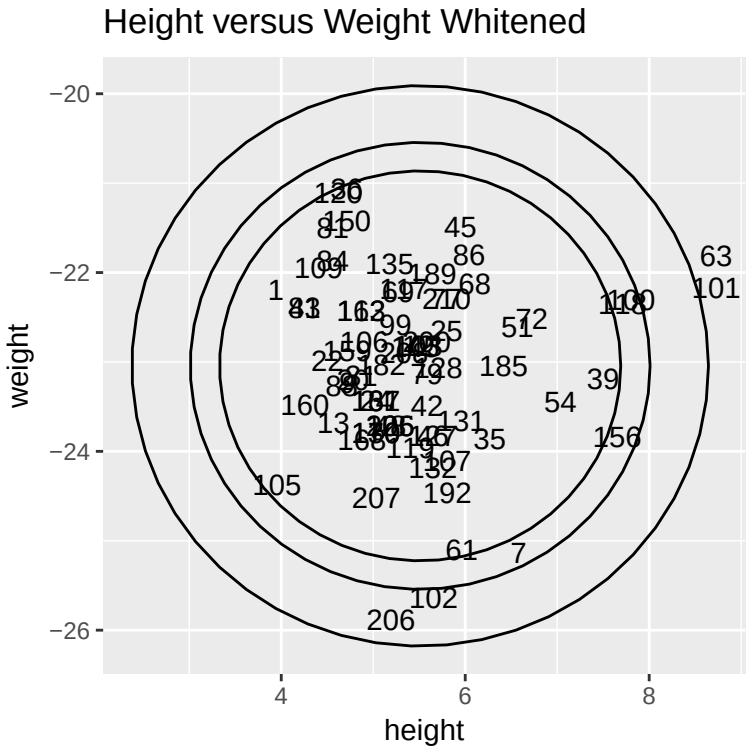
sense. Note that it is much easier to confirm that individuals 63 and 101 have average heights but above average weights.



Part c

Finally, I whiten the data by multiplying the male data by the inverse, square root of the eigenvalues and the transpose of the eigenvectors. The ellipses are circles and the data points have moved around the graph. Unlike the original data, the units of the whitened data no longer make sense, but similar to the standardized data we can interpret individuals relative to the group or other individuals. For instance, our two outliers from parts a and b have moved to above average "height" and slightly above average "weight". Therefore, compared to their counterparts, they are much taller, but they are only slightly overweight for their height.

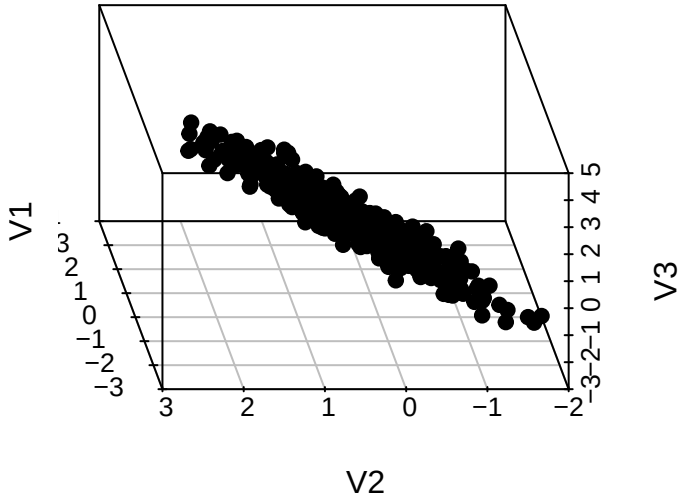
```
##      height          weight       
##  Min.   :3.944   Min.   :-25.88  
##  1st Qu.:4.871   1st Qu.:-23.71  
##  Median :5.333   Median :-23.03  
##  Mean   :5.510   Mean   :-23.04  
##  3rd Qu.:5.804   3rd Qu.:-22.30  
##  Max.   :8.729   Max.   :-21.06
```

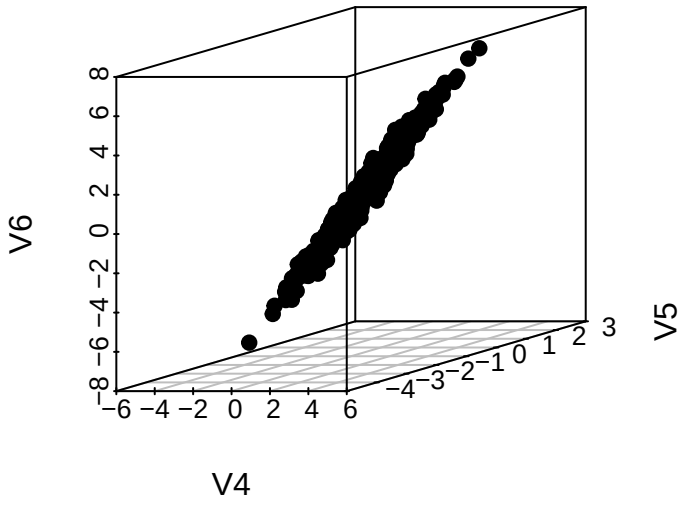


Question 4

Part a

Plotting the fa-data in a couple of 3-D graphs reveals that the data seems to lie in a 2 dimensional space of some linear combination of the all of the variables.





Part b

To identify the principal components and projections of the data, I standardize the data and then determine the eigenvalues and eigenvectors of the covariance matrix of all the data. Below is a list of the corresponding eigenvalues and then the eigenvectors. The eigenvectors tell me which linear combinations of the variables are orthoganal to each other.

```
##          [,1]     [,2]       [,3]     [,4]       [,5]       [,6]
## [1,] 3.599036 0.000000 0.00000000 0.000000 0.00000000 0.00000000
## [2,] 0.000000 3.263089 0.00000000 0.000000 0.00000000 0.00000000
## [3,] 0.000000 0.000000 0.05461724 0.000000 0.00000000 0.00000000
## [4,] 0.000000 0.000000 0.00000000 0.037706 0.00000000 0.00000000
## [5,] 0.000000 0.000000 0.00000000 0.000000 0.02505072 0.00000000
## [6,] 0.000000 0.000000 0.00000000 0.000000 0.00000000 0.01589523
## [7,] 0.000000 0.000000 0.00000000 0.000000 0.00000000 0.00000000
##           [,7]
## [1,] 0.00000000
## [2,] 0.00000000
## [3,] 0.00000000
## [4,] 0.00000000
## [5,] 0.00000000
## [6,] 0.00000000
## [7,] 0.00460539
```

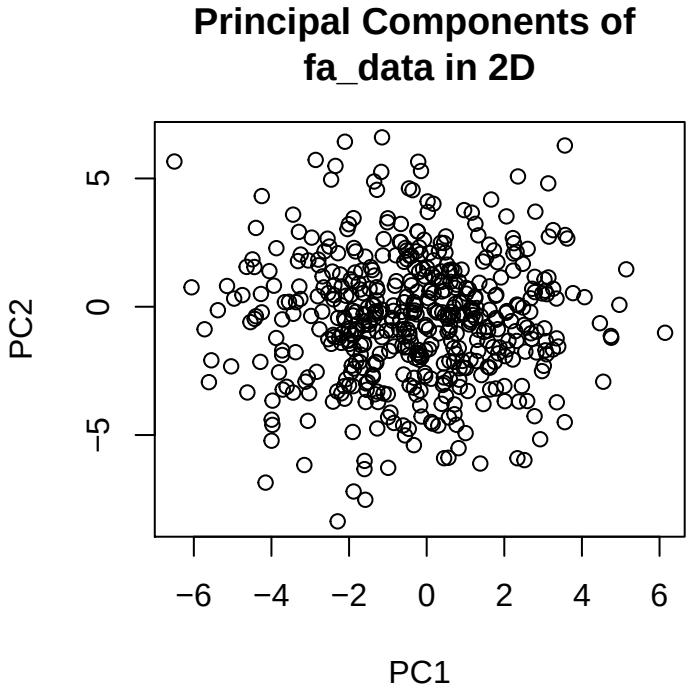
```
##            [,1]        [,2]        [,3]        [,4]        [,5]
## [1,] -0.1652398 -0.51538946  0.74212848  0.32797787 -0.09065718
## [2,] -0.4729406 -0.22352554 -0.58050190  0.57807168 -0.14380685
## [3,] -0.4900225  0.19052032  0.13618456 -0.01085282  0.79715291
## [4,]  0.2465241 -0.48715719 -0.14720228 -0.13943619  0.02684453
## [5,] -0.5191586 -0.05030812 -0.00199702 -0.66381507 -0.34762137
## [6,] -0.2895707 -0.45960906 -0.09459417 -0.24661995  0.06991978
## [7,]  0.3078149 -0.44386769 -0.25118932 -0.19295482  0.45738177
##             [,6]        [,7]
## [1,]  0.19734712  0.03975854
## [2,]  0.15616721  0.10072538
## [3,] -0.09050015  0.24753355
## [4,] -0.40957410  0.70165604
## [5,]  0.33932406  0.22651185
```

```
## [6,] -0.52510479 -0.59541169
## [7,]  0.60815199 -0.16987319
```

The two eigenvectors explain approximately 98% of the variation in the data (based on the ratio of the first two eigenvalues to the sum of all the eigenvalues.)

```
## [1] 0.9803036
```

Below is a graph of the data using two principal components. The data is now reduced to 2 dimensions.



Question 5

I read in the vehical MPG dataset and cleaned it. To clean the dataset, I removed "?" marks in the "horsepower", and I converted "cylinders", "model year", and "origin" to be factors.

As can be seen by the summary table in part a, all the cars were made during the 70s and early 80s. Examining the data set, we can see that most cars were made in the U.S. (origin 1), followed by Europe (origin 2), and then Japan (origin 3).

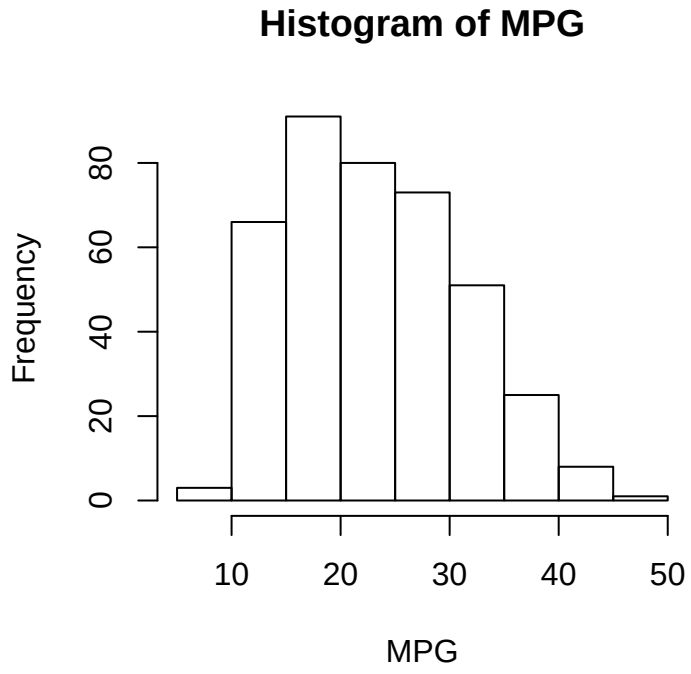
Part a

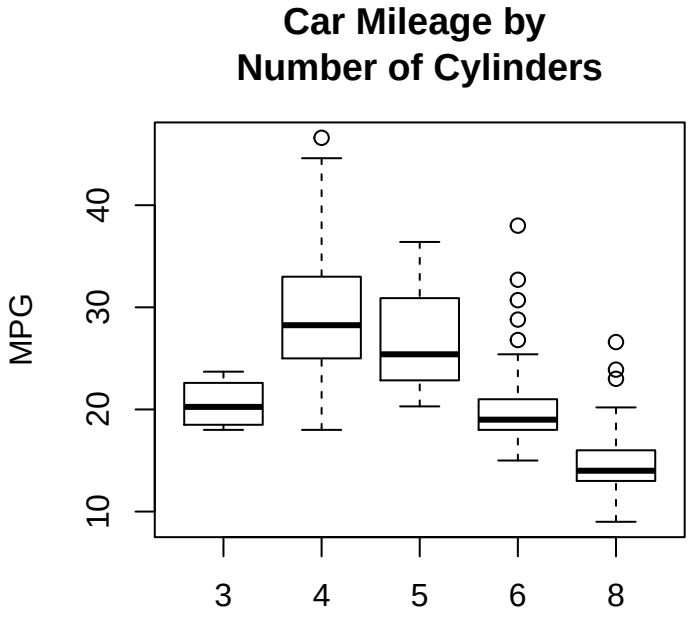
As can be seen by the summary statistics below, the magnitutde of units of the variables differs quite a bit. Miles per gallon and acceleration (m/s^2 from 0 to 60 mph) is always less than 50, horsepower is between 46 and 230, displacement (total volume of all the cyclinders in the engine) is between 68 and 455 cubic centimeters. However, weight is in the 1,000's of pounds. There appears to be some potential outliers based on the maximums of the numeric variables. I examine this further below.

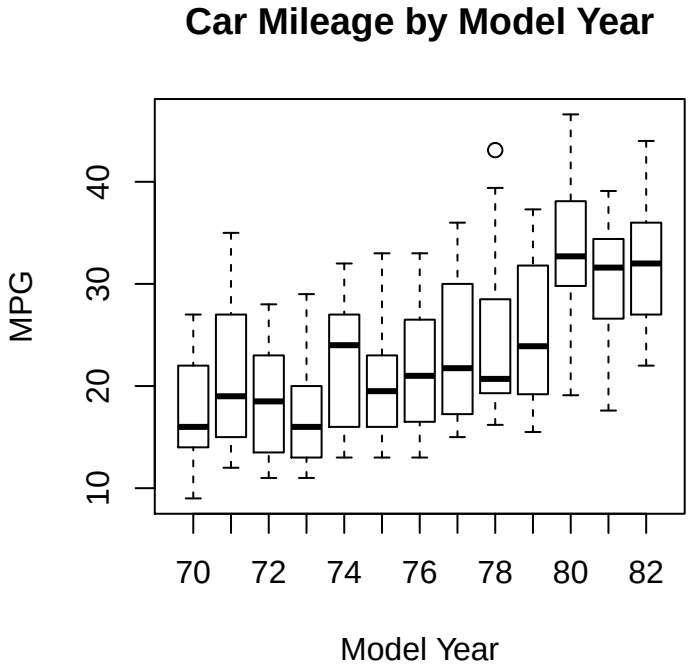
```
##       mpg        cylinders  displacement     horsepower        weight
##  Min.   : 9.00   3:  4     Min.   : 68.0   Min.   : 46.0   Min.   :1613
##  1st Qu.:17.50   4:204     1st Qu.:104.2   1st Qu.: 75.0   1st Qu.:2224
##  Median :23.00   5:  3     Median :148.5   Median : 93.5   Median :2804
##  Mean   :23.51   6: 84     Mean   :193.4   Mean   :104.5   Mean   :2970
```

```
##  3rd Qu.:29.00   8:103     3rd Qu.:262.0   3rd Qu.:126.0   3rd Qu.:3608
##  Max.   :46.60             Max.   :455.0   Max.   :230.0   Max.   :5140
##                                            NA's   :6
##   acceleration     model.year  origin           car.name
##  Min.   : 8.00   73     : 40   1:249   ford pinto    :  6
##  1st Qu.:13.82   78     : 36   2: 70   amc matador   :  5
##  Median :15.50   76     : 34   3: 79   ford maverick :  5
##  Mean   :15.57   82     : 31           toyota corolla:  5
##  3rd Qu.:17.18   75     : 30           amc gremlin   :  4
##  Max.   :24.80   70     : 29           amc hornet    :  4
##                  (Other):198           (Other)       :369
```

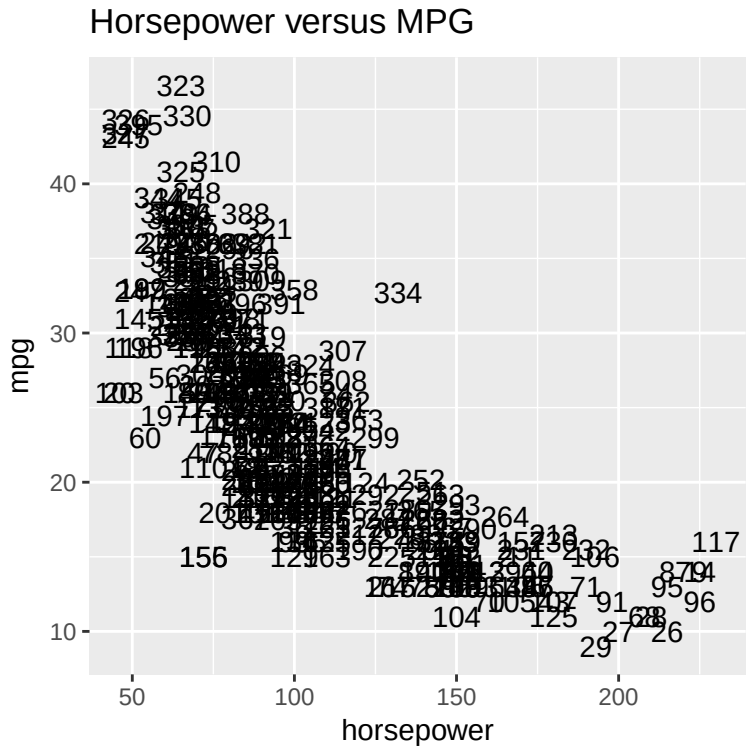
The histogram of the MPG suggests that there are no outliers within the entire dataset. However, if we control for the number of cylinders in the engine, there are quite a few outliers, but again, only one oultier if we control for the model year.



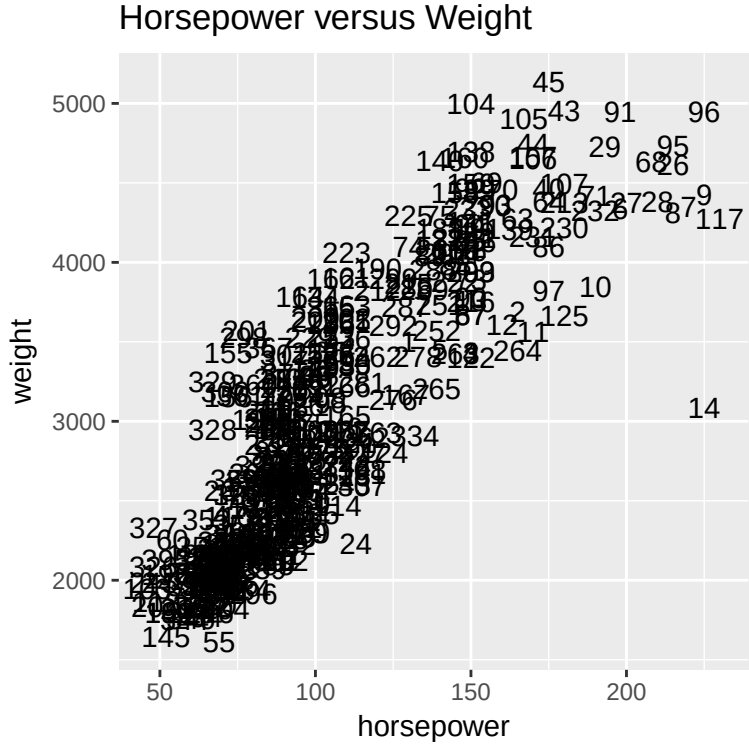




It is also interesting to graph MPG, weight, acceleration by horsepower. We note the (quadratic) decrease in MPG and acceleration as horsepower increases, but the positive relationship between weight and horsepower (possibly due to increases in engine size). There also appears to be an outlier with moderate horsepower but fairly high MPG (the Japanese Datsun), an outlier with high horsepower and moderate weight (the American Buick Estate Wagon), and an outlier with high horsepower and moderate to high acceleration (the American HI 1200d).

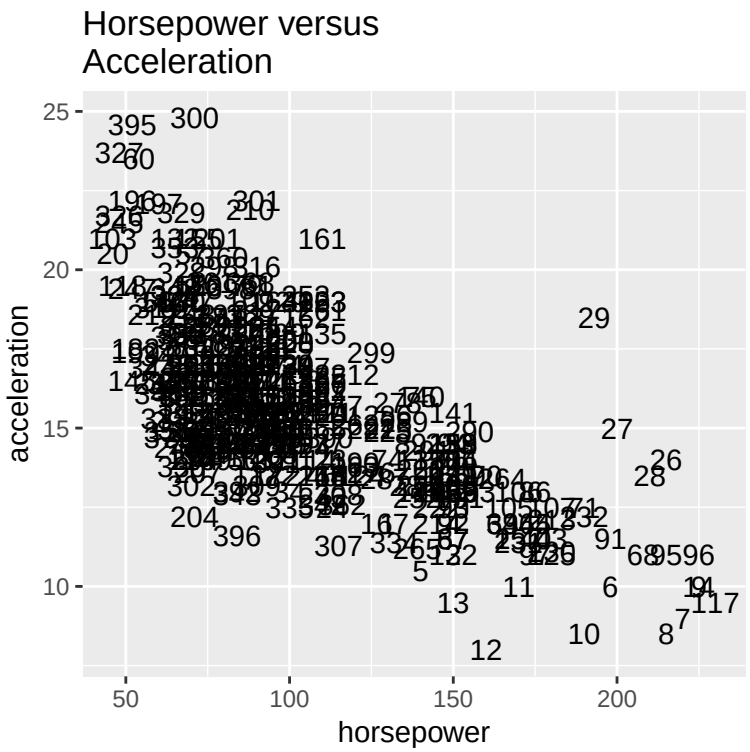



```
##      mpg cylinders displacement horsepower weight acceleration model.year
## 334 32.7         6          168        132   2910         11.4         80
##     origin      car.name
## 334      3 datsun 280-zx
```




```
##    mpg cylinders displacement horsepower weight acceleration model.year
## 14  14         8          455        225   3086           10         70
```

```
##    origin                car.name
## 14      1 buick estate wagon (sw)
```




```
##    mpg cylinders displacement horsepower weight acceleration model.year
## 29   9         8          304        193   4732         18.5         70
##    origin car.name
## 29      1 hi 1200d
```

Part b and c

The variables I include for PCA analysis are MPG, displacement, horsepower, weight and acceleration. Before performing PCA, I re-center each of the variables to 0 (by subtracting the mean). I then perform PCA using covariates and then correlation.

Below is a summary of the covariate PCA results. Based on these results, I would choose only one principal component since the first principal component explains 99.76% of the variance in the data.

```
## Importance of components:
##                             Comp.1       Comp.2       Comp.3       Comp.4
## Standard deviation     854.5889090 38.851346159 1.613893e+01 4.215858e+00
## Proportion of Variance   0.9975543  0.002061741 3.557719e-04 2.427694e-05
## Cumulative Proportion    0.9975543  0.999615997 9.999718e-01 9.999960e-01
##                              Comp.5
## Standard deviation     1.701526e+00
## Proportion of Variance 3.954569e-06
## Cumulative Proportion  1.000000e+00
```

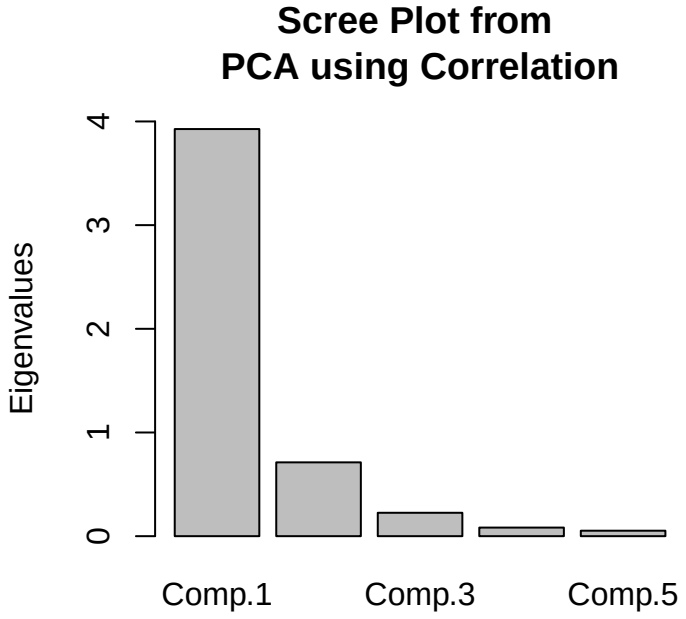
Below is a summary of the correlation PCA results. Based on these results, I would choose two principal components since the first and second principal components explain 92.77% of the variance in the data.

```
## Importance of components:
##                           Comp.1    Comp.2     Comp.3     Comp.4
```

```
## Standard deviation     1.9816040 0.8438043 0.47499662 0.28788096
## Proportion of Variance 0.7853509 0.1424011 0.04512436 0.01657509
## Cumulative Proportion  0.7853509 0.9277520 0.97287636 0.98945145
##                            Comp.5
## Standard deviation     0.22965790
## Proportion of Variance 0.01054855
## Cumulative Proportion  1.00000000
```

Of the correlation and covariate methods, I choose the correlation method to proceed because the correlation method accounts for the fact that some of the variables (particularly weight) are on a different scale compared to the other variables.

The scree plot below confirms that I should use two principal components from the correlation PCA.



Part d

The factors are the linear combinations of the original variables that capture the dimensions of the data (analogous to eigenvectors). The variable loadings tell me the weights of each variable that was used to construct each factor; they are the correlation between the original variables and the factors.

Examining the factor loadings for the correlation PCA reveals that the weights are approximately evenly distributed amongst MPG, displacement, horsepower, weight, and acceleration in the first component. There is slightly more weight attributed to displacement and horsepower in the first component. A potential interpretation is that MPG, displacement, horsepower, weight, and acceleration equally determine car performance or car preferences. The second component has significantly more weight on acceleration and slightly more weight on MPG and weight. A potential interpretation is that if the MPG, displacement, horsepower, and weights of two cars are similar, then the defining factor between them is acceleration.

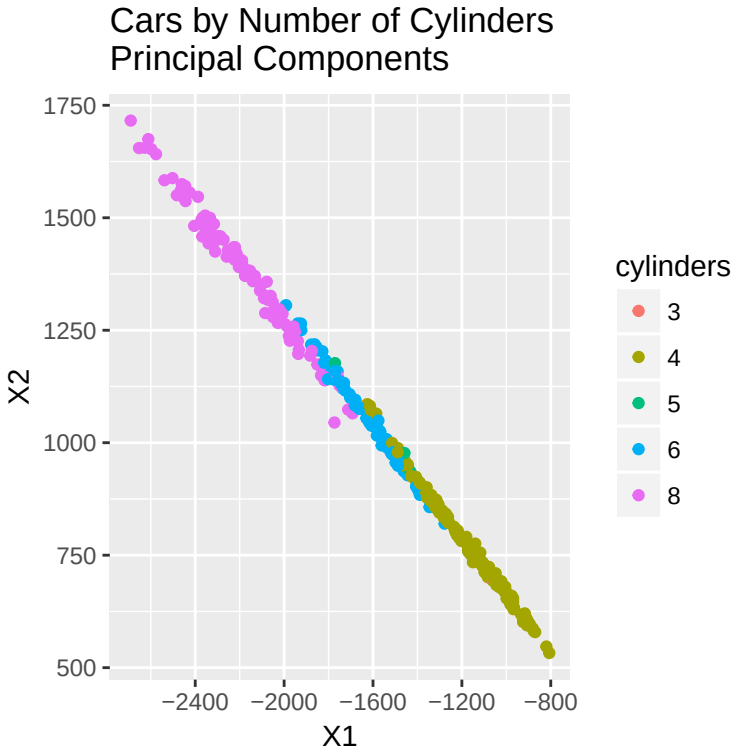
```
##
## Loadings:
##              Comp.1 Comp.2 Comp.3 Comp.4 Comp.5
## mpg           0.444 -0.304  0.839
## displacement -0.483  0.135  0.371 -0.476  0.620
## horsepower   -0.484 -0.124  0.206  0.826  0.160
## weight       -0.471  0.326  0.305 -0.159 -0.744
```

```
## acceleration  0.335  0.876  0.150  0.257  0.178
##
##                Comp.1 Comp.2 Comp.3 Comp.4 Comp.5
## SS loadings       1.0    1.0    1.0    1.0    1.0
## Proportion Var    0.2    0.2    0.2    0.2    0.2
## Cumulative Var    0.2    0.4    0.6    0.8    1.0
```

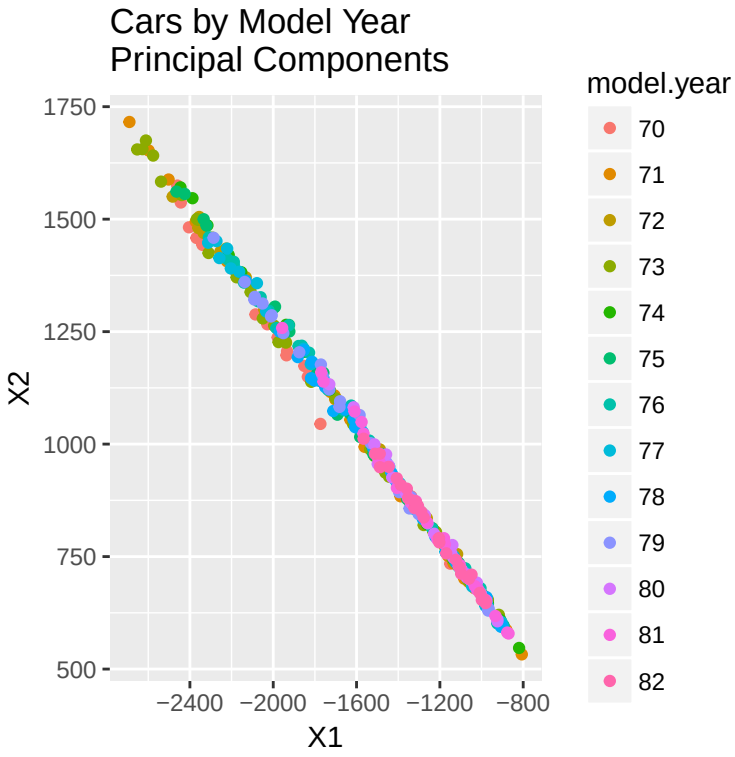
Part e

I projected the data onto the first two principal components and plotted it with a color distinction based on the number of cyclinders, model year and origin.

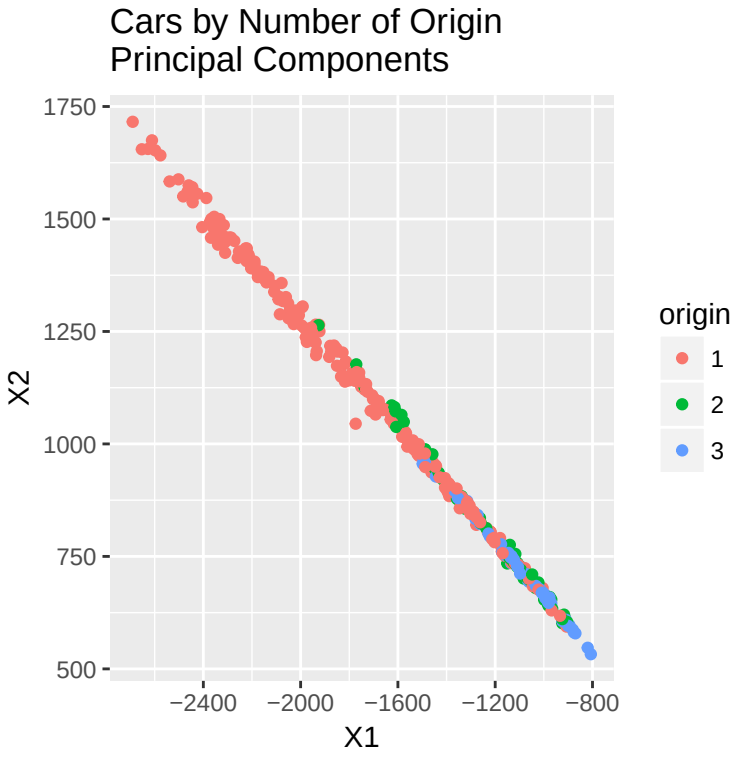
The number of cylinders in the engine appears to impact car performance heavily. As can be seen in the graph below, cars with a high number of cylinders tend to have low performance in all categories whereas cars with fewer cylinders tend to have moderate to high performance in all categories but have at least moderate to high performance in acceleration. Interestingly there is one car with high performance in all the categories but moderate performance in acceleration. (Manually filtering the data reveals this to be the Oldsmobile Cutlass Ciera.)



As would be expected, there's an increase in general performance as model year increases, which can be seen in the graph below.



Finally, it is interesting to note the differences in performance based on origin. American cars tend to be all over the spectrum whereas European and Japanese cars consistently perform better in acceleration.

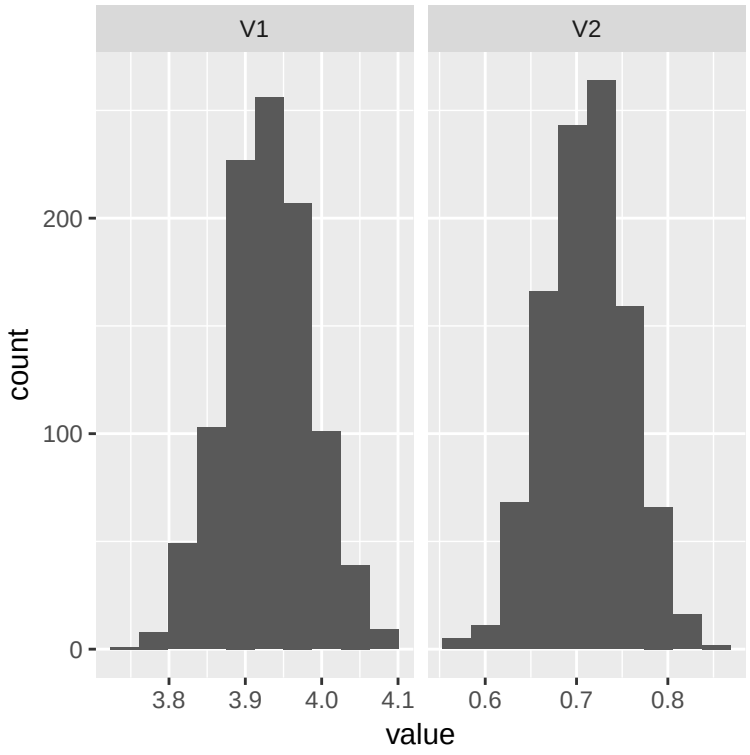


Part f

Below I've used bootstrap to estimate the 95% confidence interval for the percent of variance explained by the first two principal components. I've also graphed the histograms of the bootstrapped eigenvalues. The confidence interval for the PCs are fairly tight indicating that the two principals components that were choosen are good estimates of the data.

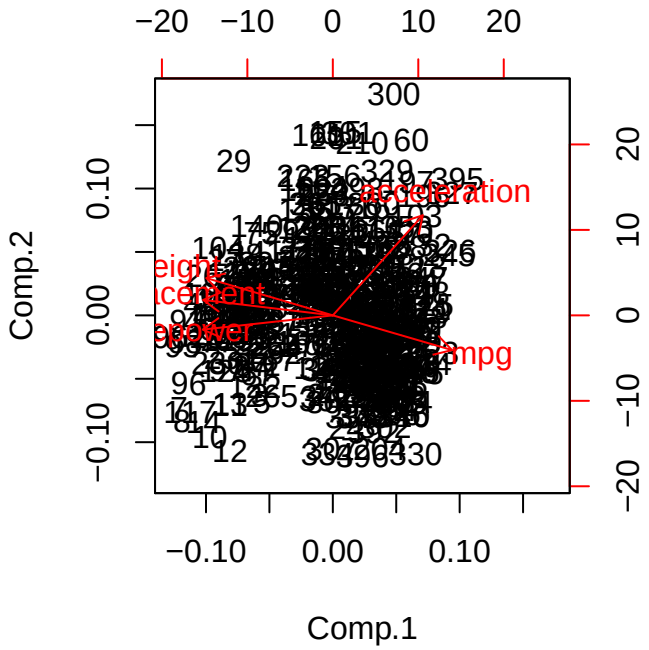
```
##         Comp.1    Comp.2
## 2.5%  3.821916 0.6244264
## 97.5% 4.037066 0.7975007
```

```
## No id variables; using all as measure variables
```



Part g

Below is the PCA biplot of the cars data. Not surprisingly, acceleration and MPG are nearly orthoganal and weight is in the opposite direction as MPG.




```
knitr::opts_chunk$set(echo = FALSE, warning = FALSE, fig.align = "center", fig.height = 4, fig.width = 4

##########################################
# Set working directory and load libraries
##########################################

setwd("~/Documents/UMich/Classes/2018 Spring/STATS 503/Homework/HW1/")
library(mixtools)
library(ggplot2)
library(plotly)
library(matrixcalc)
library(scatterplot3d)
library(reshape2)
library(knitr)


##########################################
# Get the data that we're interested in
##########################################

# Read in txt file
height_weight <- read.table("heightWeightData.txt", col.names = c("gender", "height", "weight"))

##########################################
# Question 3, Part a
# Fit a 2-dim Gaussian to the male data
# Plot male data with the ellipse
##########################################

# Extract only height and weight of males
male_hw <- subset(height_weight[, 2:3], height_weight$gender == 1)

# Figure out the mean vector and the covariance matrix
```

```r
mu <- apply(male_hw, 2, mean)
sigma <- cov(male_hw)

# Learn how to use ggplot and plot the height and weight with male ellipse
male_hw_ell_plot = ggplot(male_hw, aes(x = height, y = weight))
male_hw_ell_plot + ggtitle("Height versus Weight") +
  geom_text(aes(label=rownames(male_hw))) +
    stat_ellipse(data = male_hw, type = "norm", level = 0.90) +
    stat_ellipse(data = male_hw, type = "norm", level = 0.95) +
    stat_ellipse(data = male_hw, type = "norm", level = 0.99)

male_hw[c("63", "101"),]


##########################################
# Question 3, Part b
# Standardize and repeat part a
##########################################

# standardize the data
x_jbar <- apply(male_hw, 2, mean)
sd_j <- apply(male_hw, 2, sd)
male_hw_stand = male_hw
male_hw_stand$height <- (male_hw_stand$height - x_jbar[1])/sd_j[1]
male_hw_stand$weight <- (male_hw_stand$weight - x_jbar[2])/sd_j[2]

# Figure out the mean vector and the covariance matrix
mu_stand <- apply(male_hw_stand, 2, mean)
sigma_stand <- cov(male_hw_stand)

# Learn how to use ggplot and plot the height and weight with male ellipse
male_hw_stand_ell_plot = ggplot(male_hw_stand, aes(x = height, y = weight))
male_hw_stand_ell_plot + ggtitle("Height versus Weight Standardized") +
  geom_text(aes(label=rownames(male_hw_stand))) +
  stat_ellipse(data = male_hw_stand, type = "norm", level = 0.90) +
  stat_ellipse(data = male_hw_stand, type = "norm", level = 0.95) +
  stat_ellipse(data = male_hw_stand, type = "norm", level = 0.99)


##########################################
# Question 3, part c
# Whitening/sphereing the data
##########################################

# Compute the eigenvector and eigenvalues of the covariance matrix
eigenvalues = eigen(cov(male_hw))$values
eigenvectors = eigen(cov(male_hw))$vectors

# Whitening the data
white_trans <- solve(diag(eigenvalues,2))^(1/2) %*% t(eigenvectors) %*% t(male_hw)
white <- as.data.frame(t(white_trans))
colnames(white) = c("height", "weight")
summary(white)
```

```r
# Plot whitened male data with ellipse
male_white = ggplot(white, aes(x= height, y=weight))
male_white + ggtitle("Height versus Weight Whitened") +
  geom_text(aes(label=rownames(white))) +
  stat_ellipse(data = white, type = "norm", level = .90) +
  stat_ellipse(data = white, type = "norm", level = .95) +
  stat_ellipse(data = white, type = "norm", level = .99)


##########################################
# Question 4
##########################################

# Load in the data
fa_data <- read.table("fa_data.txt")

# Part a) Create some visualizations of the data
with(fa_data, scatterplot3d(V1, V2, V3, pch = 19, angle=280))
with(fa_data, scatterplot3d(V4, V5, V6, pch = 19, angle=20))


# Part b)
# Standardize the data
fa_stand = scale(fa_data)

# Find the PCA components
# find the eigenvalues and eigenvectors of the covariance matrix
eigenvalues = as.matrix(diag(eigen(cov(fa_stand))$values, nrow = 7))
eigenvalues # First two eigenvalues are really big; scaler of the direction
eigenvectors = eigen(cov(fa_stand))$vectors # tells you the direction
eigenvectors
# Proportion of the first two PCs
(eigenvalues[1,1] + eigenvalues[2,2]) / matrix.trace(eigenvalues)

# Reducing fa-data to 2 dimensions

# Keep the eigenvectors that we want; i.e. the first two PCs
eigenvec_keep <- eigenvectors[, 1:2]

# Get the new dataset by multiplying the transpose of the PCs and the transpose of the data
fa_data_rd <- fa_data_rd_test <- data.frame(as.matrix(fa_data) %*% eigenvec_keep)
plot(fa_data_rd$X1, fa_data_rd$X2, xlab = "PC1", ylab = "PC2",
     main = "Principal Components of \nfa_data in 2D")

##########################################
# Question 5
##########################################

# Read in data and clean
cars <- read.table("auto-mpg.data", col.names = c("mpg", "cylinders", "displacement", "horsepower", "we

# contains "?"
cars$horsepower <- as.numeric(as.character(cars$horsepower))
```

```r
# Converting to factors
cars$cylinders <- as.factor(cars$cylinders)
cars$model.year <- as.factor(cars$model.year)
cars$origin <- as.factor(cars$origin)


# Create exploratory graphs and summaries
summary(cars)


hist(cars$mpg, xlab = "MPG", main = "Histogram of MPG")

boxplot(mpg~cylinders,data=cars, main="Car Mileage by \nNumber of Cylinders",
        xlab="Number of Cylinders", ylab="MPG")

boxplot(mpg~model.year,data=cars, main="Car Mileage by Model Year",
        xlab="Model Year", ylab="MPG")


horse_mpg = ggplot(cars, aes(x= horsepower, y=mpg))
horse_mpg + ggtitle("Horsepower versus MPG") +
  geom_text(aes(label=rownames(cars)))
cars[334,]

horse_weight = ggplot(cars, aes(x= horsepower, y=weight))
horse_weight + ggtitle("Horsepower versus Weight") +
  geom_text(aes(label=rownames(cars)))
cars[14,]

horse_acc = ggplot(cars, aes(x= horsepower, y=acceleration))
horse_acc + ggtitle("Horsepower versus \nAcceleration") +
  geom_text(aes(label=rownames(cars)))
cars[29,]


# Center the data for PCA
# The variables of interest are: MPG, displacement, horsepower, weight, and acceleration
pca_el_cars <- cars[,c(1, 3:6)]

pca_el_cars_cent <- as.matrix(scale(na.omit(pca_el_cars), scale = F)) # matrix so we can do PCA #standa

# Perform PCA using covariates
pca_cov_results <- princomp(pca_el_cars_cent, cor = F)
summary(pca_cov_results)

# Perform PCA using correlation
# Pick correlation because the variables are not on similar scales
pca_cor_results <- princomp(pca_el_cars_cent, cor = T)
summary(pca_cor_results)


# Scree plot; plots the sqrt of the eigenvalues or just the eigenvalues in decreasing order--look for g
barplot(pca_cor_results$sdev^2, ylim = c(0, 4), ylab = "Eigenvalues",
```

```
        main = "Scree Plot from \nPCA using Correlation")


loadings(pca_cor_results)


# Variable loadings; correlation coefficients of the variables and the factors
cars_eigenvec <- loadings(pca_cor_results)[, 1:2]

# Plot of the variables in 2D
cars_pc <- as.matrix(pca_el_cars) %*% cars_eigenvec
cars_pc <- data.frame(cars_pc)
colnames(cars_pc) <- c("X1", "X2")
cars_pc <- cbind(cars_pc, cars[, c("cylinders", "model.year", "origin", "car.name")])

# Include colors by different categorical variables to determine maximum separation
ggplot(cars_pc) + ggtitle("Cars by Number of Cylinders \nPrincipal Components") +
  geom_point(aes(x=X1, y=X2, col = cylinders))


ggplot(cars_pc) + ggtitle("Cars by Model Year \nPrincipal Components") +
  geom_point(aes(x=X1, y=X2, col = model.year))


ggplot(cars_pc) + ggtitle("Cars by Number of Origin \nPrincipal Components") +
  geom_point(aes(x=X1, y=X2, col = origin))


# bootstrap the confidence interval for the percent of variance explained by the first two PC's

#create the bootstrap sample
bootstrap_cars_i <- lapply(1:1000, function(i) {sample(1:nrow(pca_el_cars_cent), replace = T)}) #create

# calculate eigenvalues of the bootstrap samples
eigen_bootstrap_cars <- sapply(bootstrap_cars_i, function(j) {princomp(pca_el_cars_cent[j,], cor = T)$s

bootstrap_summary_cars <- apply(eigen_bootstrap_cars, 1, function(result) {c(quantile(result, probs = c
bootstrap_summary_cars[, 1:2]

bootstrap_plot_results_cars <- as.data.frame(t(eigen_bootstrap_cars))
colnames(bootstrap_plot_results_cars) <- sapply(1:5, function(i) {paste("V", i, sep = "")})
bootstrap_plot_results_cars <- melt(bootstrap_plot_results_cars[, 1:2])
ggplot(bootstrap_plot_results_cars) + geom_histogram(aes(value), bins = 10) +
  facet_wrap(~variable, scales = "free_x")


# PCA biplot
biplot(pca_cor_results)


##########################################
# Extra Code
##########################################
```

```r
# Plot the points using basic R
plot(male_hw$height, male_hw$weight, type = "n", xlab = "Height", ylab = "Weight",
     main = "test")
ellipse(mu, sigma, alpha=0.1, npoints = 200, newplot = FALSE)
text(male_hw$height, male_hw$weight, labels = row.names(male_hw))
```