Modern Data Mining, HW 2

Kevin Sun William Walsh Hanson Wang

Due: 11:59 PM, Sunday, 02/13

Contents

Overview

Principle Component Analysis is widely used in data exploration, dimension reduction, data visualization. The aim is to transform original data into uncorrelated linear combinations of the original data while keeping the information contained in the data. High dimensional data tends to show clusters in lower dimensional view.

Clustering Analysis is another form of EDA. Here we are hoping to group data points which are close to each other within the groups and far away between different groups. Clustering using PC's can be effective. Clustering analysis can be very subjective in the way we need to summarize the properties within each group.

Both PCA and Clustering Analysis are so called unsupervised learning. There is no response variables involved in the process.

For supervised learning, we try to find out how does a set of predictors relate to some response variable of the interest. Multiple regression is still by far, one of the most popular methods. We use linear models as a working model for its simplicity and interpretability. It is important that we use domain knowledge as much as we can to determine the form of the response as well as the function format of the factors.

0.1 Objectives

- PCA
- SVD
- Clustering Analysis
- Linear Regression

0.2 Review materials

- Study Module 2: PCA
- Study Module 3: Clustering Analysis
- Study Module 4: Multiple regression

0.3 Data needed

- NLSY79.csv
- brca_subtype.csv
- brca_x_patient.csv

1 Case study 1: Self-seteem

Self-esteem generally describes a person's overall sense of self-worthiness and personal value. It can play significant role in one's motivation and success throughout the life. Factors that influence self-esteem can be inner thinking, health condition, age, life experiences etc. We will try to identify possible factors in our data that are related to the level of self-esteem.

In the well-cited National Longitudinal Study of Youth (NLSY79), it follows about 13,000 individuals and numerous individual-year information has been gathered through surveys. The survey data is open to public here. Among many variables we assembled a subset of variables including personal demographic variables in different years, household environment in 79, ASVAB test Scores in 81 and Self-Esteem scores in 81 and 87 respectively.

The data is store in NLSY79.csv.

Here are the description of variables:

Personal Demographic Variables

- Gender: a factor with levels “female” and “male”
- Education05: years of education completed by 2005
- HeightFeet05, HeightInch05: height measurement. For example, a person of 5’10 will be recorded as HeightFeet05=5, HeightInch05=10.
- Weight05: weight in lbs.
- Income87, Income05: total annual income from wages and salary in 2005.
- Job87, Job05: job type in 1987 and 2005, including Protective Service Occupations, Food Preparation and Serving Related Occupations, Cleaning and Building Service Occupations, Entertainment Attendants and Related Workers, Funeral Related Occupations, Personal Care and Service Workers, Sales and Related Workers, Office and Administrative Support Workers, Farming, Fishing and Forestry Occupations, Construction Trade and Extraction Workers, Installation, Maintenance and Repairs Workers, Production and Operating Workers, Food Preparation Occupations, Setters, Operators and Tenders, Transportation and Material Moving Workers

Household Environment

- Imagazine: a variable taking on the value 1 if anyone in the respondent’s household regularly read magazines in 1979, otherwise 0
- Inewspaper: a variable taking on the value 1 if anyone in the respondent’s household regularly read newspapers in 1979, otherwise 0
- Ilibrary: a variable taking on the value 1 if anyone in the respondent’s household had a library card in 1979, otherwise 0
- MotherEd: mother’s years of education
- FatherEd: father’s years of education
- FamilyIncome78

Variables Related to ASVAB test Scores in 1981

Test	Description
AFQT	percentile score on the AFQT intelligence test in 1981
Coding	score on the Coding Speed test in 1981
Auto	score on the Automotive and Shop test in 1981
Mechanic	score on the Mechanic test in 1981
Elec	score on the Electronics Information test in 1981
Science	score on the General Science test in 1981
Math	score on the Math test in 1981
Arith	score on the Arithmetic Reasoning test in 1981
Word	score on the Word Knowledge Test in 1981
Parag	score on the Paragraph Comprehension test in 1981
Numer	score on the Numerical Operations test in 1981

Self-Esteem test 81 and 87

We have two sets of self-esteem test, one in 1981 and the other in 1987. Each set has same 10 questions. They are labeled as Esteem81 and Esteem87 respectively followed by the question number. For example, Esteem81_1 is Esteem question 1 in 81.

The following 10 questions are answered as 1: strongly agree, 2: agree, 3: disagree, 4: strongly disagree

- Esteem 1: “I am a person of worth”

- Esteem 2: “I have a number of good qualities”
- Esteem 3: “I am inclined to feel like a failure”
- Esteem 4: “I do things as well as others”
- Esteem 5: “I do not have much to be proud of”
- Esteem 6: “I take a positive attitude towards myself and others”
- Esteem 7: “I am satisfied with myself”
- Esteem 8: “I wish I could have more respect for myself”
- Esteem 9: “I feel useless at times”
- Esteem 10: “I think I am no good at all”

1.1 Data preparation

Load the data. Do a quick EDA to get familiar with the data set. Pay attention to the unit of each variable. Are there any missing values?

- They data has 46 columns and 2431 individual observations in total. As shown in the summary output, there are no missing values. All of the variables are numeric.

```
data.esteem=read.csv("data/NLSY79.csv")
dim(data.esteem)
```

```
## [1] 2431   46
```

```
names(data.esteem)
```

```
##  [1] "Subject"       "Gender"        "Education05"   "Income87"
##  [5] "Job05"         "Income05"      "Weight05"      "HeightFeet05"
##  [9] "HeightInch05"  "Imagazine"     "Inewspaper"    "Ilibrary"
## [13] "MotherEd"      "FatherEd"      "FamilyIncome78" "Science"
## [17] "Arith"         "Word"          "Parag"         "Number"
## [21] "Coding"        "Auto"          "Math"          "Mechanic"
## [25] "Elec"          "AFQT"          "Esteem81_1"    "Esteem81_2"
## [29] "Esteem81_3"    "Esteem81_4"    "Esteem81_5"    "Esteem81_6"
## [33] "Esteem81_7"    "Esteem81_8"    "Esteem81_9"    "Esteem81_10"
## [37] "Esteem87_1"    "Esteem87_2"    "Esteem87_3"    "Esteem87_4"
## [41] "Esteem87_5"    "Esteem87_6"    "Esteem87_7"    "Esteem87_8"
## [45] "Esteem87_9"    "Esteem87_10"
```

```
summary(data.esteem)
```

```
##     Subject          Gender       Education05      Income87
##  Min.   :    2   female:1199   Min.   : 6.0   Min.   :   -2
##  1st Qu.: 1592   male  :1232   1st Qu.:12.0   1st Qu.: 4500
##  Median : 3137                 Median :13.0   Median :12000
##  Mean   : 3504                 Mean   :13.9   Mean   :13399
##  3rd Qu.: 4668                 3rd Qu.:16.0   3rd Qu.:19000
##  Max.   :12140                 Max.   :20.0   Max.   :59387
##
##                                                              Job05
##  10 TO 430: Executive, Administrative and Managerial Occupations: 377
##  5000 TO 5930: Office and Administrative Support Workers        : 360
##  4700 TO 4960: Sales and Related Workers                         : 205
##  6200 TO 6940: Construction Trade and Extraction Workers         : 135
##  2200 TO 2340: Teachers                                          : 120
##  9000 TO 9750: Transportation and Material Moving Workers        : 117
##  (Other)                                                         :1117
##     Income05          Weight05       HeightFeet05     HeightInch05      Imagazine
##  Min.   :     63   Min.   : 81   Min.   :-4.00   Min.   : 0.00   Min.   :0.000
##  1st Qu.: 22650   1st Qu.:150   1st Qu.: 5.00   1st Qu.: 2.00   1st Qu.:0.000
##  Median : 38500   Median :180   Median : 5.00   Median : 5.00   Median :1.000
##  Mean   : 49415   Mean   :183   Mean   : 5.18   Mean   : 5.32   Mean   :0.718
##  3rd Qu.: 61350   3rd Qu.:209   3rd Qu.: 5.00   3rd Qu.: 8.00   3rd Qu.:1.000
##  Max.   :703637   Max.   :380   Max.   : 8.00   Max.   :11.00   Max.   :1.000
##
##   Inewspaper        Ilibrary        MotherEd       FatherEd     FamilyIncome78
##  Min.   :0.000   Min.   :0.00   Min.   : 0.0   Min.   : 0.0   Min.   :    0
##  1st Qu.:1.000   1st Qu.:1.00   1st Qu.:11.0   1st Qu.:10.0   1st Qu.:11167
##  Median :1.000   Median :1.00   Median :12.0   Median :12.0   Median :20000
##  Mean   :0.861   Mean   :0.77   Mean   :11.7   Mean   :11.8   Mean   :21252
##  3rd Qu.:1.000   3rd Qu.:1.00   3rd Qu.:12.0   3rd Qu.:14.0   3rd Qu.:27500
##  Max.   :1.000   Max.   :1.00   Max.   :20.0   Max.   :20.0   Max.   :75001
##
##     Science          Arith           Word           Parag          Number
##  Min.   : 0.0   Min.   : 0.0   Min.   : 0.0   Min.   : 0.0   Min.   : 0.0
##  1st Qu.:13.0   1st Qu.:13.0   1st Qu.:23.0   1st Qu.:10.0   1st Qu.:29.0
##  Median :17.0   Median :19.0   Median :28.0   Median :12.0   Median :36.0
##  Mean   :16.3   Mean   :18.6   Mean   :26.6   Mean   :11.2   Mean   :35.5
##  3rd Qu.:20.0   3rd Qu.:25.0   3rd Qu.:32.0   3rd Qu.:14.0   3rd Qu.:44.0
##  Max.   :25.0   Max.   :30.0   Max.   :35.0   Max.   :15.0   Max.   :50.0
##
##     Coding          Auto           Math         Mechanic          Elec
##  Min.   : 0.0   Min.   : 0.0   Min.   : 0.0   Min.   : 0.0   Min.   : 0.0
##  1st Qu.:38.0   1st Qu.:10.0   1st Qu.: 9.0   1st Qu.:11.0   1st Qu.: 9.0
##  Median :48.0   Median :14.0   Median :14.0   Median :14.0   Median :12.0
##  Mean   :47.1   Mean   :14.3   Mean   :14.3   Mean   :14.4   Mean   :11.6
##  3rd Qu.:57.0   3rd Qu.:18.0   3rd Qu.:20.0   3rd Qu.:18.0   3rd Qu.:15.0
##  Max.   :84.0   Max.   :25.0   Max.   :25.0   Max.   :25.0   Max.   :20.0
##
##       AFQT         Esteem81_1     Esteem81_2     Esteem81_3     Esteem81_4
##  Min.   :  0.0   Min.   :1.00   Min.   :1.00   Min.   :1.00   Min.   :1.00
##  1st Qu.: 31.9   1st Qu.:1.00   1st Qu.:1.00   1st Qu.:3.00   1st Qu.:1.00
##  Median : 57.0   Median :1.00   Median :1.00   Median :4.00   Median :2.00
##  Mean   : 54.7   Mean   :1.42   Mean   :1.42   Mean   :3.51   Mean   :1.57
##  3rd Qu.: 78.2   3rd Qu.:2.00   3rd Qu.:2.00   3rd Qu.:4.00   3rd Qu.:2.00
##  Max.   :100.0   Max.   :4.00   Max.   :4.00   Max.   :4.00   Max.   :4.00
##
##    Esteem81_5     Esteem81_6     Esteem81_7     Esteem81_8     Esteem81_9
##  Min.   :1.00   Min.   :1.00   Min.   :1.00   Min.   :1.00   Min.   :1.00
##  1st Qu.:3.00   1st Qu.:1.00   1st Qu.:1.00   1st Qu.:3.00   1st Qu.:3.00
##  Median :4.00   Median :2.00   Median :2.00   Median :3.00   Median :3.00
##  Mean   :3.46   Mean   :1.62   Mean   :1.75   Mean   :3.13   Mean   :3.16
##  3rd Qu.:4.00   3rd Qu.:2.00   3rd Qu.:2.00   3rd Qu.:4.00   3rd Qu.:4.00
##  Max.   :4.00   Max.   :4.00   Max.   :4.00   Max.   :4.00   Max.   :4.00
##
##   Esteem81_10     Esteem87_1     Esteem87_2     Esteem87_3     Esteem87_4
##   Min.   :1.0    Min.   :1.00    Min.   :1.0    Min.   :1.00    Min.   :1.0
##   1st Qu.:3.0    1st Qu.:1.00    1st Qu.:1.0    1st Qu.:3.00    1st Qu.:1.0
##   Median :3.0    Median :1.00    Median :1.0    Median :4.00    Median :1.0
##   Mean   :3.4    Mean   :1.38    Mean   :1.4    Mean   :3.58    Mean   :1.5
##   3rd Qu.:4.0    3rd Qu.:2.00    3rd Qu.:2.0    3rd Qu.:4.00    3rd Qu.:2.0
##   Max.   :4.0    Max.   :4.00    Max.   :4.0    Max.   :4.00    Max.   :4.0
##
##     Esteem87_5      Esteem87_6      Esteem87_7      Esteem87_8      Esteem87_9
##   Min.   :1.00    Min.   :1.00    Min.   :1.00    Min.   :1.0    Min.   :1.00
##   1st Qu.:3.00   1st Qu.:1.00   1st Qu.:1.00    1st Qu.:3.0    1st Qu.:3.00
##   Median :4.00   Median :2.00   Median :2.00    Median :3.0    Median :3.00
##   Mean   :3.53    Mean   :1.59    Mean   :1.72    Mean   :3.1     Mean   :3.06
##   3rd Qu.:4.00    3rd Qu.:2.00    3rd Qu.:2.00    3rd Qu.:4.0    3rd Qu.:4.00
##   Max.   :4.00    Max.   :4.00    Max.   :4.00    Max.   :4.0    Max.   :4.00
##
##   Esteem87_10
##    Min.   :1.00
##   1st Qu.:3.00
##   Median :3.00
##  Mean  :3.37 ##
3rd    Qu.:4.00  ##
Max.   :4.00##
```

1.2 Self esteem evaluation

Let concentrate on Esteem scores evaluated in 87.

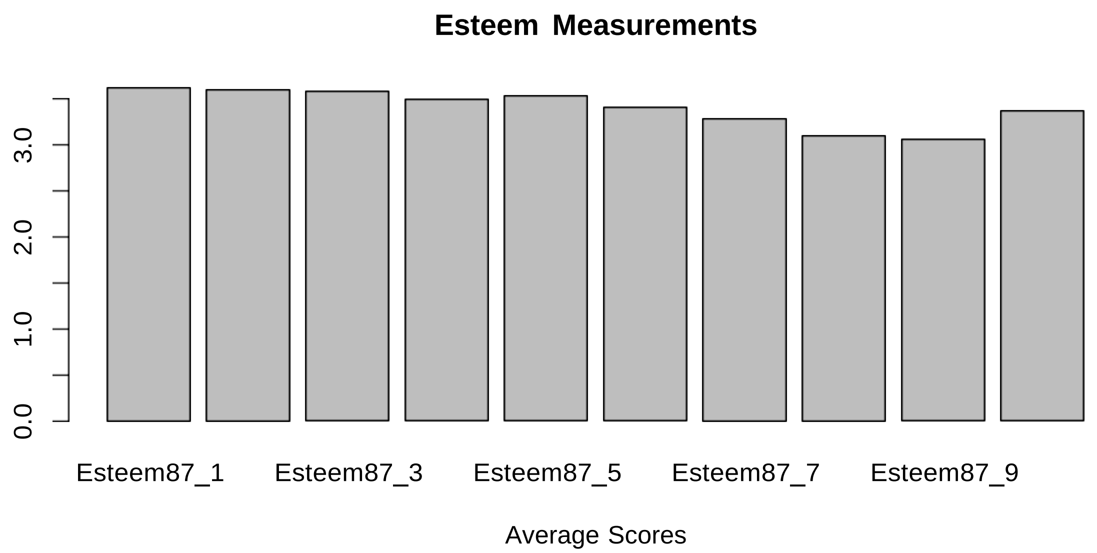
1. Reverse Esteem 1, 2, 4, 6, and 7 so that a higher score corresponds to higher self-esteem. (Hint: if we store the esteem data in data.esteem, then data.esteem[, c(1, 2, 4, 6, 7)] <- 5 - data.esteem[, c(1, 2, 4, 6, 7)] to reverse the score.)

```
data.esteem[,  c(37, 38, 40, 42, 43)]  <- 5 - data.esteem[,  c(37, 38, 40, 42, 43)]
```

2. Write a brief summary with necessary plots about the 10 esteem measurements.

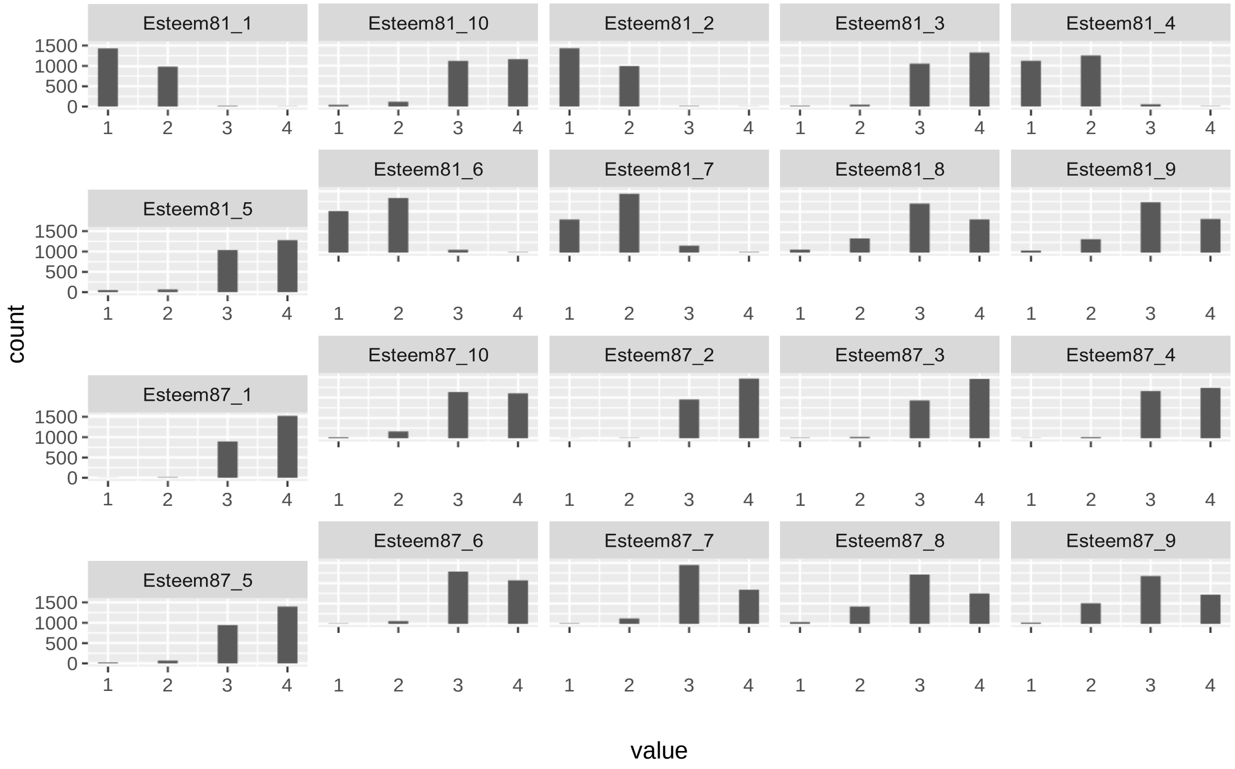
The average esteem scores seem to be roughly around 3.0 ~ 3.5.

```
barplot(colMeans(data.esteem[,c(37, 38, 39, 40, 41, 42, 43, 44, 45, 46)]), main="Esteem Measurements",x
```



The esteem scores are relatively consistent across the years, with only Esteem_4 changing from a majority 3 score to a majority 4 score.

```
library(tidyr)
library(ggplot2)
graphable_esteem <- data.esteem[,27:46]
ggplot(gather(graphable_esteem), aes(value)) +
    geom_histogram(bins = 10) +
    facet_wrap(~key, scales = 'free_x')
```



3. Do esteem scores all positively correlated? Report the pairwise correlation table and write a brief summary.

The correlation coefficient is a statistical measure of the strength of the relationship between the relative movements of two variables. There appears to be quite a strong positive correlation among all the esteem scores. For example, we see that question 1 (“I am a person of worth”) and question 2 (“I have a number of good qualities”) are highly correlated with a correlation coefficient of 0.7.

```
res2 <- cor(data.esteem[,c(37, 38, 39, 40, 41, 42, 43, 44, 45, 46)])
res2
```

```
##              Esteem87_1 Esteem87_2 Esteem87_3 Esteem87_4 Esteem87_5 Esteem87_6
## Esteem87_1        1.000      0.704      0.448      0.528      0.399      0.464
## Esteem87_2        0.704      1.000      0.443      0.551      0.402      0.481
## Esteem87_3        0.448      0.443      1.000      0.408      0.549      0.410
## Esteem87_4        0.528      0.551      0.408      1.000      0.381      0.509
## Esteem87_5        0.399      0.402      0.549      0.381      1.000      0.405
## Esteem87_6        0.464      0.481      0.410      0.509      0.405      1.000
## Esteem87_7        0.379      0.410      0.343      0.422      0.370      0.600
## Esteem87_8        0.273      0.283      0.351      0.295      0.381      0.409
## Esteem87_9        0.236      0.259      0.349      0.287      0.354      0.364
## Esteem87_10       0.312      0.330      0.460      0.366      0.436      0.442
##              Esteem87_7 Esteem87_8 Esteem87_9 Esteem87_10
## Esteem87_1        0.379      0.273      0.236       0.312
## Esteem87_2        0.410      0.283      0.259       0.330
## Esteem87_3        0.343      0.351      0.349       0.460
## Esteem87_4        0.422      0.295      0.287       0.366
## Esteem87_5        0.370      0.381      0.354       0.436
## Esteem87_6        0.600      0.409      0.364       0.442
## Esteem87_7        1.000      0.389      0.352       0.390
## Esteem87_8        0.389      1.000      0.430       0.438
## Esteem87_9        0.352      0.430      1.000       0.579
## Esteem87_10       0.390      0.438      0.579       1.000
```

4. PCA on 10 esteem measurements. (centered but no scaling)

   a) Report the PC1 and PC2 loadings. Are they unit vectors? Are they orthogonal?

As shown in the output below, they are indeed orthogonal unit vectors. The PCs are all independent from one another and the squared sum of the PC1 Coefficient equals 1.

```
#Centering
library(dplyr)
center_scale <- function(x) {
    scale(x, center= TRUE, scale = TRUE)
}
data.esteem.centered <- center_scale(data.esteem[,c(37, 38, 39, 40, 41, 42, 43, 44, 45, 46)])
data.esteem.centered <- as.data.frame(data.esteem.centered)
pc.2 <- prcomp(data.esteem.centered)
names(pc.2)
```

```
## [1] "sdev"     "rotation" "center"   "scale"    "x"
```

```
pc.2$rotation[, 1:2]
```

```
##               PC1     PC2
## Esteem87_1  0.324 -0.4452
## Esteem87_2  0.333 -0.4283
## Esteem87_3  0.322  0.0115
## Esteem87_4  0.324 -0.2877
## Esteem87_5  0.315  0.0793
## Esteem87_6  0.347 -0.0492
## Esteem87_7  0.315  0.0196
## Esteem87_8  0.280  0.3619
## Esteem87_9  0.277  0.4917
## Esteem87_10 0.318  0.3918
```

```
sum=0
for (val in pc.2$rotation[, "PC1"]){
  sum = sum + val^2
}
sum
```

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

PC1 is all positive. It makes sense since they are the questions that we flipped. Their values flipped for questions 1, 2, 4, 6

b) Are there good interpretations for PC1 and PC2? (If loadings are all negative, take the positive loadings for the ease of interpretation)

PC1s are entirely positive, so it represents the total esteem of someone surveyed, or their general view towards themselves. PC2 has negative values for question 1, 2, 4, and 6, which are the ones that we flipped earlier.

c) How is the PC1 score obtained for each subject? Write down the formula.

PC1 = 0.324Esteem87_1 + 0.333Esteem87_2 + 0.322Esteem87_3 + 0.324Esteem87_4 + 0.315Esteem87_5 + 0.347Esteem87_6 + 0.315Esteem87_7 + 0.280Esteem87_8 + 0.277Esteem87_9 + 0.318Esteem87_10

d) Are PC1 scores and PC2 scores in the data uncorrelated?

```
cor(pc.2$rotation[,1], scale(pc.2$rotation, scale = TRUE))
```

```
##      PC1    PC2     PC3   PC4    PC5    PC6   PC7    PC8    PC9   PC10
## [1,]   1 -0.707 -0.0456 0.158 -0.445 0.0163 -0.25 -0.413 -0.152 -0.121
```

```
cor(pc.2$rotation[,2], scale(pc.2$rotation, scale = TRUE))
```

```
##          PC1 PC2       PC3      PC4      PC5     PC6      PC7      PC8       PC9
## [1,] -0.707   1 -0.000137 0.000474 -0.00133 4.9e-05 -0.00075 -0.00124 -0.000456 ##
             PC10
## [1,] -0.000364
```

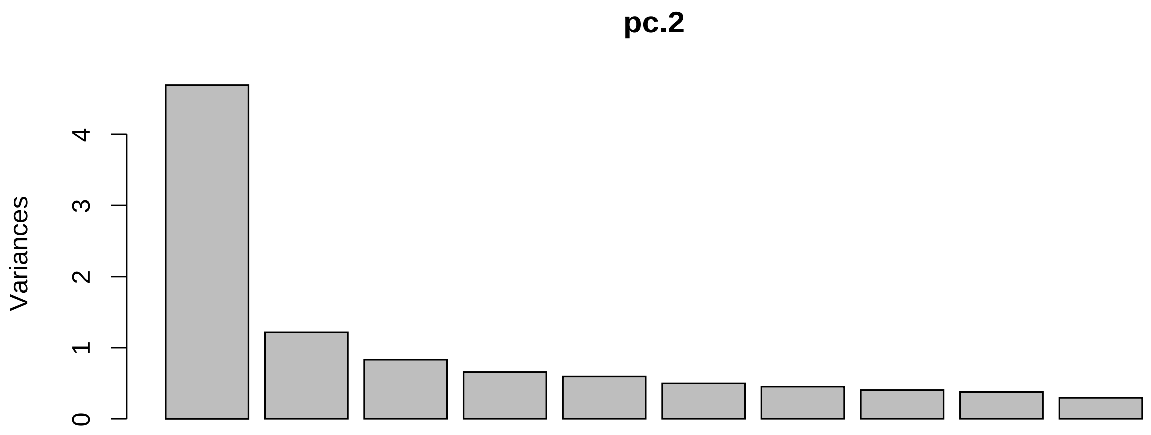
Yes, all PC scores are uncorrelated and independent.

e) Plot PVE (Proportion of Variance Explained) and summarize the plot.

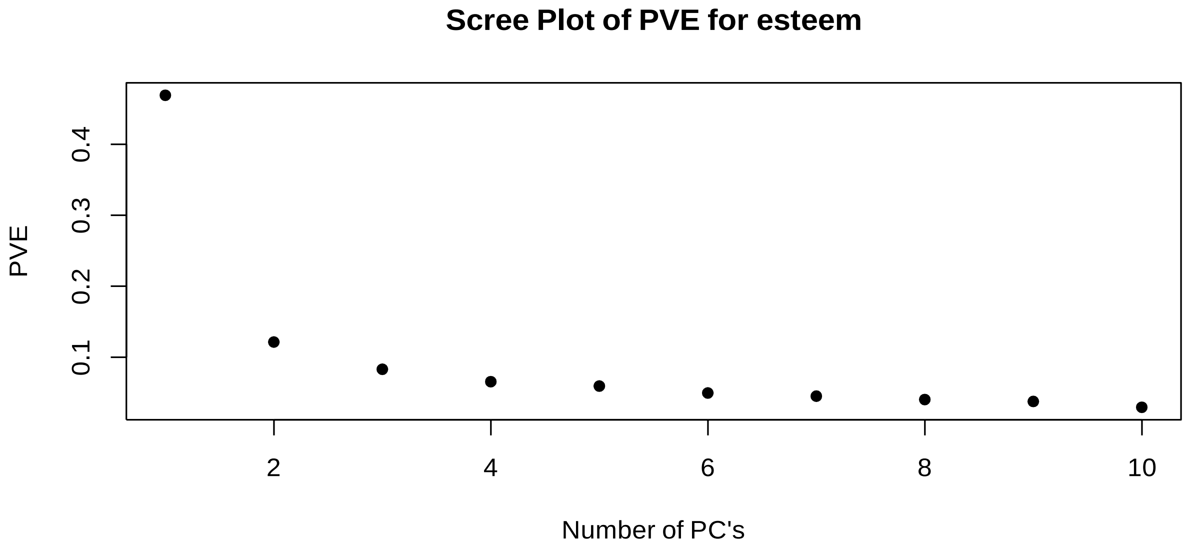
```
summary(pc.2)$importance
```

```
##                          PC1   PC2   PC3    PC4    PC5    PC6    PC7    PC8
## Standard deviation     2.166 1.102 0.911 0.8090 0.7699 0.7037 0.6714 0.6346
## Proportion of Variance 0.469 0.121 0.083 0.0654 0.0593 0.0495 0.0451 0.0403
## Cumulative Proportion  0.469 0.590 0.673 0.7388 0.7981 0.8477 0.8927 0.9330
##                           PC9   PC10
## Standard deviation     0.6133 0.5421
## Proportion of Variance 0.0376 0.0294
## Cumulative Proportion  0.9706 1.0000
```

```
#Scree plot of variances
plot(pc.2)
```



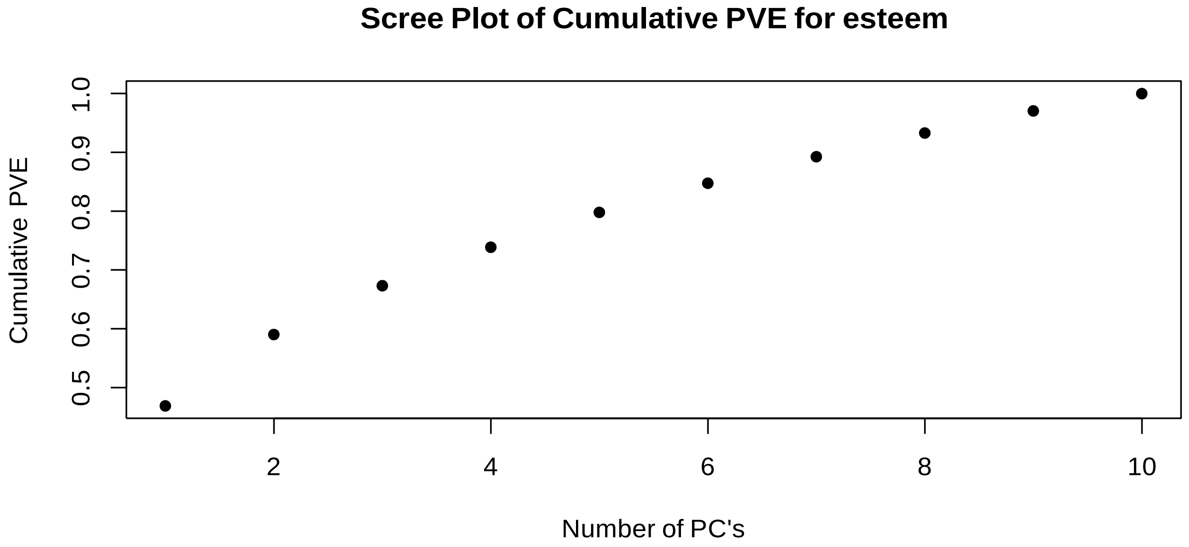
```
#Scree plot of PVE's
plot(summary(pc.2)$importance[2,],
     ylab="PVE",
     xlab="Number of PC's",
     pch=16,
     main="Scree Plot of PVE for esteem")
```



The plot shows that PC1 holds almost the majority of variance, whereas PC3 and after account for under 10% of the variability.

f) Also plot CPVE (Cumulative Proportion of Variance Explained). What proportion of the variance in the

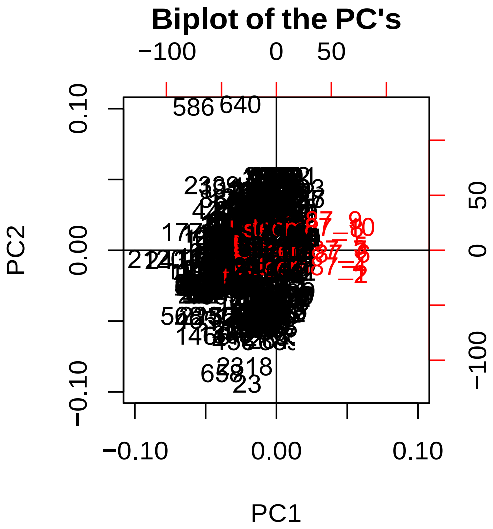
```
#Scree plot of PVE's
plot(summary(pc.2)$importance[3,],
     ylab="Cumulative PVE",
     xlab="Number of PC's",
     pch=16,
     main="Scree Plot of Cumulative PVE for esteem")
```



Around 60% of the variance in data is explained by the first two principal components.

g) PC's provide us with a low dimensional view of the self-esteem scores. Use a biplot with the first t

```
#Scree plot of PVE's
lim <- c(-.1, 0.1)
biplot(pc.2, xlim=lim, ylim=lim, main = "Biplot of the PC's")
abline(v=0, h=0)
```



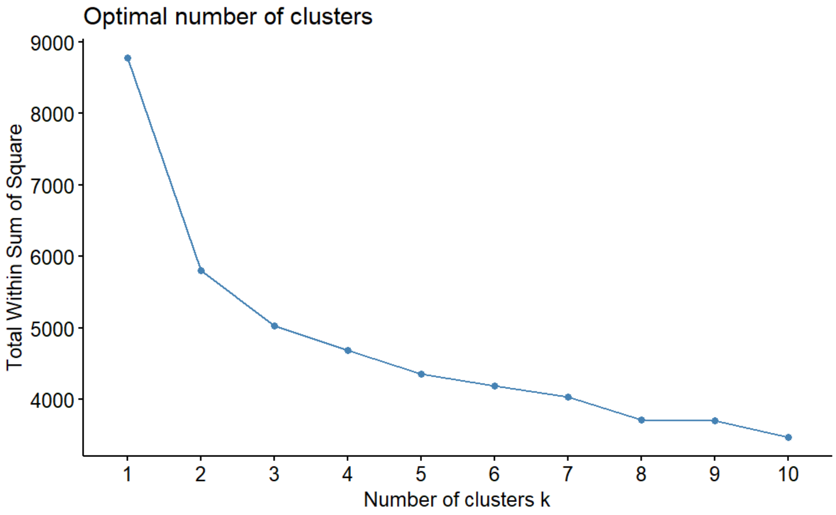
PC1 is the sum of all esteem while PC2 is the difference between all esteem questions 8, 9, 10 (positive) and questions 1, 2, 4, and 6 (negative).

5. Apply k-means to cluster subjects on the original esteem scores

   a) Find a reasonable number of clusters using within sum of squared with elbow rules.

According to elbow rule, 3 clusters may be the optimal number of clusters.

```
library("factoextra")
library(NbClust)
fviz_nbclust(data.esteem[, c(37, 38, 39, 40, 41, 42, 43, 44, 45, 46)],kmeans,
```



b) Can you summarize common features within each cluster?

```
esteem.kmeans <- kmeans(data.esteem[, c(37, 38, 39, 40, 41, 42, 43, 44, 45, 46)],centers = 3)
str(esteem.kmeans)
```

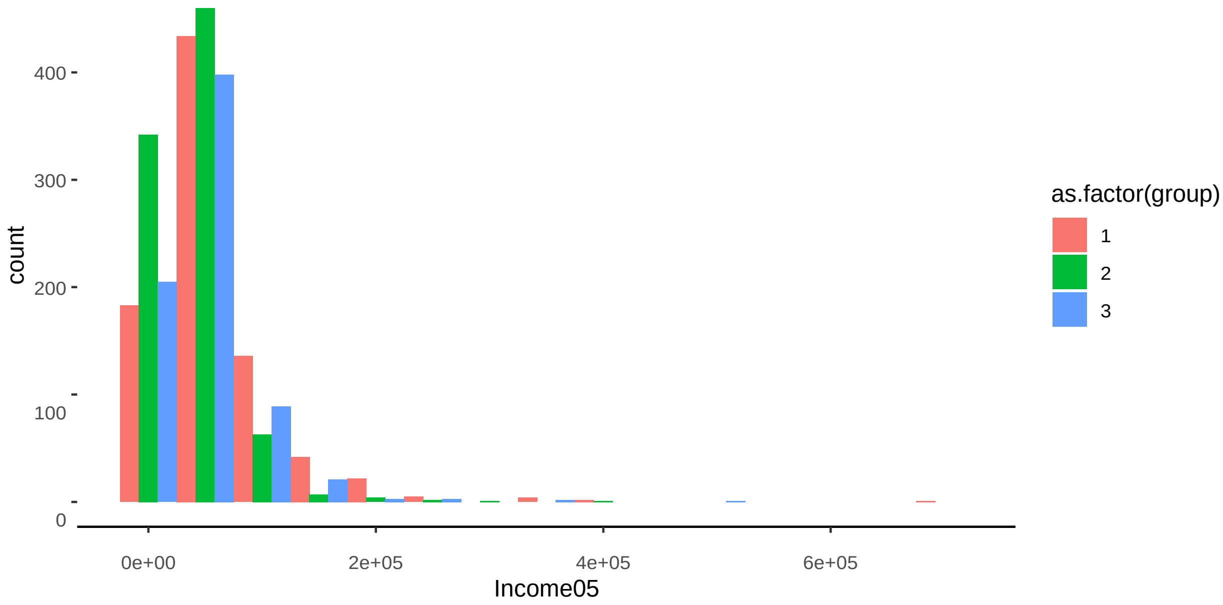
```
## List of 9
##  $ cluster     : int [1:2431] 3 3 1 3 1 1 1 2 1 1 ...
##  $ centers     : num [1:3, 1:10] 3.88 3.15 3.9 3.88 3.12 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:3] "1" "2" "3"
##   .. ..$ : chr [1:10] "Esteem87_1" "Esteem87_2" "Esteem87_3" "Esteem87_4" ...
##  $ totss       : num 8775
##  $ withinss    : num [1:3] 1225 1846 1953
##  $ tot.withinss: num 5024
##  $ betweenss    : num 3751
##  $ size         : int [1:3] 829 880 722
##  $ iter         : int 3
##  $ ifault       : int 0
##  - attr(*, "class")= chr "kmeans"
```

```
# esteem.kmeans$cluster # one label for each team, k=3 many centers
```

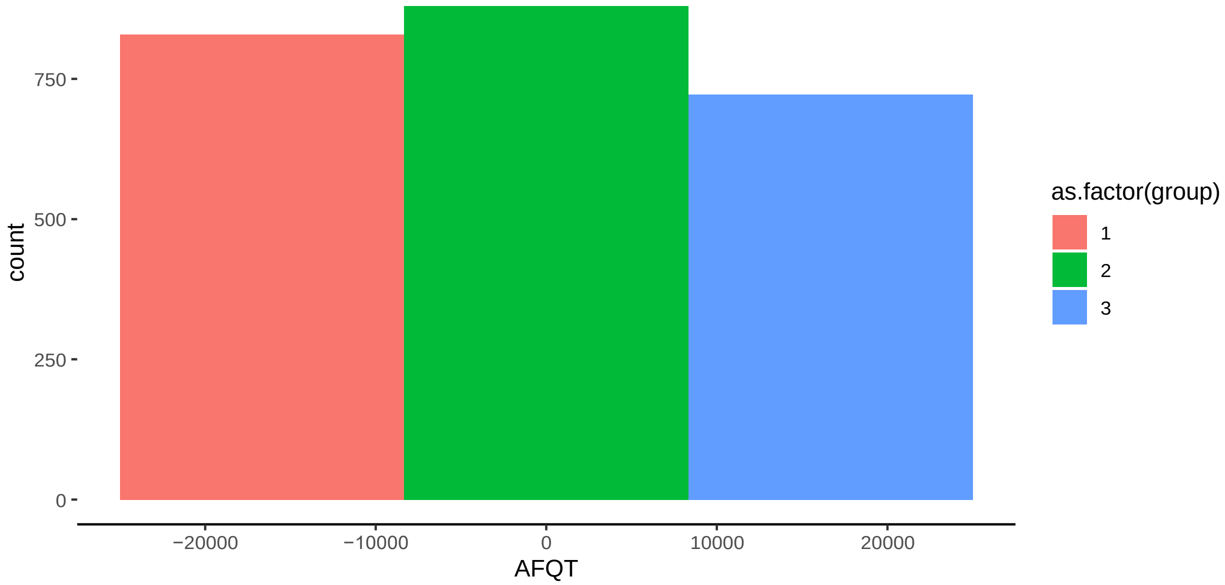
```
esteem.kmeans$size # size of each cluster
```

```
## [1] 829 880 722
```

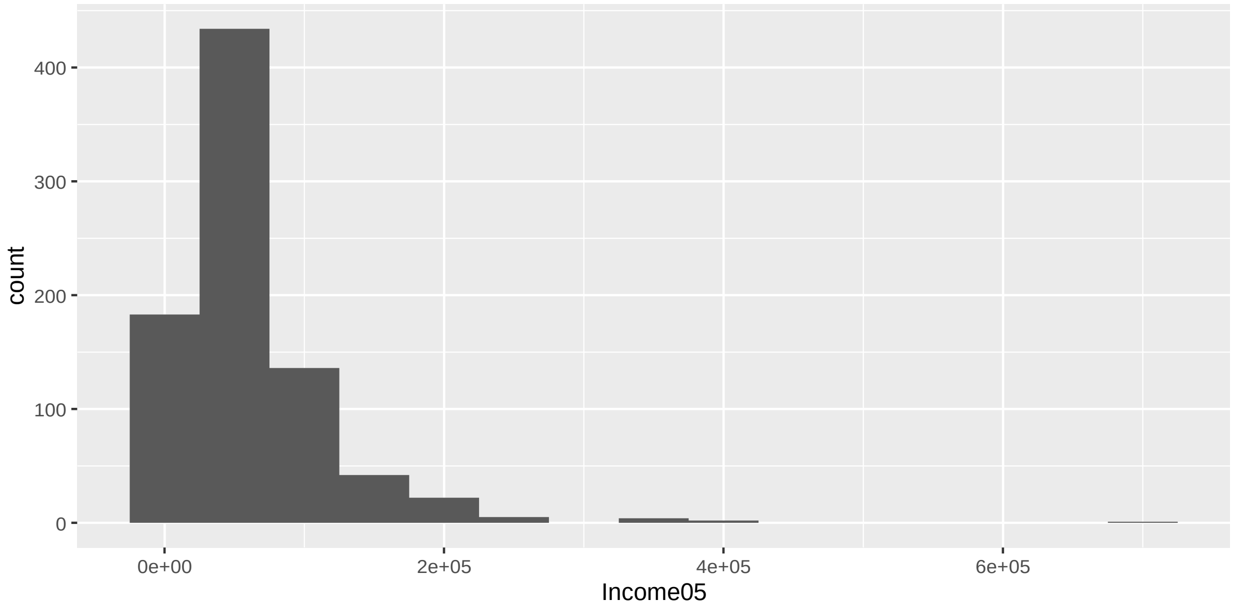
```
Income.clustered <- data.esteem %>% select (Income05) %>%
          mutate(group = esteem.kmeans$cluster) %>%
          arrange(desc(group))
ggplot(Income.clustered, aes(x = Income05, fill = as.factor(group)))+
  geom_histogram(binwidth = 50000, position = "dodge") +
  theme_classic()
```



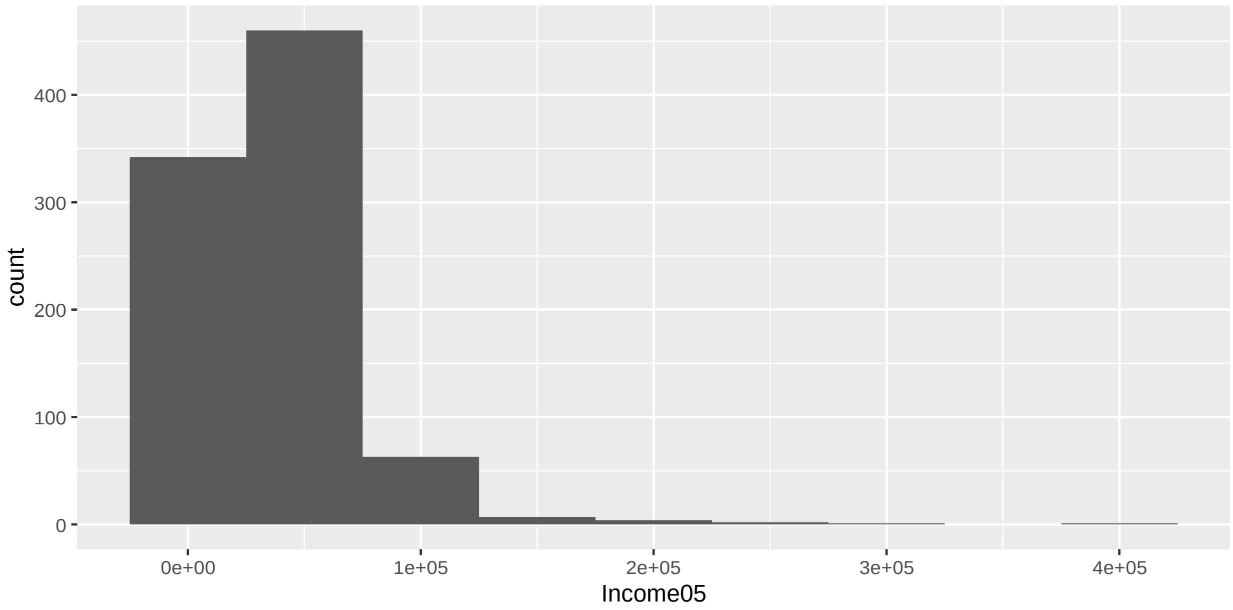
```
AFQT.clustered <- data.esteem %>% select (AFQT) %>%
          mutate(group = esteem.kmeans$cluster) %>%
          arrange(desc(group))
ggplot(AFQT.clustered, aes(x = AFQT, fill  = as.factor(group)))+
  geom_histogram(binwidth = 50000, position = "dodge") +
  theme_classic()
```



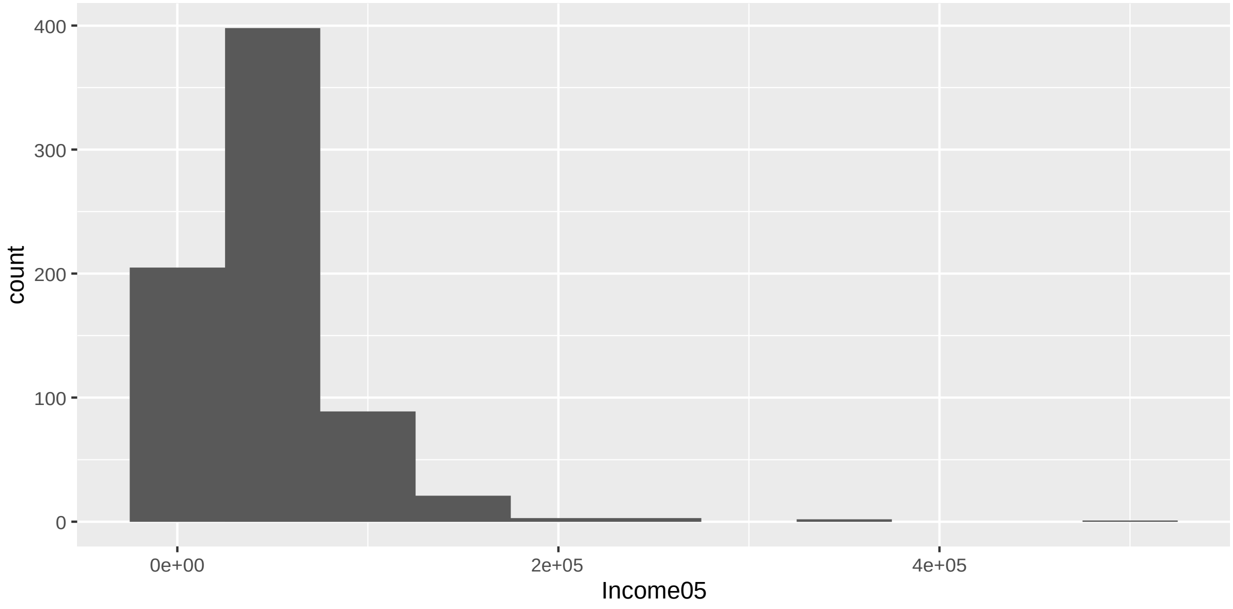
```
c1 <- data.esteem %>%
   mutate(group = esteem.kmeans$cluster)%>%
   filter(group == 1)
ggplot(c1, aes(x = Income05)) +
   geom_histogram(binwidth = 50000)
```



```
c2 <- data.esteem %>%
   mutate(group = esteem.kmeans$cluster)%>%
   filter(group == 2)
ggplot(c2, aes(x = Income05)) +
   geom_histogram(binwidth = 50000)
```



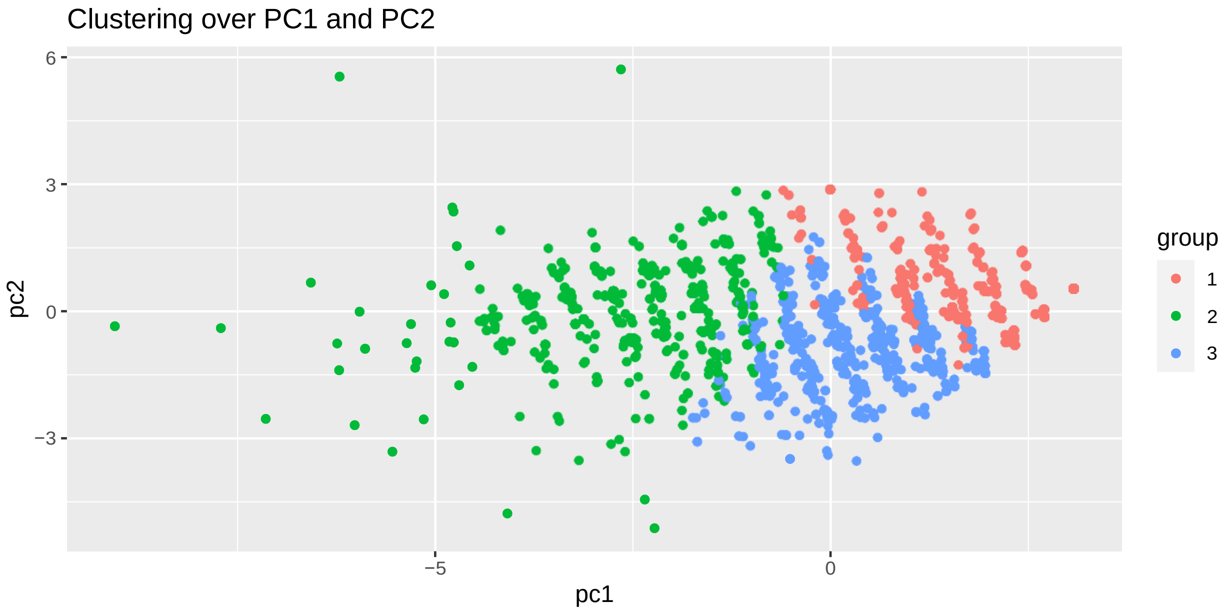
```
c3 <- data.esteem  %>%
  mutate(group = esteem.kmeans$cluster)%>%
  filter(group == 3)
ggplot(c3, aes(x = Income05)) +
  geom_histogram(binwidth = 50000)
```



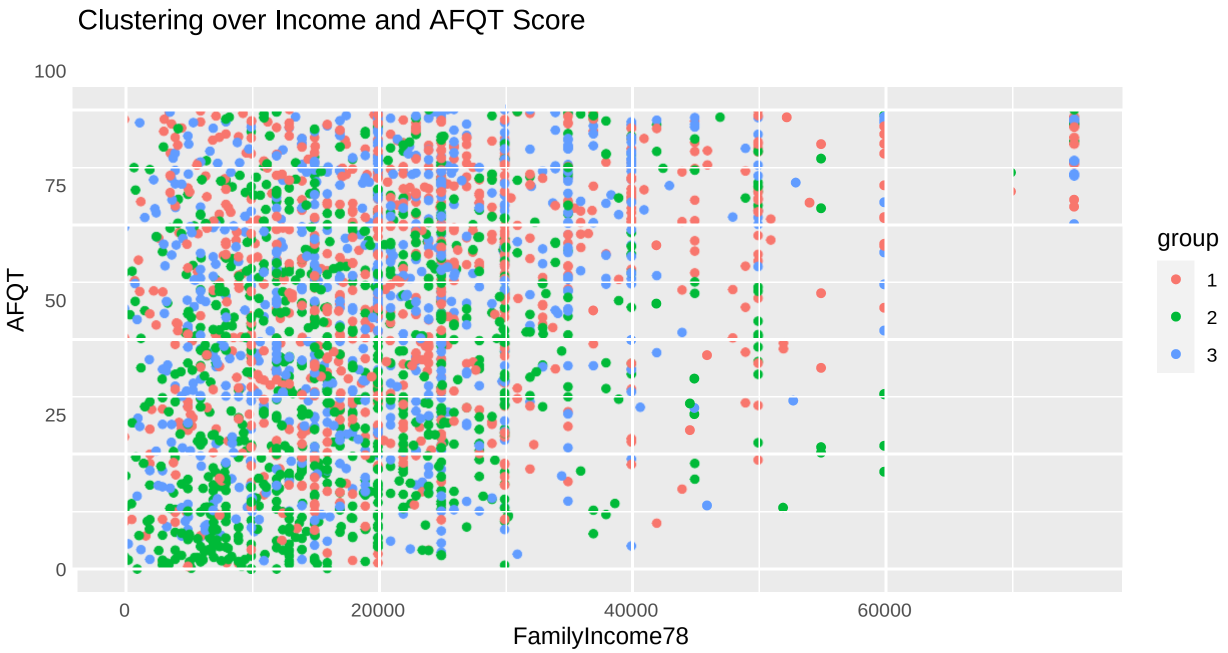
The size of the three clusters turn out to be 826, 818, and 787, respectively. The three clusters have similar spreads in terms of income, with cluster 3 being slightly more left skewed, followed by cluster 2 and cluster 1. As for AFQT, group 1 is very left skewed with most individuals having higher scores, group 3 is comparatively left-skewed, while group 2 is right-skewed and more uniform with more people scoring lower.

c) Can you visualize the clusters with somewhat clear boundaries? You may try different pairs of variab

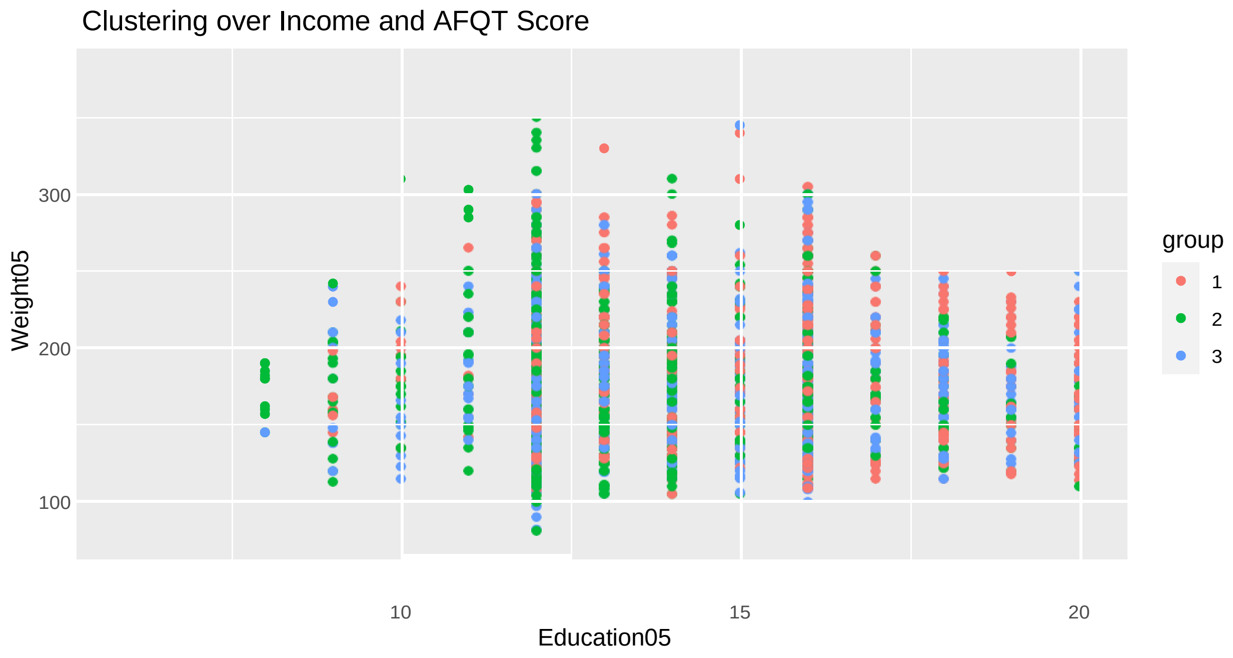
```
esteem.pca <- prcomp(data.esteem[,c(37, 38, 39, 40, 41, 42, 43, 44, 45, 46)],center= TRUE, scale = TRUE
library("ggplot2")
esteem.final1 <- data.frame(pc1 = esteem.pca$x[,1], pc2 = esteem.pca$x[,2],
          group = as.factor(esteem.kmeans$cluster))

ggplot(data = esteem.final1, aes(x = pc1, y = pc2, col=group)) + geom_point() + ggtitle("Clustering ove
```



```
esteem.final2 <- data.frame(Income = data.esteem["FamilyIncome78"], AFQT = data.esteem["AFQT"], group =
ggplot(data = esteem.final2, aes(x = FamilyIncome78, y = AFQT, col=group)) + geom_point() + ggtitle("Cl
```



```
esteem.final2 <- data.frame(Education = data.esteem["Education05"], AFQT = data.esteem["Weight05"], gro
ggplot(data = esteem.final2, aes(x = Education05, y = Weight05, col=group)) + geom_point() + ggtitle("C
```



As shown in the plots above, total esteem score is very differentiated across the different clusters. Cluster 1 has the highest esteem, followed by cluster 3, and then cluster 2. However, there is no clear differentiation among the 3 clusters in terms of variables such as weight, education, and income.

6. We now try to find out what factors are related to self-esteem? PC1 of all the Esteem scores is a good variable to summarize one's esteem scores. We take PC1 as our response variable.

   a) Prepare possible factors/variables:

- Personal information: gender, education (05, problematic), log(income) in 87, job type in 87, Body mass index as a measure of health (The BMI is defined as the body mass divided by the square of the body height, and is universally expressed in units of kg/m2). Since BMI is measured in 05, this will not be a good practice to be inclueded as possible variables.
- Household environment: Imagazine, Inewspaper, Ilibrary, MotherEd, FatherEd, FamilyIncome78. Do set indicators Imagazine, Inewspaper and Ilibrary as factors.
- Use PC1 of SVABS as level of intelligence

```
data.esteem.pca <- data.esteem %>%
  select(Gender, Education05, Job05,MotherEd, FatherEd, FamilyIncome78) %>%
  mutate(data.esteem, log_income = log(Income87+2)) %>%
  mutate(data.esteem, bmi = Weight05/(HeightFeet05^2))

data.esteem.pca <- data.esteem.pca %>%
  select(Gender, Education05, Job05,MotherEd, FatherEd, FamilyIncome78, bmi, log_income, Esteem87_1, Es

data.esteem.pca <- data.esteem.pca %>%
  mutate(pc1_esteem = 0.324*Esteem87_1 + 0.333*Esteem87_2 + 0.322*Esteem87_3 + 0.324*Esteem87_4 + 0.315

data.esteem.pca <- data.esteem.pca %>%
  mutate(log_income = log_income + 0.00000001)
```

b) Run a few regression models between PC1 of all the esteem scores and suitable variables listed in

- How did you land this model? Run a model diagnosis to see if the linear model assumptions are reaso

- Write a summary of your findings. In particular, explain what and how the variables in the model af

```
rgr.data <- na.omit(data.esteem.pca)
rgr.data <- rgr.data %>%
  filter(!is.infinite(rgr.data$log_income))

model1 <- lm(pc1_esteem ~ Gender+ Education05+Job05+MotherEd+FatherEd+FamilyIncome78+bmi+log_income, rg
summary(model1)
```

```
##
## Call:
## lm(formula = pc1_esteem ~ Gender + Education05 + Job05 + MotherEd +
##     FatherEd + FamilyIncome78 + bmi + log_income, data = rgr.data)
##
## Residuals:
##    Min     1Q Median     3Q    Max ##
-5.176  -0.944  -0.005   0.993   2.865##
## Coefficients:
##                                                                              Estimate
## (Intercept)                                                                  8.74e+00
## Gendermale                                                                   1.65e-01
## Education05                                                                  8.03e-02
## Job0510 TO 430: Executive, Administrative and Managerial Occupations         4.53e-01
## Job051000 TO 1240: Mathematical and Computer Scientists                      5.05e-01
## Job051300 TO 1560: Engineers, Architects, Surveyers, Engineering and Related Technicians 2.32e-02##
Job051600 TO 1760: Physical Scientists                                      -7.90e-01
## Job051800 TO 1860: Social Scientists and Related Workers                    -3.07e-01
## Job051900 TO 1960: Life, Physical and Social Science Technicians             2.75e-01
## Job052000 TO 2060: Counselors, Sociala and Religious Workers                 1.56e-01
## Job052100 TO 2150: Lawyers, Judges and Legal Support Workers                 2.49e-01
## Job052200 TO 2340: Teachers                                                  1.92e-01
## Job052400 TO 2550: Education, Training and Library Workers                   1.96e-01
## Job052600 TO 2760: Entertainers and Performers, Sports and Related Workers   7.62e-01
## Job052800 TO 2960: Media and Communications Workers                          3.67e-01
## Job053000 TO 3260: Health Diagnosing and Treating Practitioners              4.63e-01
## Job053300 TO 3650: Health Care Technical and Support Occupations            -1.13e-01
## Job053700 TO 3950: Protective Service Occupations                            5.67e-01
## Job054000 TO 4160: Food Preparation and Serving Related Occupations         -1.43e-01
## Job054200 TO 4250: Cleaning and Building Service Occupations                -2.59e-01
## Job054300 TO 4430: Entertainment Attendants and Related Workers             -6.97e-01
## Job054500 TO 4650: Personal Care and Service Workers                         2.71e-01
## Job054700 TO 4960: Sales and Related Workers                                 2.84e-01
## Job05500 TO 950: Management Related Occupations                              5.27e-01
## Job055000 TO 5930: Office and Administrative Support Workers                 2.97e-01
## Job056000 TO 6130: Farming, Fishing and Forestry Occupations                -1.75e-01
## Job056200 TO 6940: Construction Trade and Extraction Workers                 2.88e-02
## Job057000 TO 7620: Installation, Maintenance and Repairs Workers             1.42e-01
## Job057700 TO 7750: Production and Operating Workers                          1.87e-01
## Job057800 TO 7850: Food Preparation Occupations                              1.89e-01
## Job057900 TO 8960: Setters, Operators and Tenders                            1.39e-01
## Job059000 TO 9750: Transportation and Material Moving Workers                    -1.13e-01
## Job059990: Uncodeable                                                            -5.12e-02
## MotherEd                                                                          2.81e-02
## FatherEd                                                                          1.03e-02
## FamilyIncome78                                                                    2.94e-06
## bmi                                                                              -1.42e-02
## log_income                                                                        2.37e-02
##                                                                                  Std. Error
## (Intercept)                                                                       2.66e-01
## Gendermale                                                                        5.91e-02
## Education05                                                                       1.28e-02
## Job0510 TO 430: Executive, Administrative and Managerial Occupations              1.73e-01
## Job051000 TO 1240: Mathematical and Computer Scientists                           2.22e-01
## Job051300 TO 1560: Engineers, Architects, Surveyers, Engineering and Related Technicians  2.33e-01
## Job051600 TO 1760: Physical Scientists                                            6.24e-01
## Job051800 TO 1860: Social Scientists and Related Workers                          5.18e-01
## Job051900 TO 1960: Life, Physical and Social Science Technicians                  4.82e-01
## Job052000 TO 2060: Counselors, Sociala and Religious Workers                      2.51e-01
## Job052100 TO 2150: Lawyers, Judges and Legal Support Workers                      3.52e-01
## Job052200 TO 2340: Teachers                                                       2.00e-01
## Job052400 TO 2550: Education, Training and Library Workers                        2.76e-01
## Job052600 TO 2760: Entertainers and Performers, Sports and Related Workers        2.95e-01
## Job052800 TO 2960: Media and Communications Workers                               3.71e-01
## Job053000 TO 3260: Health Diagnosing and Treating Practitioners                   2.16e-01
## Job053300 TO 3650: Health Care Technical and Support Occupations                  2.02e-01
## Job053700 TO 3950: Protective Service Occupations                                 2.30e-01
## Job054000 TO 4160: Food Preparation and Serving Related Occupations               2.18e-01
## Job054200 TO 4250: Cleaning and Building Service Occupations                      2.20e-01
## Job054300 TO 4430: Entertainment Attendants and Related Workers                   4.14e-01
## Job054500 TO 4650: Personal Care and Service Workers                              2.46e-01
## Job054700 TO 4960: Sales and Related Workers                                      1.82e-01
## Job05500 TO 950: Management Related Occupations                                   1.99e-01
## Job055000 TO 5930: Office and Administrative Support Workers                      1.73e-01
## Job056000 TO 6130: Farming, Fishing and Forestry Occupations                      4.34e-01
## Job056200 TO 6940: Construction Trade and Extraction Workers                      1.95e-01
## Job057000 TO 7620: Installation, Maintenance and Repairs Workers                  2.01e-01
## Job057700 TO 7750: Production and Operating Workers                               2.38e-01
## Job057800 TO 7850: Food Preparation Occupations                                   6.23e-01
## Job057900 TO 8960: Setters, Operators and Tenders                                 1.99e-01
## Job059000 TO 9750: Transportation and Material Moving Workers                     1.98e-01
## Job059990: Uncodeable                                                             1.21e+00
## MotherEd                                                                          1.25e-02
## FatherEd                                                                          9.22e-03
## FamilyIncome78                                                                    1.94e-06
## bmi                                                                               1.40e-02
## log_income                                                                        8.59e-03
##                                                                                  t value
## (Intercept)                                                                       32.87
## Gendermale                                                                         2.79
## Education05                                                                        6.25
## Job0510 TO 430: Executive, Administrative and Managerial Occupations               2.62
## Job051000 TO 1240: Mathematical and Computer Scientists                            2.27
## Job051300 TO 1560: Engineers, Architects, Surveyers, Engineering and Related Technicians  0.10
## Job051600 TO 1760: Physical Scientists                                            -1.27
## Job051800 TO 1860: Social Scientists and Related Workers                                      -0.59
## Job051900 TO 1960: Life, Physical and Social Science Technicians                                0.57
## Job052000 TO 2060: Counselors, Sociala and Religious Workers                                    0.62
## Job052100 TO 2150: Lawyers, Judges and Legal Support Workers                                    0.71
## Job052200 TO 2340: Teachers                                                                     0.96
## Job052400 TO 2550: Education, Training and Library Workers                                      0.71
## Job052600 TO 2760: Entertainers and Performers, Sports and Related Workers                      2.59
## Job052800 TO 2960: Media and Communications Workers                                             0.99
## Job053000 TO 3260: Health Diagnosing and Treating Practitioners                                 2.14
## Job053300 TO 3650: Health Care Technical and Support Occupations                               -0.56
## Job053700 TO 3950: Protective Service Occupations                                               2.46
## Job054000 TO 4160: Food Preparation and Serving Related Occupations                            -0.65
## Job054200 TO 4250: Cleaning and Building Service Occupations                                   -1.18
## Job054300 TO 4430: Entertainment Attendants and Related Workers                                -1.68
## Job054500 TO 4650: Personal Care and Service Workers                                            1.10
## Job054700 TO 4960: Sales and Related Workers                                                    1.56
## Job05500 TO 950: Management Related Occupations                                                 2.64
## Job055000 TO 5930: Office and Administrative Support Workers                                    1.71
## Job056000 TO 6130: Farming, Fishing and Forestry Occupations                                   -0.40
## Job056200 TO 6940: Construction Trade and Extraction Workers                                    0.15
## Job057000 TO 7620: Installation, Maintenance and Repairs Workers                                0.70
## Job057700 TO 7750: Production and Operating Workers                                             0.79
## Job057800 TO 7850: Food Preparation Occupations                                                 0.30
## Job057900 TO 8960: Setters, Operators and Tenders                                               0.70
## Job059000 TO 9750: Transportation and Material Moving Workers                                  -0.57
## Job059990: Uncodeable                                                                          -0.04
## MotherEd                                                                                        2.25
## FatherEd                                                                                        1.11
## FamilyIncome78                                                                                  1.52
## bmi                                                                                            -1.01
## log_income                                                                                      2.76
##                                                                                             Pr(>|t|)
## (Intercept)                                                                                 < 2e-16
## Gendermale                                                                                   0.0053
## Education05                                                                                 4.9e-10
## Job0510 TO 430: Executive, Administrative and Managerial Occupations                         0.0090
## Job051000 TO 1240: Mathematical and Computer Scientists                                      0.0232
## Job051300 TO 1560: Engineers, Architects, Surveyers, Engineering and Related Technicians     0.9209
## Job051600 TO 1760: Physical Scientists                                                       0.2059
## Job051800 TO 1860: Social Scientists and Related Workers                                     0.5529
## Job051900 TO 1960: Life, Physical and Social Science Technicians                             0.5691
## Job052000 TO 2060: Counselors, Sociala and Religious Workers                                 0.5338
## Job052100 TO 2150: Lawyers, Judges and Legal Support Workers                                 0.4792
## Job052200 TO 2340: Teachers                                                                  0.3392
## Job052400 TO 2550: Education, Training and Library Workers                                   0.4768
## Job052600 TO 2760: Entertainers and Performers, Sports and Related Workers                   0.0097
## Job052800 TO 2960: Media and Communications Workers                                          0.3228
## Job053000 TO 3260: Health Diagnosing and Treating Practitioners                              0.0323
## Job053300 TO 3650: Health Care Technical and Support Occupations                             0.5759
## Job053700 TO 3950: Protective Service Occupations                                            0.0140
## Job054000 TO 4160: Food Preparation and Serving Related Occupations                          0.5127
## Job054200 TO 4250: Cleaning and Building Service Occupations                                 0.2379
## Job054300 TO 4430: Entertainment Attendants and Related Workers                              0.0926
## Job054500 TO 4650: Personal Care and Service Workers                                         0.2699
## Job054700 TO 4960: Sales and Related Workers                                 0.1190
## Job05500 TO 950: Management Related Occupations                              0.0082
## Job055000 TO 5930: Office and Administrative Support Workers                 0.0866
## Job056000 TO 6130: Farming, Fishing and Forestry Occupations                 0.6864
## Job056200 TO 6940: Construction Trade and Extraction Workers                 0.8830
## Job057000 TO 7620: Installation, Maintenance and Repairs Workers             0.4816
## Job057700 TO 7750: Production and Operating Workers                          0.4325
## Job057800 TO 7850: Food Preparation Occupations                              0.7612
## Job057900 TO 8960: Setters, Operators and Tenders                            0.4839
## Job059000 TO 9750: Transportation and Material Moving Workers                0.5667
## Job059990: Uncodeable                                                        0.9664
## MotherEd                                                                     0.0242
## FatherEd                                                                     0.2660
## FamilyIncome78                                                               0.1290
## bmi                                                                          0.3106
## log_income                                                                   0.0058
##
## (Intercept)                                                                ***
## Gendermale                                                                 **
## Education05                                                                ***
## Job0510 TO 430: Executive, Administrative and Managerial Occupations       **
## Job051000 TO 1240: Mathematical and Computer Scientists                    *
## Job051300 TO 1560: Engineers, Architects, Surveyers, Engineering and Related Technicians
## Job051600 TO 1760: Physical Scientists
## Job051800 TO 1860: Social Scientists and Related Workers
## Job051900 TO 1960: Life, Physical and Social Science Technicians
## Job052000 TO 2060: Counselors, Sociala and Religious Workers
## Job052100 TO 2150: Lawyers, Judges and Legal Support Workers
## Job052200 TO 2340: Teachers
## Job052400 TO 2550: Education, Training and Library Workers
## Job052600 TO 2760: Entertainers and Performers, Sports and Related Workers **
## Job052800 TO 2960: Media and Communications Workers
## Job053000 TO 3260: Health Diagnosing and Treating Practitioners            *
## Job053300 TO 3650: Health Care Technical and Support Occupations
## Job053700 TO 3950: Protective Service Occupations                          *
## Job054000 TO 4160: Food Preparation and Serving Related Occupations
## Job054200 TO 4250: Cleaning and Building Service Occupations
## Job054300 TO 4430: Entertainment Attendants and Related Workers            .
## Job054500 TO 4650: Personal Care and Service Workers
## Job054700 TO 4960: Sales and Related Workers
## Job05500 TO 950: Management Related Occupations                            **
## Job055000 TO 5930: Office and Administrative Support Workers               .
## Job056000 TO 6130: Farming, Fishing and Forestry Occupations
## Job056200 TO 6940: Construction Trade and Extraction Workers
## Job057000 TO 7620: Installation, Maintenance and Repairs Workers
## Job057700 TO 7750: Production and Operating Workers
## Job057800 TO 7850: Food Preparation Occupations
## Job057900 TO 8960: Setters, Operators and Tenders
## Job059000 TO 9750: Transportation and Material Moving Workers
## Job059990: Uncodeable
## MotherEd                                                                   *
## FatherEd
## FamilyIncome78
## bmi
## log_income                                                                                    **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.2 on 2375 degrees of freedom
## Multiple R-squared: 0.118,  Adjusted R-squared:  0.104 ##
F-statistic:  8.58 on 37 and 2375 DF, p-value: <2e-16
```

```
model2 <- lm(pc1_esteem ~ Gender+ Education05+MotherEd+log_income, rgr.data)
summary(model2)
```

```
##
## Call:
## lm(formula = pc1_esteem ~ Gender + Education05 + MotherEd + log_income,
##       data = rgr.data)
##
## Residuals:
##     Min     1Q Median     3Q     Max ##
-5.530  -0.991  -0.005   1.013   2.605##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)   8.40594    0.16194   51.91  < 2e-16 ***
## Gendermale    0.12546    0.05011    2.50   0.01235 *
## Education05   0.10840    0.01094    9.91   < 2e-16 ***
## MotherEd      0.04592    0.01049    4.38   1.3e-05  ***
## log_income    0.03169    0.00856    3.70   0.00022 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.22 on 2408 degrees of freedom
## Multiple R-squared: 0.0842, Adjusted R-squared: 0.0827 ##
F-statistic:  55.4 on 4 and 2408 DF, p-value: <2e-16
```

We finalized on this model after tuning our logistic regression model by eliminating coefficients that were statistically insignificant. Our results show that having higher education, being a male, having a mother from a higher education, and higher income are all factors that boost self-esteem. On the other hand, we found that BMI, family_income, and father's education do not significantly contribute. We eliminated Job05 due to an influx of categorical variables post-one hot encoding, but found it rather intriguing that self esteem was significantly positively affected if you were an executive, in entertainer, math and CS, or in protective services.

2 Case study 2: Breast cancer sub-type

The Cancer Genome Atlas (TCGA), a landmark cancer genomics program by National Cancer Institute (NCI), molecularly characterized over 20,000 primary cancer and matched normal samples spanning 33 cancer types. The genome data is open to public from the Genomic Data Commons Data Portal (GDC).

In this study, we focus on 4 sub-types of breast cancer (BRCA): basal-like (basal), Luminal A-like (lumA), Luminal B-like (lumB), HER2-enriched. The sub-type is based on PAM50, a clinical-grade luminal-basal classifier.

- Luminal A cancers are low-grade, tend to grow slowly and have the best prognosis.
- Luminal B cancers generally grow slightly faster than luminal A cancers and their prognosis is slightly worse.
- HER2-enriched cancers tend to grow faster than luminal cancers and can have a worse prognosis, but they are often successfully treated with targeted therapies aimed at the HER2 protein.
- Basal-like breast cancers or triple negative breast cancers do not have the three receptors that the other sub-types have so have fewer treatment options.

We will try to use mRNA expression data alone without the labels to classify 4 sub-types. Classification without labels or prediction without outcomes is called unsupervised learning. We will use K-means and spectrum clustering to cluster the mRNA data and see whether the sub-type can be separated through mRNA data.

We first read the data using data.table::fread() which is a faster way to read in big data than read.csv().

1. Summary and transformation

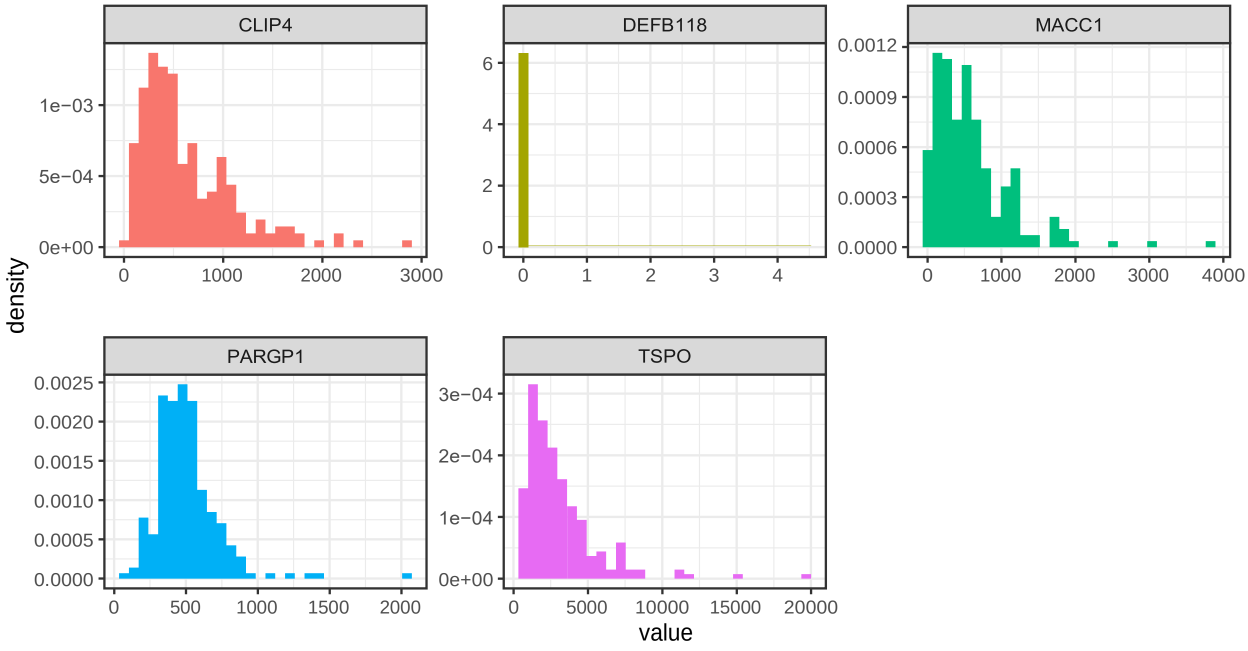
    a) How many patients are there in each sub-type?

```
brca <- fread("data/brca_subtype.csv")

# get the sub-type information
brca_subtype <- brca$BRCA_Subtype_PAM50
brca <- brca[,-1]

table(brca_subtype)
```

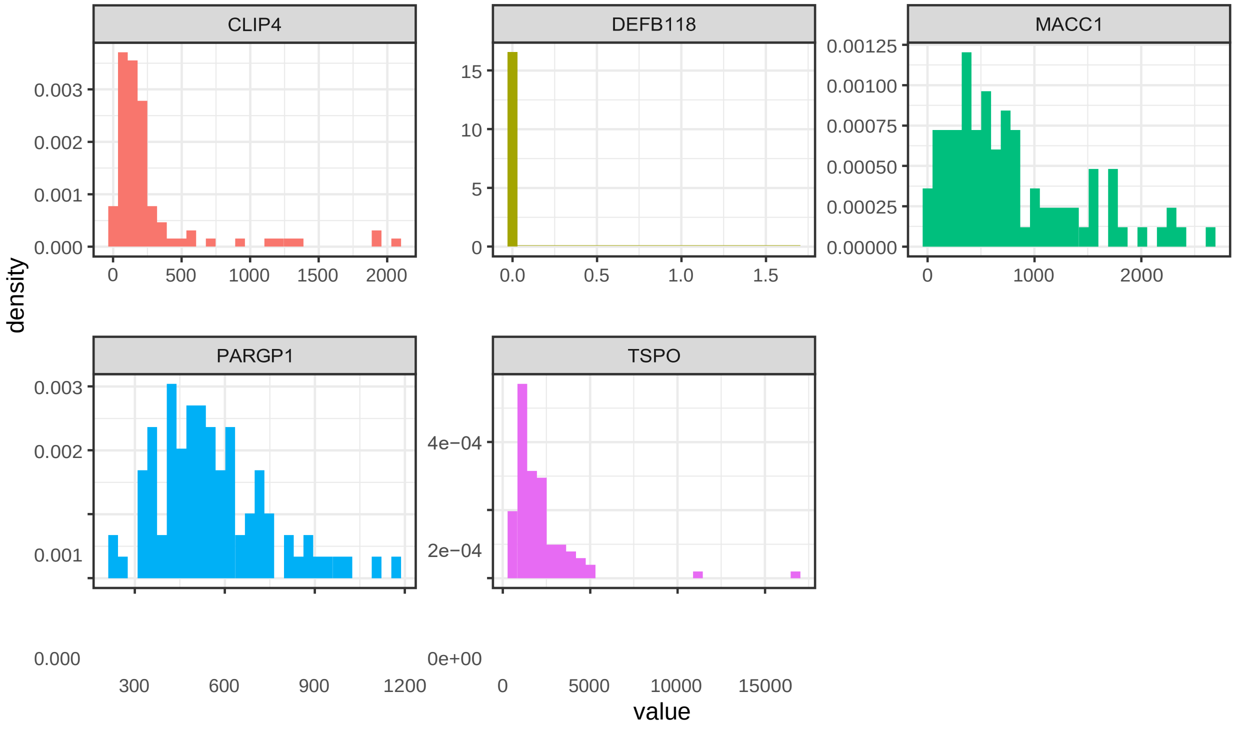
```
## brca_subtype
##  Basal  Her2  LumA  LumB
##    208    91   628   233
```

b) Randomly pick 5 genes and plot the histogram by each sub-type.

```
set.seed(10)
brca_sample_idx <- sample(ncol(brca), 5)
basal_indices <- which(brca_subtype == "Basal")
her2_indices <- which(brca_subtype == "Her2")
luma_indices <- which(brca_subtype == "LumA")
lumb_indices <- which(brca_subtype == "LumB")
brca[basal_indices,] %>%
  select(all_of(brca_sample_idx)) %>%        # select column by index
  pivot_longer(cols = everything()) %>%       # for facet(0)
  ggplot(aes(x = value, y = ..density..)) +
  geom_histogram(aes(fill = name)) +
  facet_wrap(~name, scales = "free") +
  theme_bw() +
  theme(legend.position = "none")
```

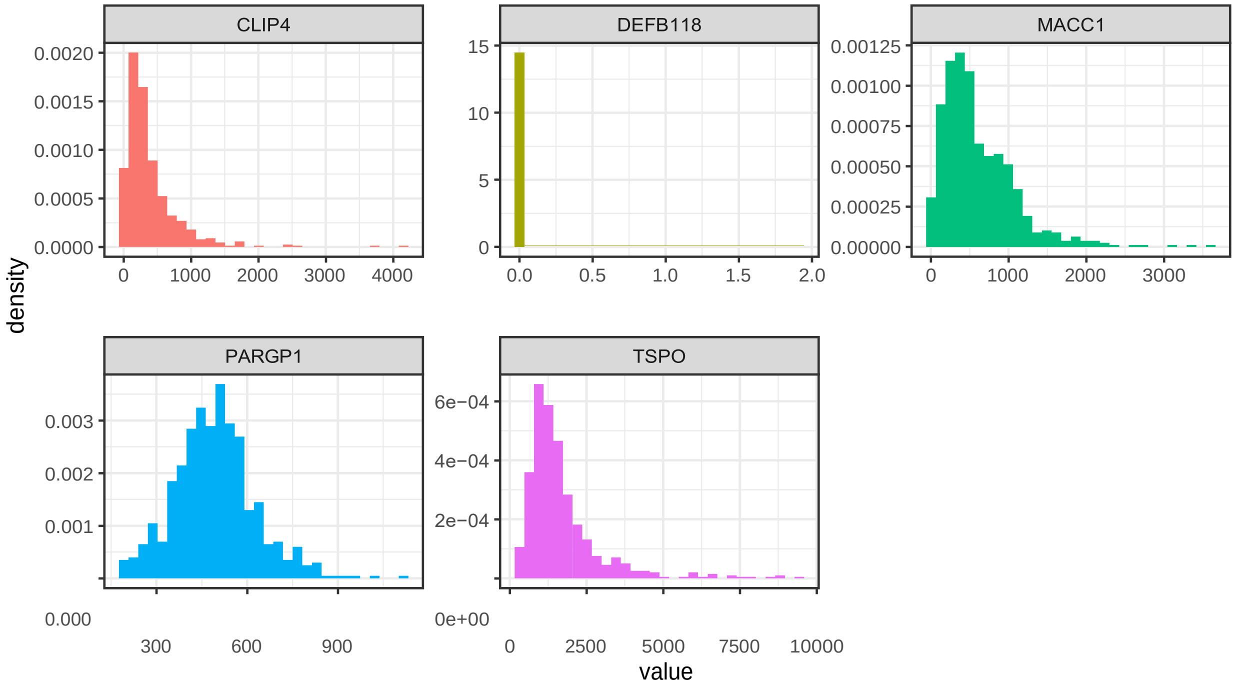


```
brca[her2_indices,]   %>%
   select(all_of(brca_sample_idx)) %>%          # select column by index
   pivot_longer(cols = everything()) %>%          # for facet(0)
   ggplot(aes(x  =  value,  y =  ..density..))  +
   geom_histogram(aes(fill = name)) +
   facet_wrap(~name, scales = "free") +
   theme_bw()  +
   theme(legend.position = "none")
```

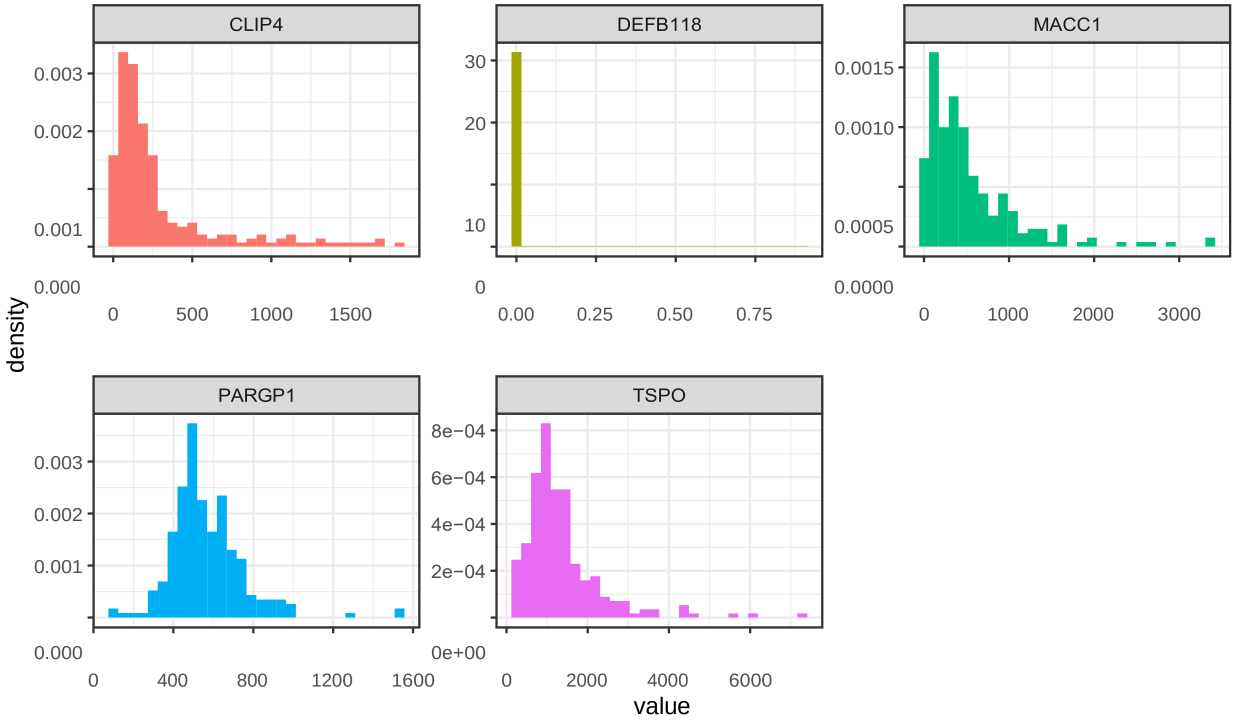


```
brca[luma_indices,] %>%
  select(all_of(brca_sample_idx)) %>%           # select column by index
  pivot_longer(cols = everything())  %>%         #  for  facet(0)
```

```
    ggplot(aes(x = value, y = ..density..)) +
    geom_histogram(aes(fill = name)) +
    facet_wrap(~name, scales = "free") +
    theme_bw() +
    theme(legend.position = "none")
```



```
brca[lumb_indices,]  %>%
    select(all_of(brca_sample_idx)) %>%         # select column by index
    pivot_longer(cols = everything()) %>%        # for facet(0)
    ggplot(aes(x  =  value,  y =  ..density..)) +
    geom_histogram(aes(fill = name)) +
    facet_wrap(~name, scales = "free") +
    theme_bw() +
    theme(legend.position = "none")
```



c) Remove gene with zero count and no variability. Then apply logarithmic transform.

```
require(caret)
dim(brca)
```

```
## [1]  1160 19947
```

```
# remove genes with 0 counts
sel_cols <- which(colSums(abs(brca)) != 0)
brca_sub <- brca[, sel_cols, with=F]
dim(brca_sub)
```

```
## [1]  1160 19669
```

```
# remove genes with no variability (SD=0)
# after removing 0 counts, there are no genes/cols with all same values
# brca_sub[,-nearZeroVar(brca_sub)]
dim(brca_sub)
```

```
## [1]  1160 19669
```

```
# log
brca_sub <- log2(as.matrix(brca_sub+1e-10))
```

2. Apply kmeans on the transformed dataset with 4 centers and output the discrepancy table between the real sub-type brca_subtype and the cluster labels.

```
system.time({brca_sub_kmeans <- kmeans(x = brca_sub, 4)})
```

```
##      user    system  elapsed
##   11.420     0.212    11.689
```

```
# save the results as RDS
saveRDS(brca_sub_kmeans, "data/brca_kmeans.RDS")

# read in tcga_sub_kmeans
brca_sub_kmeans <- readRDS("data/brca_kmeans.RDS")

# discrepancy table
table(brca_subtype, brca_sub_kmeans$cluster)
```

```
##
## brca_subtype   1   2   3   4
##        Basal   1  17   3 187
##         Her2  39   9  26  17
##         LumA 392  82 154   0
##         LumB  98  22 111   2
```

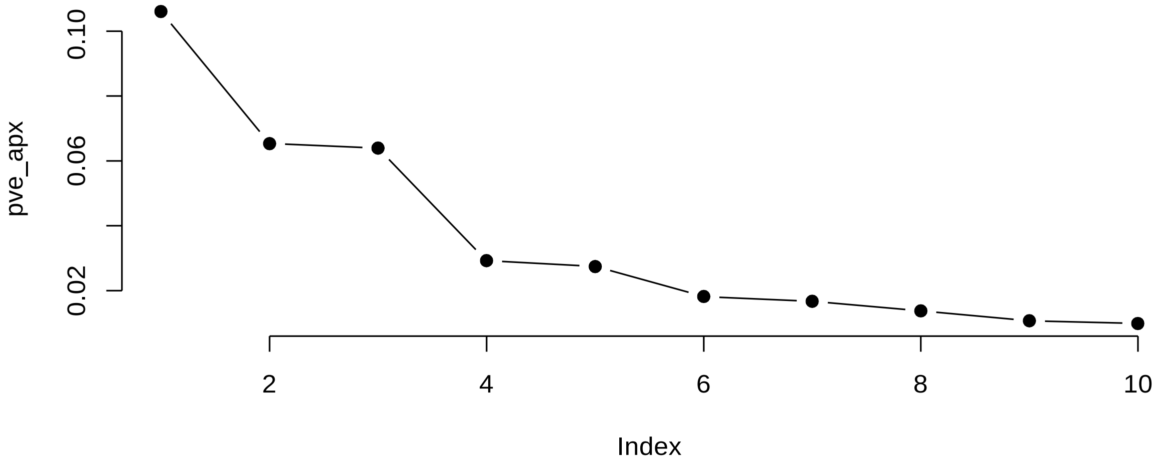
3. Spectrum clustering: to scale or not to scale?

   a) Apply PCA on the centered and scaled dataset. How many PCs should we use and why? You are encouraged to use irlba::irlba().

```
require("irlba")
# center and scale the data
brca_sub_scaled_centered <- scale(as.matrix(brca_sub), center = T, scale = T)
svd_ret <- irlba::irlba(brca_sub_scaled_centered, nv = 10)
names(svd_ret)
```

```
## [1] "d"     "u"     "v"     "iter"  "mprod"
```

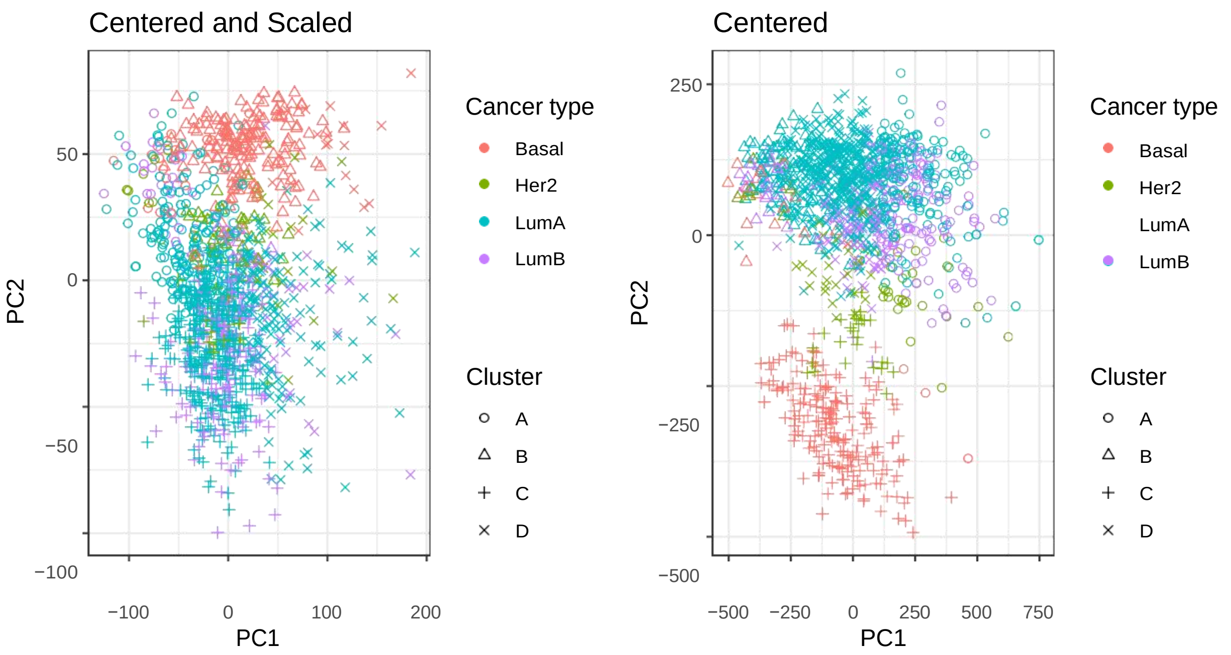
```
# Approximate the PVE
svd_var <- svd_ret$d^2/(nrow(brca_sub_scaled_centered)-1)
pve_apx <- svd_var/ncol(brca)
plot(pve_apx, type="b", pch = 19, frame = FALSE)
```



We may look at the scree plot of PVE's and apply elbow rules: take the number of PC's when there is a sharp drop in the scree plot. We can either choose 2 or 4 PC, in this case, to capture more cumnulative explained variance, we choose 4 PC.

b) Plot PC1 vs PC2 of the centered and scaled data and PC1 vs PC2 of the centered but unscaled data sid

```
require("gridExtra")
require("grid")

# get pc score
pc_score <- brca_sub_scaled_centered %*% svd_ret$v[, 1:3]

# apply kmean
kmean_ret <- kmeans(x = pc_score, 4)

p1 <- data.table(x = pc_score[,1],
                 y = pc_score[,2],
                 col = as.factor(brca_subtype),
                 cl = as.factor(kmean_ret$cluster)) %>%
  ggplot() +
  geom_point(aes(x = x, y = y, col = col, shape = cl)) +
  scale_color_manual(labels = c("Basal", "Her2", "LumA", "LumB"),
                     values = scales::hue_pal()(4)) +
  scale_shape_manual(labels = c("A", "B", "C", "D"),
                     values = c(1, 2, 3, 4)) +
  theme_bw() +
  labs(color = "Cancer type", shape = "Cluster") +
  xlab("PC1") +
  ylab("PC2")  +
  ggtitle("Centered and Scaled")

brca_sub_centered <- scale(as.matrix(brca_sub), center = T, scale = F)
svd_ret <- irlba::irlba(brca_sub_centered, nv = 10)

# Approximate the PVE
svd_var <- svd_ret$d^2/(nrow(brca_sub_centered)-1)
pve_apx <- svd_var/ncol(brca) # plot also shows 4 PC by elbow method

pc_score <- brca_sub_centered %*% svd_ret$v[, 1:3]

# apply kmean
kmean_ret <- kmeans(x = pc_score, 4)

p2 <- data.table(x = pc_score[,1],
                 y = pc_score[,2],
                 col = as.factor(brca_subtype),
                 cl = as.factor(kmean_ret$cluster)) %>%
  ggplot() +
  geom_point(aes(x = x, y = y, col = col, shape = cl)) +
  scale_color_manual(labels = c("Basal", "Her2", "LumA", "LumB"),
                     values = scales::hue_pal()(4)) +
  scale_shape_manual(labels = c("A", "B", "C", "D"),
                     values = c(1, 2, 3, 4)) +
  theme_bw() +
  labs(color = "Cancer type", shape = "Cluster") +
  xlab("PC1")  +
  ylab("PC2") +
  ggtitle("Centered")

grid.arrange(p1, p2, nrow  =  1)
```

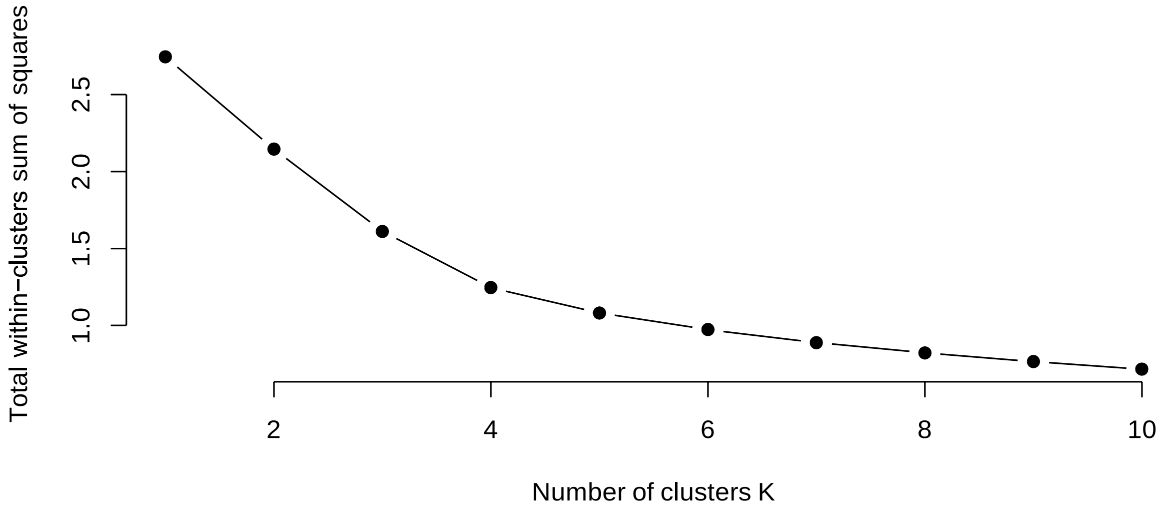


In our case, scaling does not seem to be necessary, since both plots maintain similar cluster shapes. However, in general, clusterings is scale-sensitive when dealing with features of different units, because the Euclidean distance algorithm will weigh variables with higher numbers more. Since in our case, there is only one data type, scaling is not that necessary.

4. Spectrum clustering: center but do not scale the data

   a) Use the first 4 PCs of the centered and unscaled data and apply kmeans. Find a reasonable number of clusters using within sum of squared with the elbow rule.

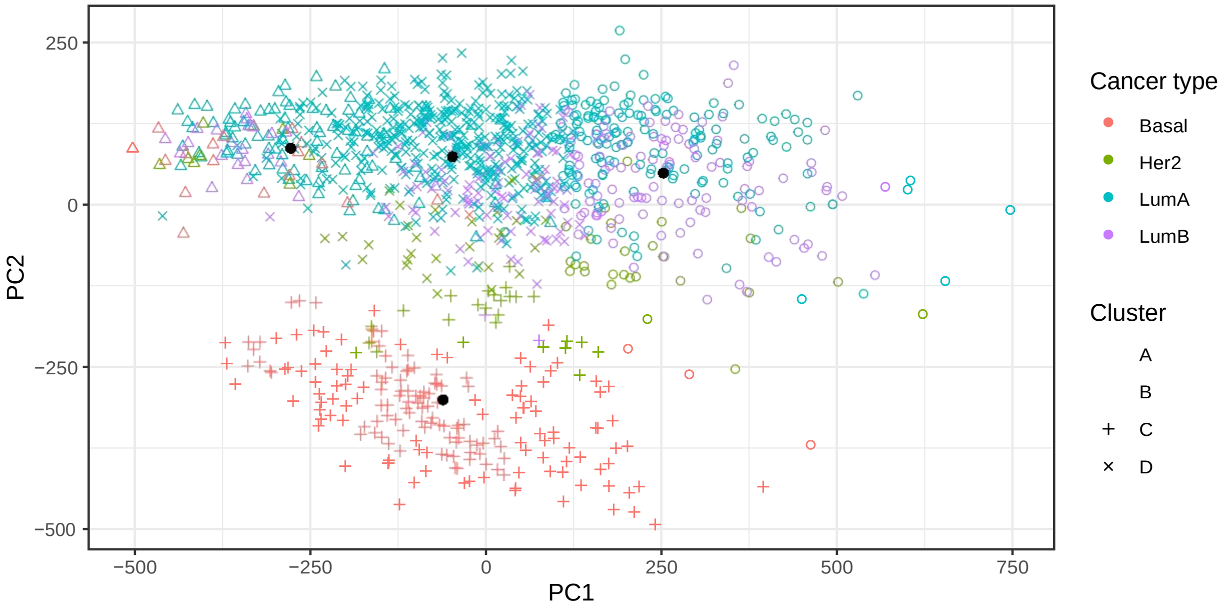
```
k.values <- 1:10

# function to compute total within-cluster sum of square
wss <- function(df, k) {
  kmeans(df, k, nstart = 10)$tot.withinss
}

# extract wss for 1:10 clusters
wss_values <- map_dbl(k.values, function(k) wss(svd_ret$v[, 1:3], k))
plot(k.values, wss_values,
       type="b", pch = 19, frame = FALSE,
       xlab="Number of clusters K",
       ylab="Total within-clusters sum of squares")
```



Elbow rule agrees that a good number of clusters is 4.

b) Choose an optimal cluster number and apply kmeans. Compare the real sub-type and the clustering labe

```
p2 <- data.table(x  =  pc_score[,1],
                 y  =  pc_score[,2],
                 col = as.factor(brca_subtype),
                 cl = as.factor(kmean_ret$cluster))  %>%
  ggplot() +
  geom_point(aes(x = x, y = y, col = col, shape = cl)) +
  scale_color_manual(labels = c("Basal", "Her2", "LumA", "LumB"),
                     values = scales::hue_pal()(4)) +
  scale_shape_manual(labels = c("A", "B", "C", "D"),
                     values = c(1, 2, 3, 4)) +
  theme_bw() +
  labs(color = "Cancer type", shape = "Cluster") +
  xlab("PC1") +
  ylab("PC2")   +
  geom_point(aes(x=kmean_ret$centers[1,1], y=kmean_ret$centers[1,2]), colour="black") +
  geom_point(aes(x=kmean_ret$centers[2,1], y=kmean_ret$centers[2,2]), colour="black") +
  geom_point(aes(x=kmean_ret$centers[3,1], y=kmean_ret$centers[3,2]), colour="black") +
  geom_point(aes(x=kmean_ret$centers[4,1], y=kmean_ret$centers[4,2]), colour="black")
p2
```



Clustering result compared to the real sub-type is good for one of the sub-types, but the other three are very hard to distinguish.

c) Compare the clustering result from applying kmeans to the original data and the clustering result fr

```
table(brca_subtype,  kmean_ret$cluster)
```

```
##
## brca_subtype   1   2   3   4
##        Basal   3  17 187   1
##         Her2  27   9  26  29
##         LumA 161  91   0 376
##         LumB 108  22   2 101
```

Compared to the results from applying kmeans to the original data, PCA helps a little bit, with more distinguishability in the 4 clusters. PCA reduces the dimensionality of the data but still retains the information and variance of the data, which means that kmeans will take less time to train. Additionally, PCA helps whiten the data and normalize/scale it, and since kmeans is sensitive to these aspects, performance will be increased.

d) Now we have an x patient with breast cancer but with unknown sub-type. We have this patient’s mRNA

```
#unscaled data
pca_unscaled <- prcomp(brca_sub, center = T, scale. = F)
pca_unscaled$rotation<- pca_unscaled$rotation[, 1:20]
pca_unscaled$x <- pca_unscaled$x[, 1:20]
pve_unscaled <- summary(pca_unscaled)$importance[2, 1:10]

kmeans_unscaled <- kmeans(x = pca_unscaled$x[,1:4], 4)

df_unscaled<-(cbind.data.frame(PC1=pca_unscaled$x[,1],
                               PC2=pca_unscaled$x[,2],
                                    brca_subtype,
                                    cluster=as.factor(kmeans_unscaled$cluster)))

x_patient <- fread("data/brca_x_patient.csv")

dim(x_patient)
```

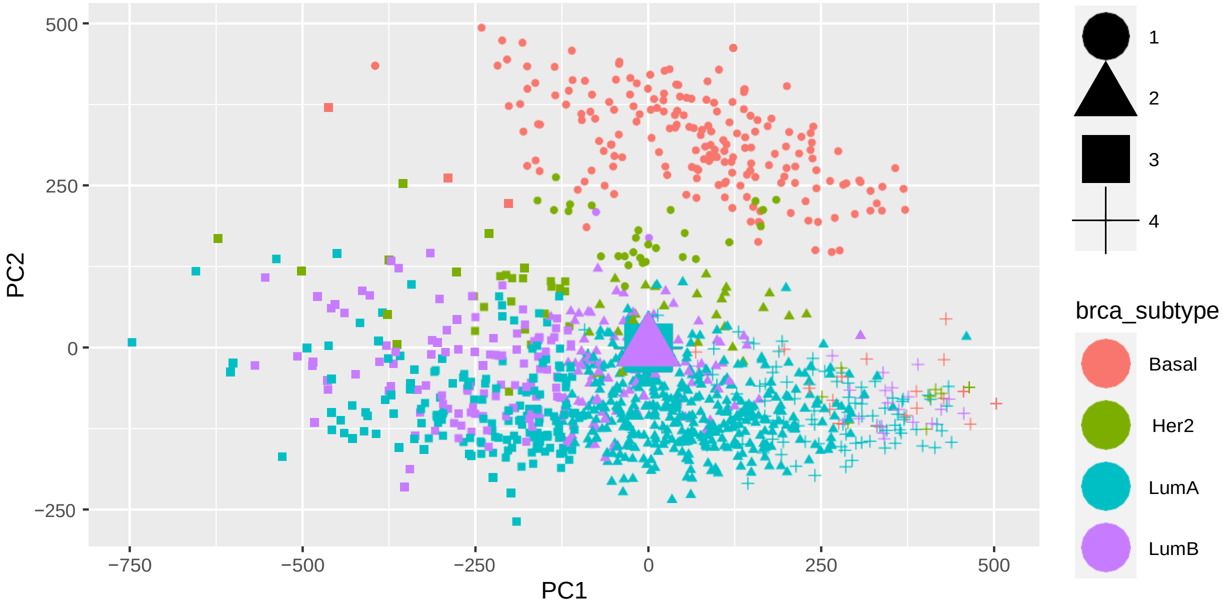
```
## [1]     1 19947
```

```
# remove genes with 0 counts
sel_cols <- which(colSums(abs(x_patient)) != 0)
x_patient_sub <- x_patient[, sel_cols, with=F]
dim(x_patient_sub)
```

```
## [1]     1 17193
```

```
# log
x_patient_sub <- log2(as.matrix(x_patient_sub+1e-10))

x_patient_sub <- scale(as.matrix(x_patient_sub),center=T,scale=F)

pc1_loadings<-as.data.frame(pca_unscaled$rotation[,1])
pc2_loadings<-as.data.frame(pca_unscaled$rotation[,2])

x_pc1<-sum(pc1_loadings*x_patient_sub)
x_pc2<-sum(pc2_loadings*x_patient_sub)
```

```
ggplot(df_unscaled, aes(x=PC1,y=PC2,col=brca_subtype,shape=cluster))+
          geom_point()+
  geom_point(x=x_pc1,y=x_pc2,size=10)
```



```
table(brca_subtype, kmeans_unscaled$cluster)
```

```
##
## brca_subtype   1   2   3   4
##        Basal 187   1   3  17
##         Her2  28  27  27   9
##         LumA   0 376 161  91
##         LumB   2  99 110  22
```

```
centroid1_dist<-sqrt((kmeans_unscaled$centers[1,1]-x_pc1)^2 + (kmeans_unscaled$centers[1,2]-x_pc2)^2)
centroid2_dist<-sqrt((kmeans_unscaled$centers[2,1]-x_pc1)^2 + (kmeans_unscaled$centers[2,2]-x_pc2)^2)
centroid3_dist<-sqrt((kmeans_unscaled$centers[3,1]-x_pc1)^2 + (kmeans_unscaled$centers[3,2]-x_pc2)^2)
centroid4_dist<-sqrt((kmeans_unscaled$centers[4,1]-x_pc1)^2 + (kmeans_unscaled$centers[4,2]-x_pc2)^2)

dists<-rbind(c("cluster1 dist","cluster2 dist","cluster3 dist","cluster4 dist"),
             c(centroid1_dist,centroid2_dist,centroid3_dist,centroid4_dist))

dists
```

```
##      [,1]               [,2]               [,3]
## [1,] "cluster1 dist"    "cluster2 dist"    "cluster3 dist"
## [2,] "304.376612038529" "89.4657826434346" "256.553961436401"
##      [,4]
## [1,] "cluster4 dist"
## [2,] "290.787367635422"
```

The new patient is closest to cluster 2, which is most likely to be the LumA subtype.

3 Case study 3: Auto data set

This question utilizes the Auto dataset from ISLR. The original dataset contains 408 observations about cars. It is similar to the CARS dataset that we use in our lectures. To get the data, first install the package ISLR. The Auto dataset should be loaded automatically. We'll use this dataset to practice the methods learn so far. Original data source is here: https://archive.ics.uci.edu/ml/datasets/auto+mpg

Get familiar with this dataset first. Tip: you can use the command ?ISLR::Auto to view a description of the dataset.

```
#load libraries
library(ggplot2)
library(GGally)

autoraw <- Auto[, c(1, 2, 3,4,5,6)]
library(ISLR)
?ISLR::Auto
dim(Auto)
```

```
## [1] 392   9
```

```
str(Auto)
```

```
## 'data.frame':     392 obs. of  9 variables:
##  $ mpg         : num  18 15 18 16 17 15 14 14 14 15 ...
##  $ cylinders   : num  8 8 8 8 8 8 8 8 8 8 ...
##  $ displacement: num  307 350 318 304 302 429 454 440 455 390 ...
##  $ horsepower  : num  130 165 150 150 140 198 220 215 225 190 ...
##  $ weight      : num  3504 3693 3436 3433 3449 ...
##  $ acceleration: num  12 11.5 11 12 10.5 10 9 8.5 10 8.5 ...
##  $ year        : num  70 70 70 70 70 70 70 70 70 70 ...
##  $ origin      : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ name        : Factor w/ 304 levels "amc ambassador brougham",..: 49 36 231 14 161 141 54 223 241
```

```
head(Auto)
```

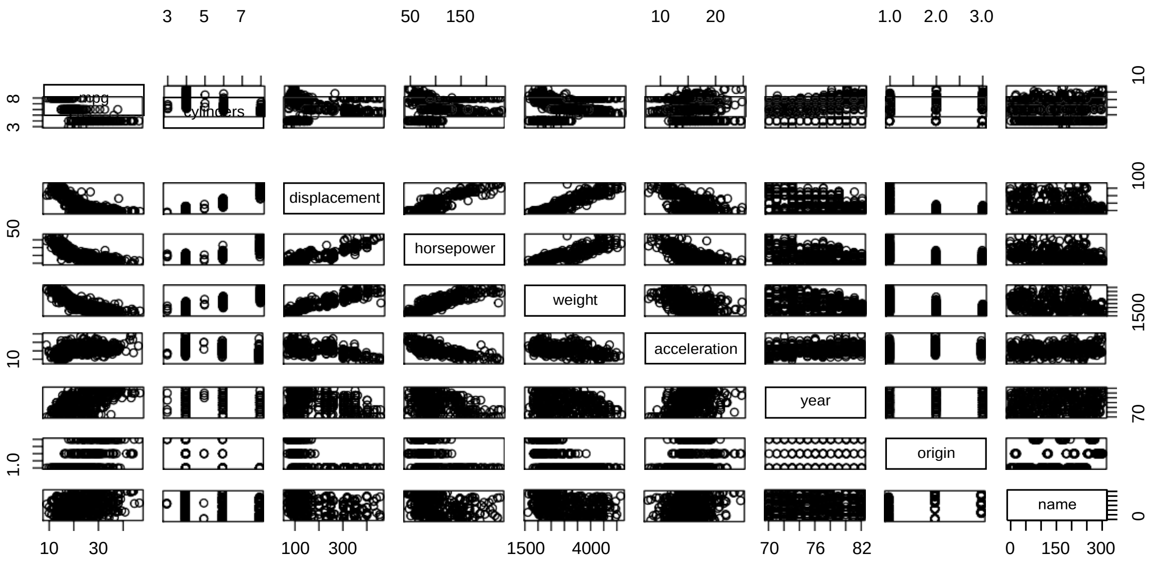
```
##   mpg cylinders displacement horsepower weight acceleration year origin
## 1  18         8          307        130   3504         12.0   70      1
## 2  15         8          350        165   3693         11.5   70      1
## 3  18         8          318        150   3436         11.0   70      1
## 4  16         8          304        150   3433         12.0   70      1
## 5  17         8          302        140   3449         10.5   70      1
## 6  15         8          429        198   4341         10.0   70      1
##                        name
## 1 chevrolet chevelle malibu
## 2         buick skylark 320
## 3        plymouth satellite
## 4             amc rebel sst
## 5               ford torino
## 6          ford galaxie 500
```

```
##       mpg          cylinders      displacement   horsepower        weight    
##  Min.   : 9.0   Min.   :3.00   Min.   : 68   Min.   : 46.0   Min.   :1613  
##  1st Qu.:17.0   1st Qu.:4.00   1st Qu.:105   1st Qu.: 75.0   1st Qu.:2225  
##  Median :22.8   Median :4.00   Median :151   Median : 93.5   Median :2804  
##  Mean   :23.4   Mean   :5.47   Mean   :194   Mean   :104.5   Mean   :2978  
##  3rd Qu.:29.0   3rd Qu.:8.00   3rd Qu.:276   3rd Qu.:126.0   3rd Qu.:3615  
##  Max.   :46.6   Max.   :8.00   Max.   :455   Max.   :230.0   Max.   :5140  
##                                                                            
##   acceleration       year        origin                      name    
##  Min.   : 8.0   Min.   :70   Min.   :1.00   amc matador       :  5  
##  1st Qu.:13.8   1st Qu.:73   1st Qu.:1.00   ford pinto        :  5  
##  Median :15.5   Median :76   Median :1.00   toyota corolla    :  5  
##  Mean   :15.5   Mean   :76   Mean   :1.58   amc gremlin       :  4  
##  3rd Qu.:17.0   3rd Qu.:79   3rd Qu.:2.00   amc hornet        :  4  
##  Max.   :24.8   Max.   :82   Max.   :3.00   chevrolet chevette:  4  
##                                             (Other)           :365  
```

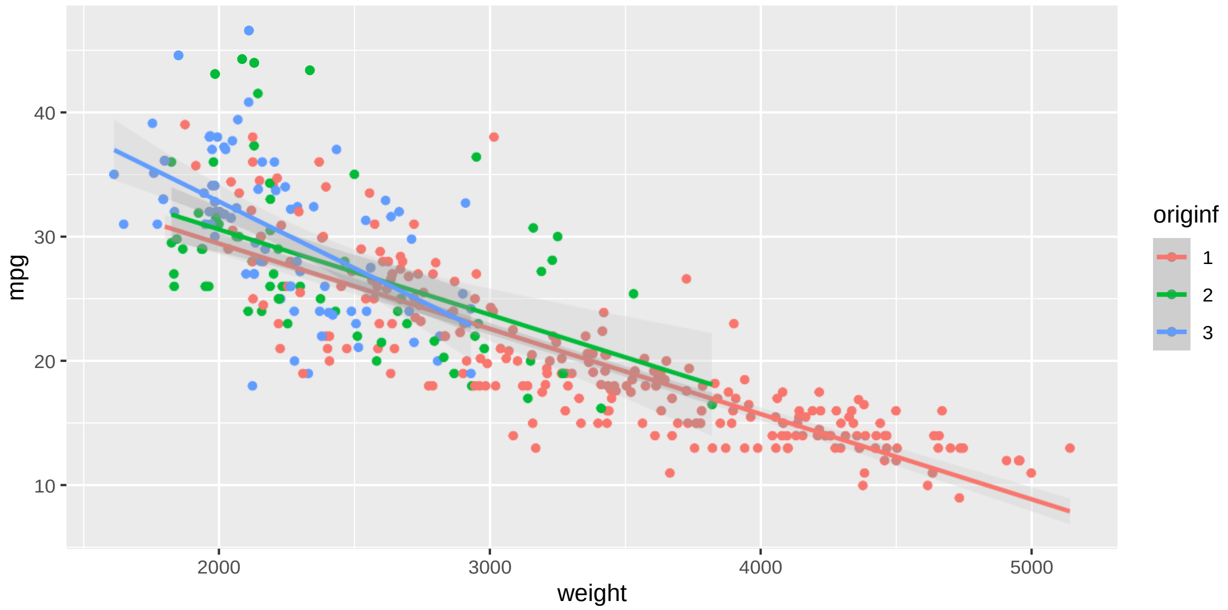
3.1 EDA

Explore the data, with particular focus on pairwise plots and summary statistics. Briefly summarize your findings and any peculiarities in the data.

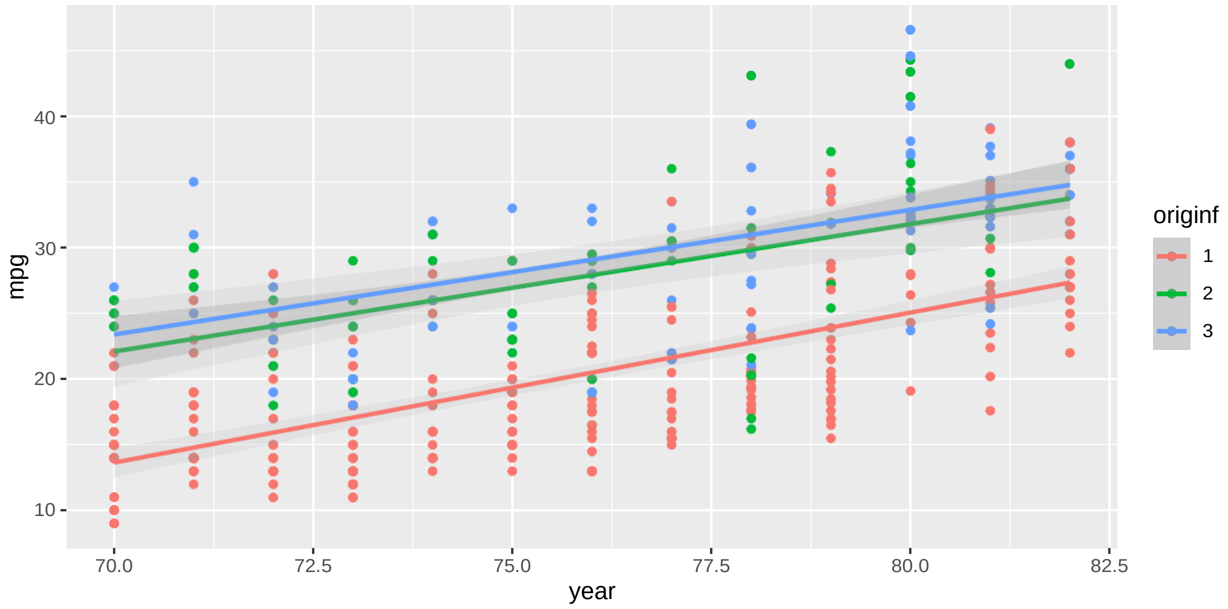
```
pairs(Auto)
```



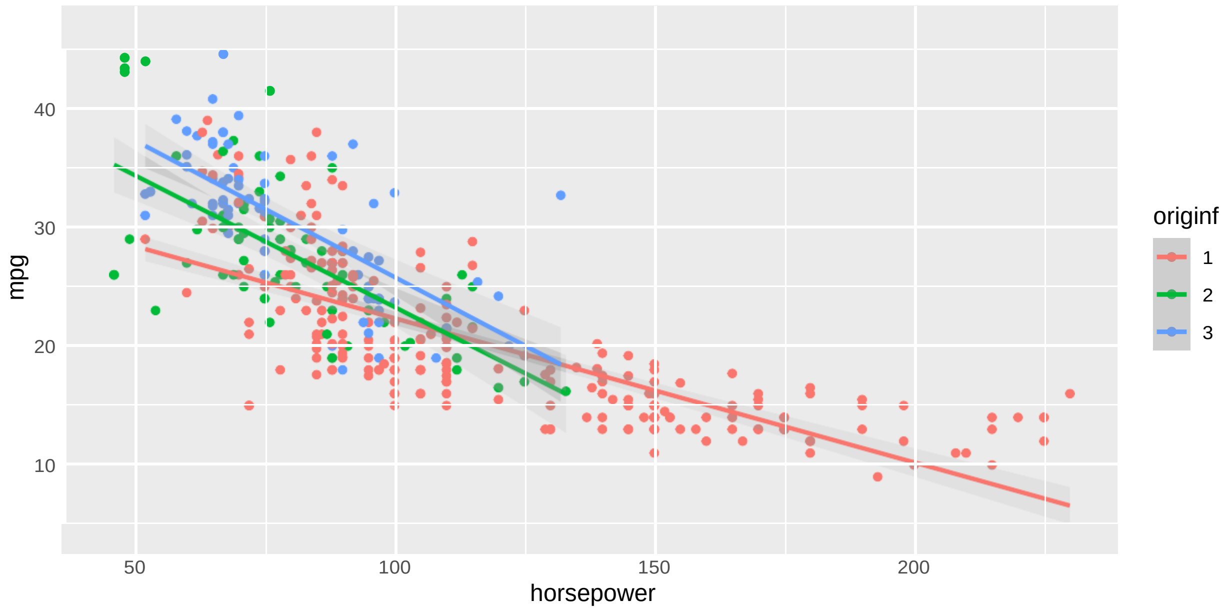
```
Auto$originf <- as.factor(Auto$origin)
Auto$yearf <- as.factor(Auto$year)
Auto$cylf <- as.factor(Auto$cylinders)


#Auto["originf"][Auto["originf"] == 1] <- "American"
#Auto["originf"][Auto["originf"] == 2] <- "European"
#Auto["originf"][Auto["originf"]  == 3] <- "Japanese"


Auto %>% ggplot() + aes(x = weight, y = mpg, col = originf) + geom_point() +
geom_smooth(method='lm', formula= y~x)
```



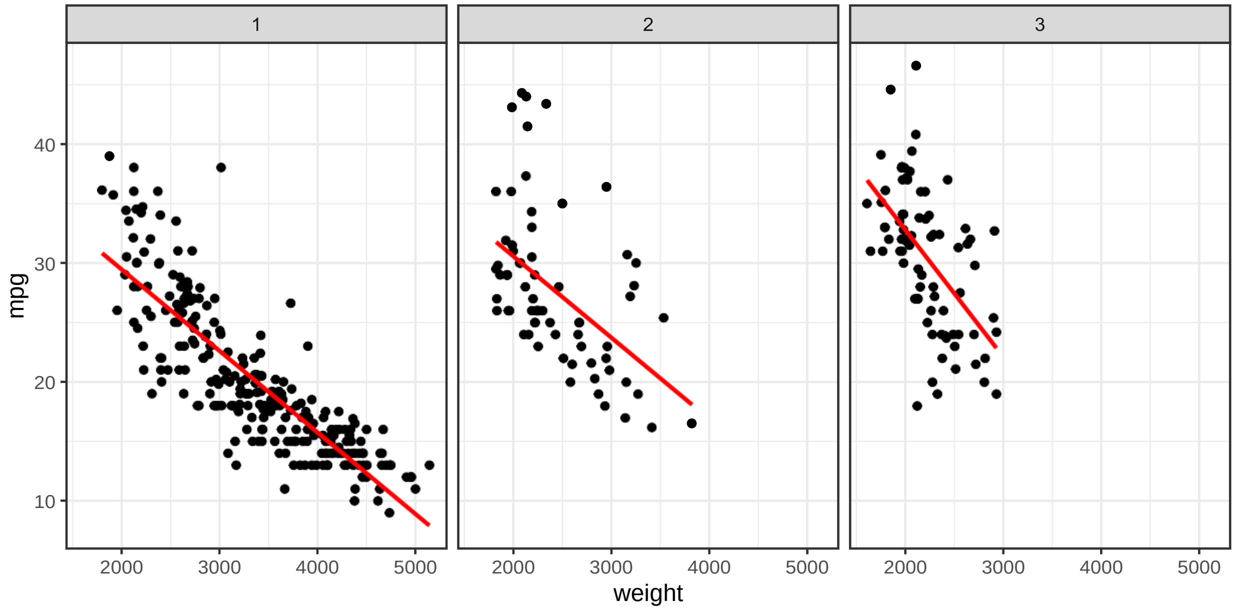
```
Auto %>% ggplot() + aes(x = year, y = mpg, col = originf) + geom_point() +
geom_smooth(method='lm', formula= y~x)
```



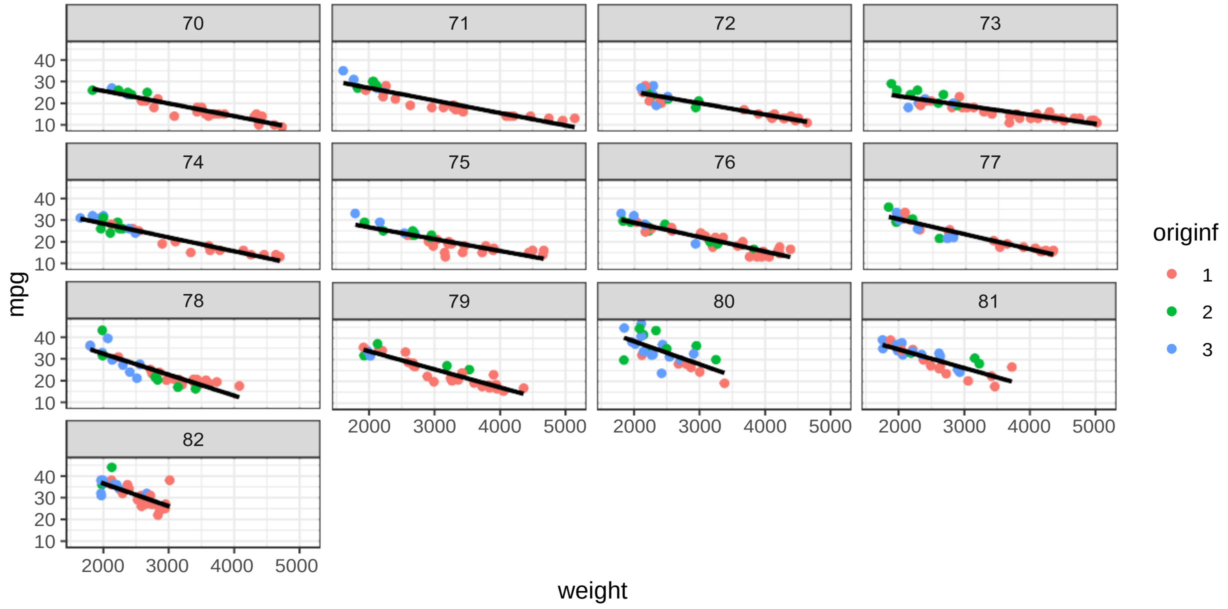
```
Auto %>% ggplot() + aes(x = horsepower, y = mpg, col = originf) + geom_point() +
geom_smooth(method='lm', formula= y~x)
```



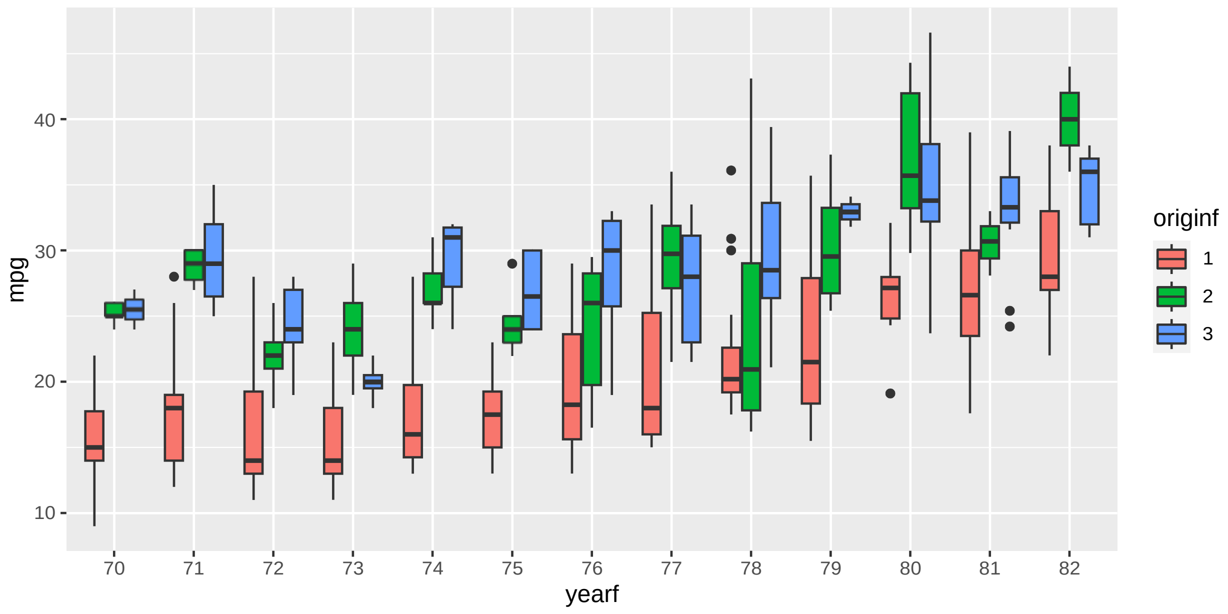
```
Auto %>%
ggplot(aes(x=weight, y=mpg, group = originf)) +
geom_point()+
geom_smooth(method="lm", formula=y~x, se=F,color = "red")+
facet_wrap(~origin) +
theme_bw()
```



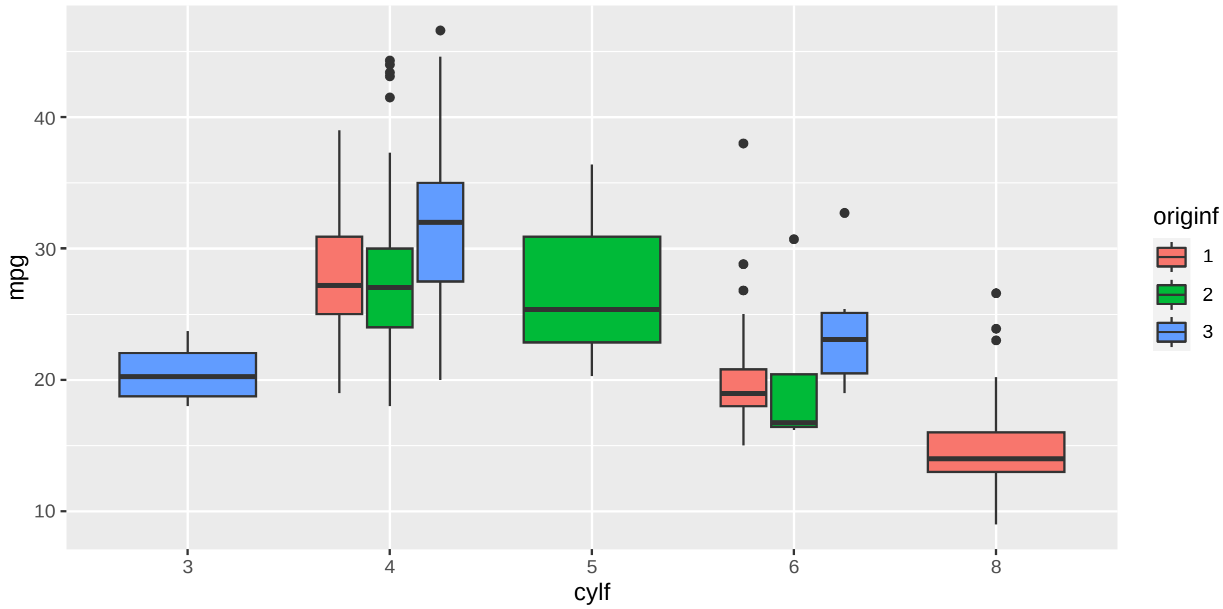
```
Auto %>%
ggplot(aes(x=weight, y=mpg, group = year, col = originf)) +
geom_point()+
geom_smooth(method="lm", formula=y~x, se=F,color = "black")+
facet_wrap(~year) +
theme_bw()
```



```
Auto %>%
  ggplot(aes(x = yearf, y = mpg, fill  = originf)) +
  geom_boxplot()
```



```
Auto %>%
  ggplot(aes(x = cylf, y = mpg, fill = originf)) +
  geom_boxplot()
```



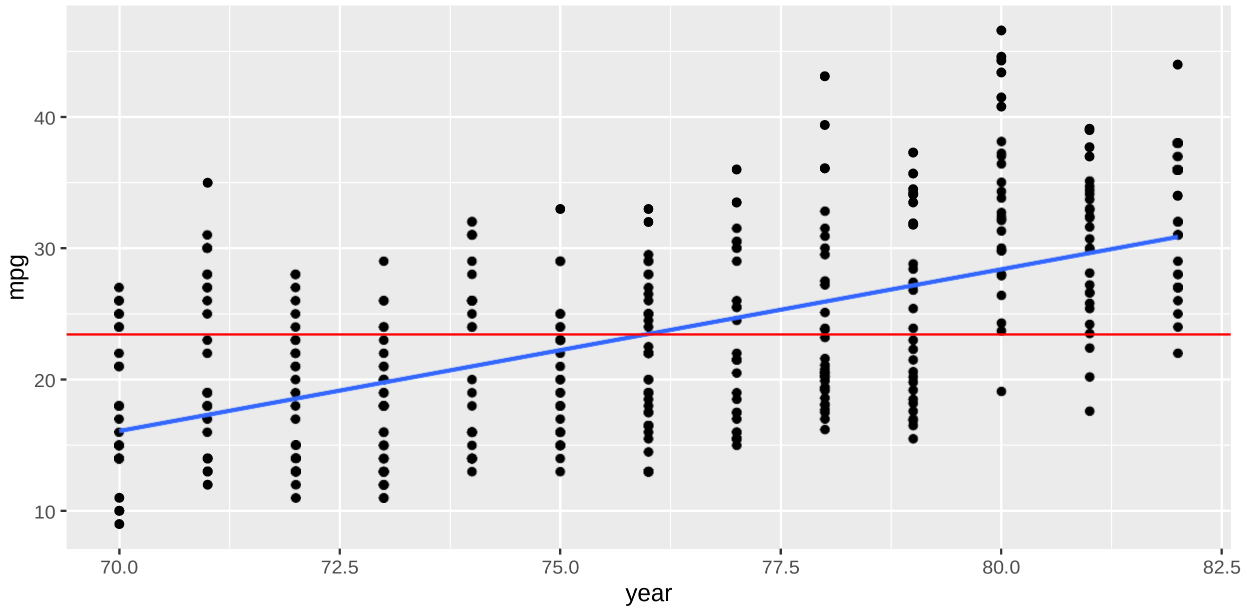
From the above plots, we can see numerous relationships; however later inspection shows variables to be correlated. It appears mpg increases in year, decreases in weight, displacement, cylinders, and horsepower.

Differences between the three origins are present, even when isolating other influences. Therefore, when building models, we will keep variables and work to eliminate those that are highly correlated and uninformative.

3.2 What effect does time have on MPG?

a) Start with a simple regression of mpg vs. year and report R's summary output. Is year a significant variable at the .05 level? State what effect year has on mpg, if any, according to this model.

```
fit1 <- lm(mpg ~ year, data = Auto)
ggplot(Auto, aes(x = year , y = mpg)) +
  geom_point() +
  geom_smooth(method="lm",se=F) +
geom_hline(aes(yintercept = mean(mpg)), color = "red")
```



```
summary(fit1)
```

```
##
## Call:
## lm(formula = mpg ~ year, data = Auto)
##
## Residuals:
##      Min      1Q  Median      3Q     Max
## -12.021  -5.441  -0.441   4.974  18.209
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -70.0117     6.6452   -10.5   <2e-16 ***
## year          1.2300     0.0874    14.1   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.36 on 390 degrees of freedom
## Multiple R-squared:  0.337,  Adjusted R-squared:  0.335
## F-statistic:   198 on 1 and 390 DF,  p-value: <2e-16
```

From fit1, the year variable has a p-value of near 0, so it is significant at the 0.05 threshold. This aligns with comparison of the average and fitted model because one shows no trend and the other shows a clear upward trend.

From the above analysis, the beta value for year was found to be 1.23, so the fit says that for a increase in year, mpg (on average), increases by 1.23mpg.

b) Add horsepower on top of the variable year to your linear model. Is year still a significant variable at the .05 level? Give a precise interpretation of the year's effect found here.

```
fit2 <- lm(mpg ~ year + horsepower, data = Auto)
summary(fit2)
```

```
##
## Call:
## lm(formula = mpg ~ year + horsepower, data = Auto)
##
## Residuals:
##       Min       1Q  Median       3Q   Max##
-12.077   -3.078   -0.431    2.588   15.315 ##
## Coefficients:
##                 Estimate Std. Error t value  Pr(>|t|)
## (Intercept) -12.73917     5.34903   -2.38     0.018 *
## year          0.65727     0.06626    9.92    <2e-16 ***
## horsepower   -0.13165     0.00634  -20.76    <2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 4.39 on 389 degrees of freedom
## Multiple R-squared: 0.685,  Adjusted R-squared:  0.684 ##
F-statistic: 424 on 2 and 389 DF,  p-value: <2e-16
```

Year remained statistically significant at the 0.05 level, but its coefficient decreased by nearly 50% after the introduction of a horsepower variable.

Now, the interpretation for the year beta coefficient is as follows: with mpg held constant, an increase in year, on average, increases mpg by 0.65.

c) The two 95% CI’s for the coefficient of year differ among (i) and (ii). How would you explain the difference to a non-statistician?

The 95% CI for i is beta_year = 1.2300 +/- 2(0.0874) The 95% CI for ii is beta_year = 0.65727 +/- 2(0.06626)

This difference arises because of the introduction of another variable, which captures information from the horsepower data. Now, since we now predict based on horespower and year, we dont have to just guess basses on year. With more information, we can tighten our confidence interval on the year coeeficient. Without the additional information from horsepower, we had to have a wider confidence interval because we had less information (variance) to base our model on

d) Create a model with interaction by fitting lm(mpg ~ year * horsepower). Is the interaction effect significant at .05 level? Explain the year effect (if any).

```
fit3 <- lm(mpg ~ year * horsepower, data = Auto)
summary(fit3)
```

```
##
## Call:
## lm(formula = mpg ~ year * horsepower, data = Auto)
##
## Residuals:
```

```
##       Min        1Q   Median       3Q       Max
## -12.349   -2.451   -0.456    2.406   14.444
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -1.27e+02     1.21e+01    -10.45     <2e-16 ***
## year                   2.19e+00  1.61e-01  13.59  <2e-16  ***
## horsepower           1.05e+00    1.15e-01      9.06      <2e-16  ***##
year:horsepower -1.60e-02     1.56e-03   -10.22     <2e-16 *** ## -
--
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.9 on 388 degrees of freedom
## Multiple R-squared: 0.752,  Adjusted R-squared:  0.75 ##
F-statistic: 393 on 3 and 388 DF, p-value: <2e-16
```

The interaction term was found to be significant at the 0.05 level. The effect of year, now with the interaction term, is two-fold. First, holding horsepower constant, increasing year increases mpg by 2.19mpg. The interaction term between year and horsepower comes into play, mpg goes down by 0.016 (horsepower * change in year)

3.3 Categorical predictors

Remember that the same variable can play different roles! Take a quick look at the variable cylinders, and try to use this variable in the following analyses wisely. We all agree that a larger number of cylinders will lower mpg. However, we can interpret cylinders as either a continuous (numeric) variable or a categorical variable.

a) Fit a model that treats cylinders as a continuous/numeric variable. Is cylinders significant at the 0.01 level? What effect does cylinders play in this model?

```
fit4 <- lm(mpg ~ cylinders, data = Auto)
summary(fit4)
```

```
##
## Call:
## lm(formula = mpg ~ cylinders, data = Auto)
##
## Residuals:
##       Min        1Q   Median       3Q       Max
## -14.241   -3.183   -0.633    2.549   17.917
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      42.916        0.835      51.4     <2e-16 ***
## cylinders        -3.558        0.146     -24.4     <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.91 on 390 degrees of freedom
## Multiple R-squared:   0.605,    Adjusted R-squared:   0.604
## F-statistic:      597 on 1 and 390 DF,   p-value: <2e-16
```

Treating cylinders as a continuous variable results in cylinders being significant at the 0.01 level. For each incremental cylinder, mpg decreases by 3.558, with a theoretical 0-cylinder car having 42.916 mpg. This is not easily interpreted, but does show a negative relationship.

b) Fit a model that treats cylinders as a categorical/factor. Is cylinders significant at the .01 level? What is the effect of cylinders in this model? Describe the cylinders effect over mpg.

```
fit5 <- lm(mpg ~ cylf, data = Auto)
summary(fit5)
```

```
##
## Call:
## lm(formula = mpg ~ cylf, data = Auto)
##
## Residuals:
##       Min        1Q  Median       3Q    Max##
-11.284    -2.904    -0.963      2.344    18.027 ##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     20.550        2.349      8.75   < 2e-16 ***
## cylf4             8.734        2.373      3.68   0.00027 ***
## cylf5             6.817        3.589      1.90   0.05825 .
## cylf6            -0.577        2.405     -0.24   0.81071
## cylf8            -5.587        2.395     -2.33   0.02015 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.7 on 387 degrees of freedom
## Multiple R-squared:  0.641,   Adjusted R-squared:  0.638
##  F-statistic:    173 on 4 and 387 DF,   p-value: <2e-16
```

Now, only cylinder4 is significant, because the other categories do not reach significance beyond the 0.01 level. To interpret this, the average MPG for a three cylinder car is the intercept, with cars in each other category differing from the intercept by the coefficient of their respective category. The only coefficient that is statistically significant is for cylf4.

c) What are the fundamental differences between treating cylinders as a continuous and categorical variable in your models?

When treating cylinders as a continuous variable, the model estimates a coefficient for each incremental change in cylinder count– meaning that the model fits a straight line.

There are two problems with this. First there are no 1,2 or 7 cylinder cars. This means that cylinders is not really a continuous variable. The second, more important issue, is that the 4th incremental cylinder and 7th/8th incremental cylinders likely have different effects.

d) Can you test the null hypothesis: fit0: mpg is linear in cylinders vs. fit1: mpg relates to cylinders as a categorical variable at .01 level?

```
anova(fit4, fit5)
```

```
## Analysis of Variance Table
##
## Model 1: mpg ~ cylinders
## Model 2: mpg ~ cylf
##   Res.Df  RSS Df Sum of Sq    F  Pr(>F)
## 1    390 9416
## 2    387 8544  3       871 13.2 3.4e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Since the p-value of the anova test was statistically significant (near 0), we succeed in rejecting the null hypothesis that mpg is linear in cylinders and show that it is categorical.

p-val < 0.01

3.4 Results

Final modeling question: we want to explore the effects of each feature as best as possible. You may explore interactions, feature transformations, higher order terms, or other strategies within reason. The model(s) should be as parsimonious (simple) as possible unless the gain in accuracy is significant from your point of view.

```
res <- cor(autoraw)
round(res, 2)
```

```
##                mpg cylinders displacement horsepower weight acceleration
## mpg           1.00     -0.78        -0.81      -0.78  -0.83         0.42
## cylinders    -0.78      1.00         0.95       0.84   0.90        -0.50
## displacement -0.81      0.95         1.00       0.90   0.93        -0.54
## horsepower   -0.78      0.84         0.90       1.00   0.86        -0.69
## weight       -0.83      0.90         0.93       0.86   1.00        -0.42
## acceleration  0.42     -0.50        -0.54      -0.69  -0.42         1.00
```

```
knitr::include_graphics("occam.png")
```

From the correlation matrix, we see that displacement and cylinders are strongly correlated (0.95) so the two probably don't provide much information on-top of each other. Horsepower also appears highly correlated (0.9) with displacement, as is weight (0.93).

Building a model using the philosophy of "Occam's razor," "plurality should not be posited without necessity," we will compare a model using all inputs and one that removes both insignificant ones and variables that are correlated with others.

Furthermore, to preserve interpret ability and avoid possible over-fitting, we will avoid higher-order terms and interaction terms.

```
model1 <- lm(mpg ~ cylf + displacement + horsepower + weight +
                acceleration + year + originf, data = Auto)
summary(model1) ### key output
```

```
##
## Call:
## lm(formula = mpg ~ cylf + displacement + horsepower + weight +
##      acceleration + year + originf, data = Auto)
##
## Residuals:
##    Min    1Q Median    3Q    Max ##
-8.680 -1.937 -0.068  1.671 12.776 ##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -2.21e+01   4.54e+00   -4.86  1.7e-06 ***
## cylf4         6.72e+00   1.65e+00    4.06  5.9e-05 ***
## cylf5         7.08e+00   2.52e+00    2.81   0.0052 **
## cylf6         3.35e+00   1.82e+00    1.84   0.0670  .
## cylf8         5.10e+00   2.11e+00    2.42   0.0161  *
## displacement  1.87e-02   7.22e-03    2.59   0.0100  **
## horsepower    -3.49e-02    1.32e-02    -2.64    0.0087 **
## weight        -5.78e-03    6.31e-04    -9.15   < 2e-16  ***
## acceleration   2.60e-02    9.30e-02     0.28    0.7802
## year           7.37e-01    4.89e-02    15.06   <  2e-16 ***
## originf2       1.76e+00    5.51e-01     3.20    0.0015 **
## originf3       2.62e+00    5.27e-01     4.96   1.0e-06 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 3.1 on 380 degrees of freedom
## Multiple R-squared:  0.847,    Adjusted R-squared:  0.842
##  F-statistic:   191 on 11 and 380 DF,  p-value: <2e-16
```

```
model2 <- lm(mpg ~ horsepower + weight +
             acceleration + year + originf, data = Auto)
summary(model2) ### key output
```

```
##
## Call:
## lm(formula = mpg ~ horsepower + weight + acceleration + year +
##      originf, data = Auto)
##
## Residuals:
##     Min     1Q Median     3Q     Max ##
-9.492  -2.090  -0.008   1.832  13.450 ##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -1.80e+01    4.63e+00    -3.89   0.00012 ***
## horsepower    -4.75e-03    1.32e-02    -0.36   0.71869
## weight        -5.66e-03    5.03e-04   -11.25   <  2e-16 ***
## acceleration   3.92e-02    9.77e-02     0.40   0.68857
## year           7.55e-01    5.17e-02    14.62   <  2e-16 ***
## originf2       1.94e+00    5.21e-01     3.72   0.00023 ***
## originf3       2.27e+00    5.26e-01     4.31      2e-05 ***
## ---
## Signif. codes: 0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 3.34 on 385 degrees of freedom
## Multiple R-squared: 0.819,  Adjusted R-squared:  0.817 ##
F-statistic: 291 on 6 and 385 DF, p-value: <2e-16
```

```
model3 <- lm(mpg ~ weight + year + originf, data = Auto)
summary(model3) ### key output
```

```
##
## Call:
## lm(formula = mpg ~ weight + year + originf, data = Auto)
##
## Residuals:
##     Min     1Q Median     3Q     Max ##
-9.603  -2.113  -0.021   1.762  13.526 ##
```

```
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -18.30694    4.01724   -4.56  7.0e-06 ***
## weight       -0.00589    0.00026  -22.65  < 2e-16 ***
## year          0.76985    0.04867   15.82  < 2e-16 ***
## originf2      1.97631    0.51797    3.82  0.00016 ***
## originf3      2.21453    0.51882    4.27  2.5e-05 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.34 on 387 degrees of freedom
## Multiple R-squared: 0.819, Adjusted R-squared: 0.817 ##
F-statistic: 438 on 4 and 387 DF, p-value: <2e-16
```

```
model4 <- lm(mpg ~ weight * year + originf, data = Auto)
```

```
AIC(model1, model2, model3, model4)
```

```
##        df  AIC
## model1 13 2013
## model2  8 2067
## model3  6 2064
## model4  7 2010
```

```
BIC(model1, model2, model3, model4)
```

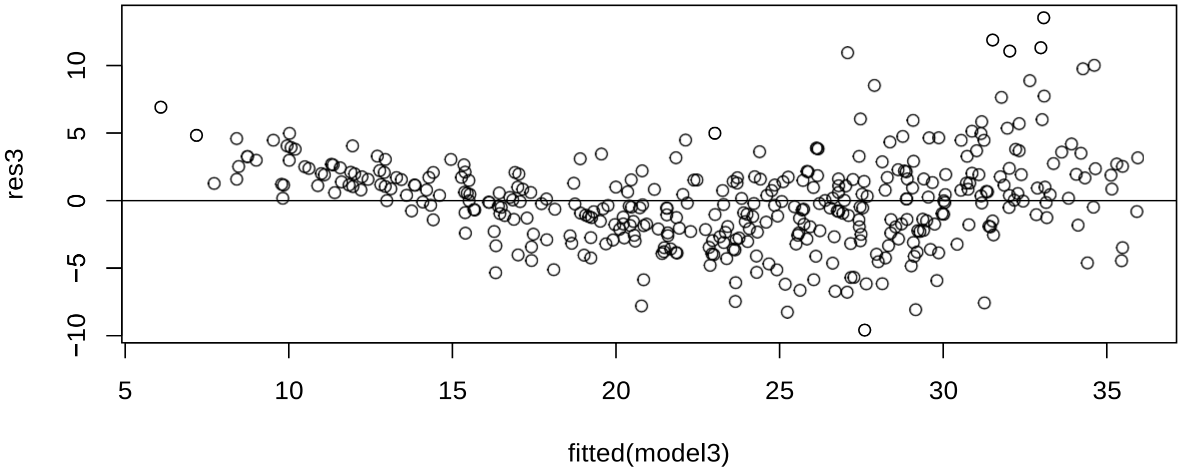
```
##        df  BIC
## model1 13 2064
## model2  8 2099
## model3  6 2088
## model4  7 2038
```

a) Describe the final model. Include diagnostic plots with particular focus on the model residuals and diagnoses.

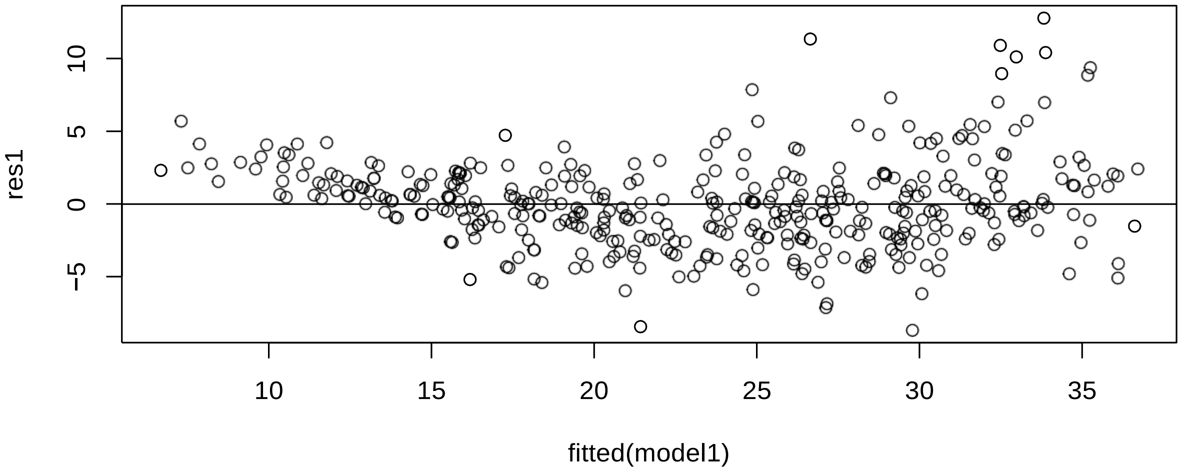
For parsimony, we select model3 with just three variables: weight, year, and origin. From further inspection of the residual plots, there appears to be limited gain from the complexity of model1 relative to model3.

A model with an interaction term for year * weight, the story being fuel efficiencies could be non-linear in more advanced cars, especially for heavier ones, but this model, from inspection of BIC, AIC, and the plots below has limited benefit.

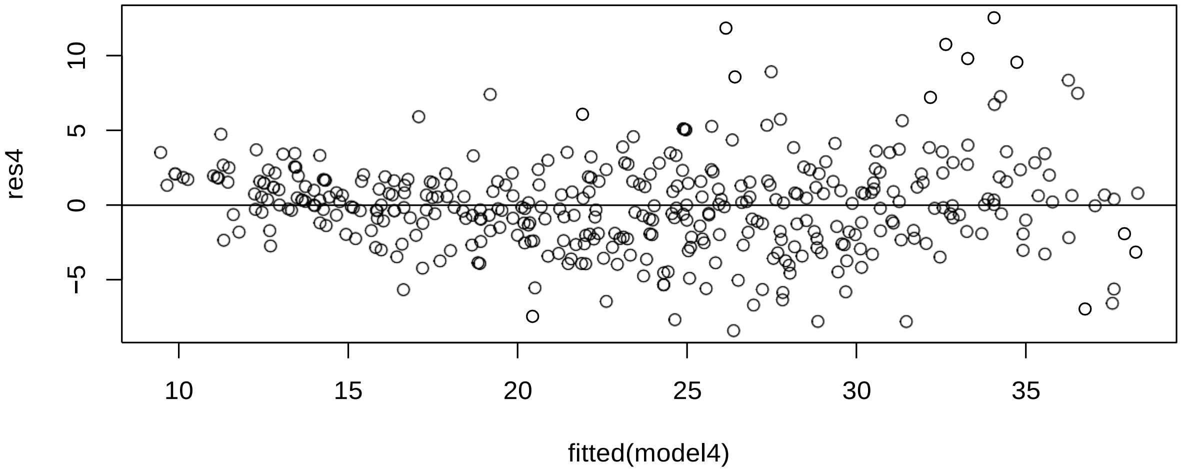
```
res3 <- resid(model3)
plot(fitted(model3), res3)
abline(0,0)
```



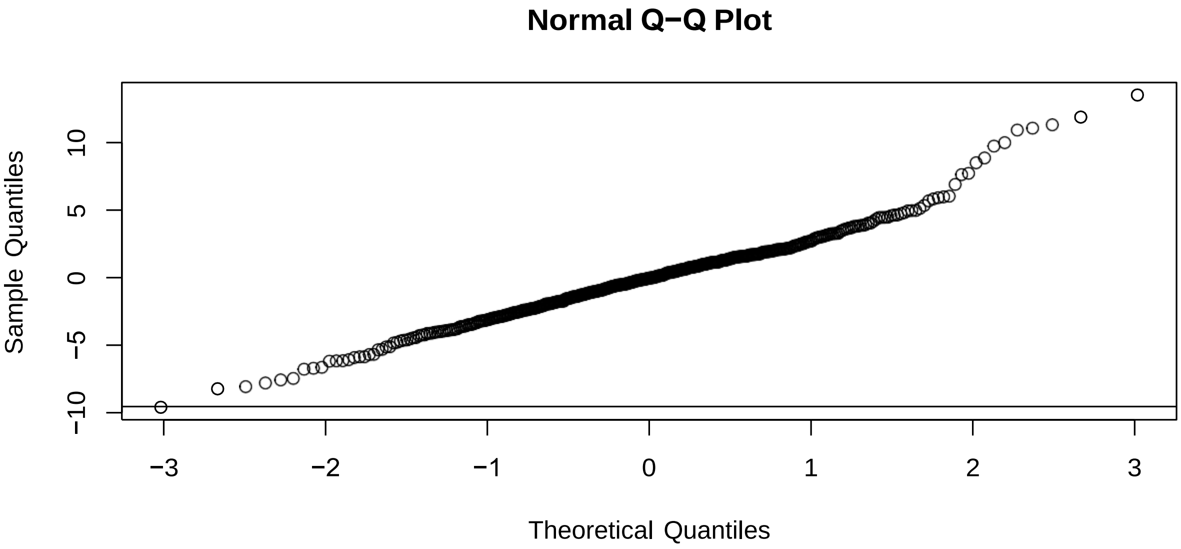
```
res1 <- resid(model1)
plot(fitted(model1), res1)
abline(0,0)
```



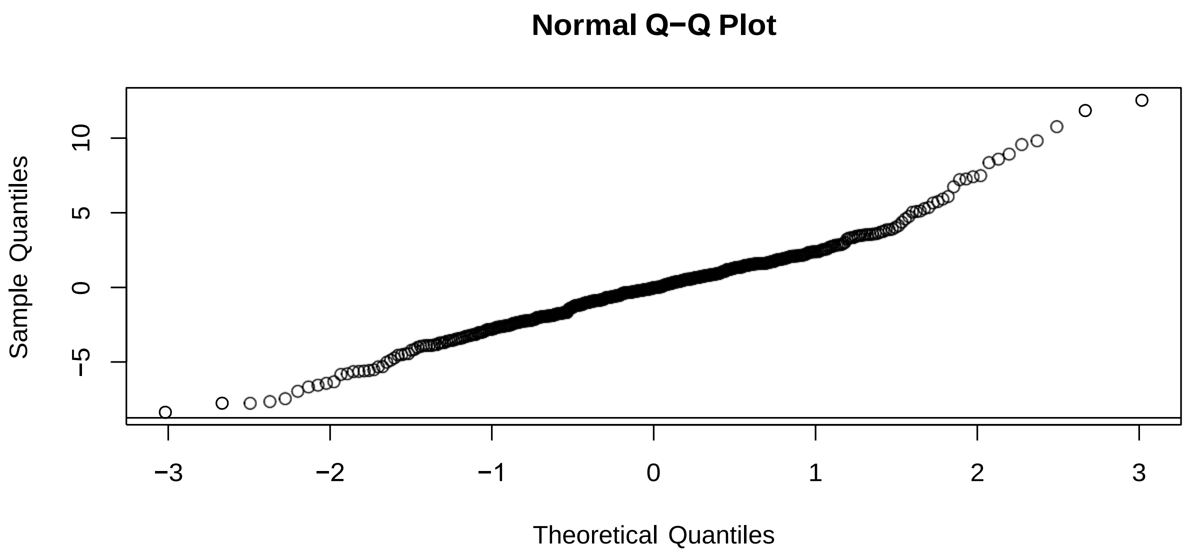
```
res4 <- resid(model4)
plot(fitted(model4), res4)
abline(0,0)
```



```
qqnorm(res3)
qqline(res3)
```



```
qqnorm(res4)
qqline(res4)
```



From the above plots, there appear to be non-normal residual distributions on the upper tail, the upper tail of the q-q plot is fat. From inspection of the plot of residuals for model3, there appears to be more variance at higher fitted values.

There appears to be relative homoskedacicity in the middle of fitted values, but with heteroskedacity at relative outlier observations at both tails.

model4, with the interaction term,

b) Summarize the effects found.

With the selected model, model3, the effects found are as follows:

mpg is increasing with repect to year mpg decreases with additional weight

Cars made in Europe get a slight boost relative to American cars Cars made in Japan get a slight boost relative to American cars

c) Predict the mpg of the following car: A red car built in the US in 1983 that is 180 inches long, has eight cylinders, displaces 350 cu. inches, weighs 4000 pounds, and has a horsepower of 260. Also givea 95% CI for your prediction.

```
newcar <-  Auto[1, ]  # Create a new row with same structure as in data1
newcar[1] <- "NA" # Assign features for the new car
newcar$origin <- 1
newcar$cylinders <- 8
newcar$displacement <- 350
newcar$horsepower <- 260
newcar$weight <- 4000
newcar$year <- 83

predict(model3,newcar,interval="prediction",se.fit=T)
```

```
## $fit
##   fit  lwr  upr ##
1  22  15.4  28.7##
##   $se.fit
## [1] 0.484
##
## $df
## [1] 387
##
## $residual.scale
## [1] 3.34
```

```
predict(model3,newcar,interval="confidence",se.fit=T)
```

```
## $fit
##   fit  lwr upr ##
1  22 21.1   23 ##
##   $se.fit
## [1] 0.484
##
## $df
## [1] 387
##
## $residual.scale
## [1] 3.34
```

From the above results, the predicted fuel efficiency is 22 mpg with a 95% prediction interval between 15.4 and 28.7.

The confidence interval, 95% is between 21.1 and 23mpg.

4 Simple Regression through simulations

4.1 Linear model through simulations

This exercise is designed to help you understand the linear model using simulations. In this exercise, we will generate (x_i, y_i) pairs so that all linear model assumptions are met.

Presume that **x** and **y** are linearly related with a normal error $\boldsymbol{\varepsilon}$, such that $\mathbf{y} = 1 + 1.2\mathbf{x} + \boldsymbol{\varepsilon}$. The standard deviation of the error ε_i is $\sigma = 2$.

We can create a sample input vector ($n = 40$) for **x** with the following code:

```
# Generates a vector of size 40 with equally spaced values between 0 and 1, inclusive
x <- seq(0, 1, length = 40)
```

4.1.1 Generate data

Create a corresponding output vector for **y** according to the equation given above. Use set.seed(1). Then, create a scatterplot with (x_i, y_i) pairs. Base R plotting is acceptable, but if you can, please attempt to use ggplot2 to create the plot. Make sure to have clear labels and sensible titles on your plots.

4.1.2 Understand the model

i. Find the LS estimates of $\boldsymbol{\beta}_0$ and $\boldsymbol{\beta}_1$, using the lm() function. What are the true values of $\boldsymbol{\beta}_0$ and $\boldsymbol{\beta}_1$? Do the estimates look to be good?

ii. What is your RSE for this linear model fit? Is it close to $\sigma = 2$?

iii. What is the 95% confidence interval for $\boldsymbol{\beta}_1$? Does this confidence interval capture the true $\boldsymbol{\beta}_1$?

iv. Overlay the LS estimates and the true lines of the mean function onto a copy of the scatterplot you made above.

4.1.3 diagnoses

i. Provide residual plot where fitted **y**-values are on the x-axis and residuals are on the y-axis.

ii. Provide a normal QQ plot of the residuals.

iii. Comment on how well the model assumptions are met for the sample you used.

4.2 Understand sampling distribution and confidence intervals

This part aims to help you understand the notion of sampling statistics and confidence intervals. Let's concentrate on estimating the slope only.

Generate 100 samples of size n = 40, and estimate the slope coefficient from each sample. We include some sample code below, which should guide you in setting up the simulation. Note: this code is easier to follow but suboptimal; see the appendix for a more optimal R-like way to run this simulation.

```
# Inializing variables. Note b_1, upper_ci, lower_ci are vectors
x <- seq(0, 1, length = 40)
n_sim <- 100                # number of simulations
b1 <- 0                     # n_sim many LS estimates of beta_1 (=1.2). Initialize to 0 for now
upper_ci <- 0               # upper bound for beta_1. Initialize to 0 for now.
lower_ci <- 0               # lower bound for beta_1. Initialize to 0 for now.
t_star <- qt(0.975, 38)     # Food for thought: why 38 instead of 40? What is t_star?

# Perform the simulation
for (i in 1:n_sim){
  y <- 1 + 1.2 * x + rnorm(40, sd = 2)
  lse <- lm(y ~ x)
  lse_output <- summary(lse)$coefficients
  se <- lse_output[2, 2]
  b1[i] <- lse_output[2, 1]
  upper_ci[i] <- b1[i] + t_star * se
  lower_ci[i] <- b1[i] - t_star * se
}
results <- as.data.frame(cbind(se, b1, upper_ci, lower_ci))

# remove unecessary variables from our workspace
rm(se, b1, upper_ci, lower_ci, x, n_sim, b1, t_star, lse, lse_out)
```

i. Summarize the LS estimates of $\boldsymbol{\beta}_1$ (stored in results$b1). Does the sampling distribution agree with theory?

ii. How many of your 95% confidence intervals capture the true $\boldsymbol{\beta}_1$? Display your confidence intervals graphically.