Modern Data Mining, HW 2

Group Jenea Adams   Group Annan Timon

Due: 11:59 PM, Sunday, 02/12

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 a 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 on the other hand.

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 (missing), 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?

```
# temp <- read.csv('data/NLSY79.csv', header = T, stringsAsFactors = F)
# # missing values? real variables vs. factors? are varable values reasonable?
# str(temp)
# summary(temp)
# levels(as.factor(temp$Job05))
# table(as.factor(temp$Job05))
```

1.2 Self esteem evaluation

Let concentrate on Esteem scores evaluated in 87.

0. First do a quick summary over all the `Esteem` variables. Pay attention to missing values, any peculiar numbers etc. How do you fix problems discovered if there is any? Briefly describe what you have done for the data preparation.

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.)

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

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

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

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

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

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

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

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

   f) Also plot CPVE (Cumulative Proportion of Variance Explained). What proportion of the variance in the 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 two PC's to display the data. Give an interpretation of PC1 and PC2 from the plot. (try `ggbiplot` if you could, much prettier!)

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.
   b) Can you summarize common features within each cluster?
   c) Can you visualize the clusters with somewhat clear boundaries? You may try different pairs of variables and different PC pairs of the esteem scores.

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:

   - EDA the data set first.
   - Personal information: gender, education (05), log(income) in 87, job type in 87. Weight05 (lb) and HeightFeet05 together with Heightinch05. One way to summarize one's weight and height is via Body Mass Index which is defined as the body mass divided by the square of the body height, and is universally expressed in units of kg/m². Note, you need to create BMI first. Then may include it as one possible predictor.
   - Household environment: Imagazine, Inewspaper, Ilibrary, MotherEd, FatherEd, FamilyIncome78. Do set indicators `Imagazine`, `Inewspaper` and `Ilibrary` as factors.
   - You may use PC1 of ASVAB as level of intelligence

   b) Run a few regression models between PC1 of all the esteem scores and suitable variables listed in a). Find a final best model with your own criterion.

   - How did you land this model? Run a model diagnosis to see if the linear model assumptions are reasonably met.
   - Write a summary of your findings. In particular, explain what and how the variables in the model affect one's self-esteem.

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()`.

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

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

1. Summary and transformation

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

```
kable(data.frame(t(summary((brca_subtype)))),
      caption = "Patients in each sub-type")
```

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

```
num_gene <- ncol(brca)

# randomly select 10 gene
set.seed(10)
sample_idx <- sample(num_gene, 5)

# plot count number histogram for each gene
brca %>%
  select(c(all_of(sample_idx))) %>%
  cbind(.,brca_subtype) %>% # select column by index
  pivot_longer(-brca_subtype) %>%      # for facet(0)
  ggplot(aes(x = value, y = ..density..)) +
  geom_histogram(aes(fill = brca_subtype)) +
  facet_wrap(~name, scales = "free") +
  theme(legend.position = "none")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```
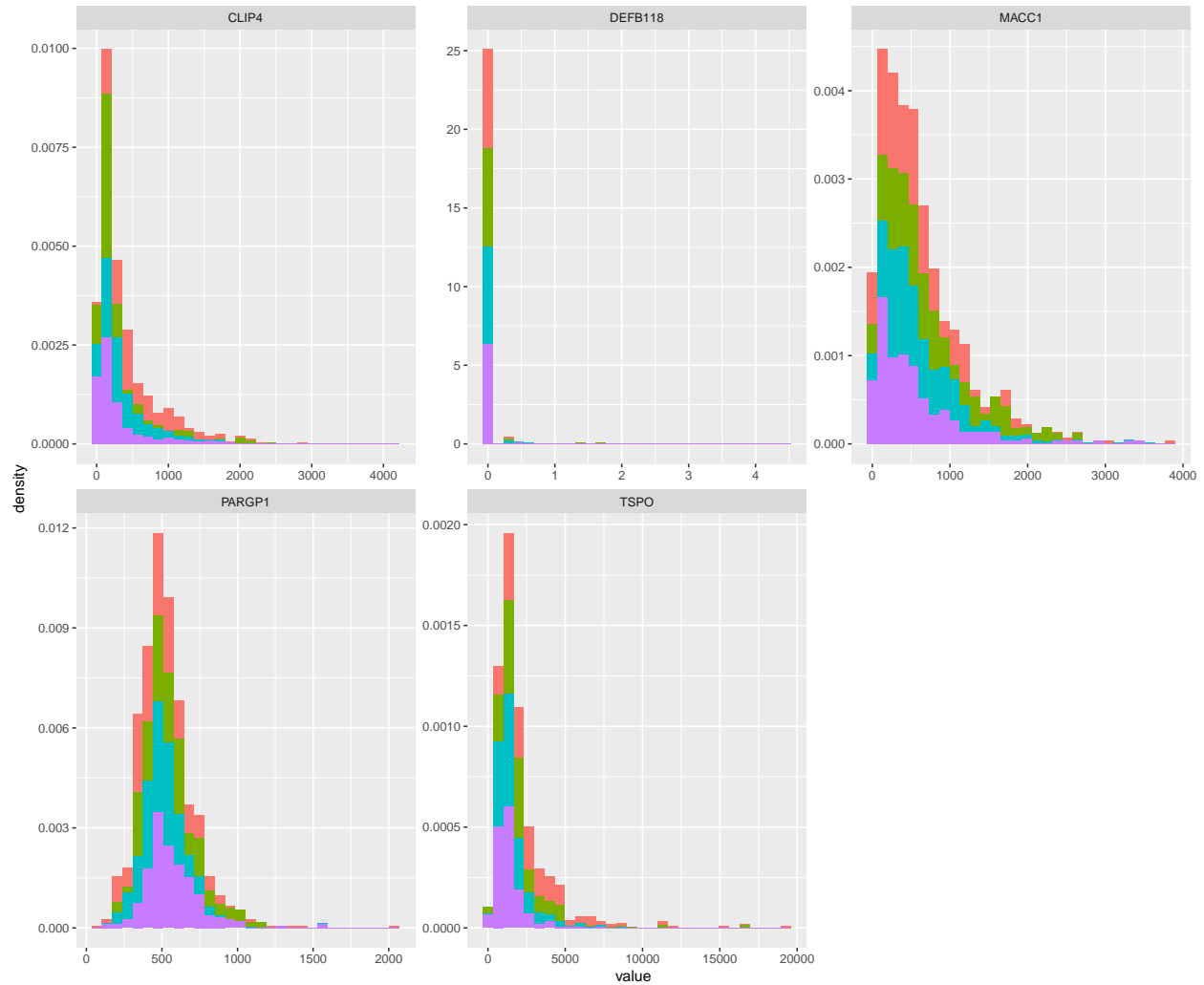
c) Remove gene with zero count and no variability. Then apply logarithmic transform.

278 genes were removed

```
# remove genes with 0 counts
subset_bool <-colSums(abs(brca)) != 0 & apply(brca,2,var) !=0
sel_cols <- which(subset_bool)
brca_sub <- brca[, sel_cols, with=F]

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

dim(brca)
dim(brca_sub)
```

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() get how long it runs
# it can take several minutes.
system.time({brca_sub_kmeans <- kmeans(x = brca_sub, 4)})

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

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

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()`.

Will use 2 PCs for the centered and scaled and 4 for the centered but not scaled

```
# center and scale the data
brca_sub_scaled_centered <- scale(as.matrix(brca_sub), center = T, scale = T)
brca_sub_unscaled_centered <- scale(as.matrix(brca_sub), center = T, scale = F)

# only calculate first few components using SVD by irlba()
## nv = 10: only calculate leading 10
svd_ret <- irlba::irlba(brca_sub_scaled_centered, nv = 10)
svd_ret_us <- irlba::irlba(brca_sub_unscaled_centered, nv = 10)

# Approximate the PVE
svd_var <- svd_ret$d^2/(nrow(brca_sub_scaled_centered)-1)
pve_apx <- svd_var/ncol(brca_sub)

svd_var_us <- svd_ret_us$d^2/(nrow(brca_sub_unscaled_centered)-1)
pve_apx_us <- svd_var_us/ncol(brca_sub)



qplot(c(1:10), pve_apx) +
  geom_line()+
  xlab("Principal Component") +
  ylab("Percentage Variance Explained") +
  ggtitle("Scree Plot")
```
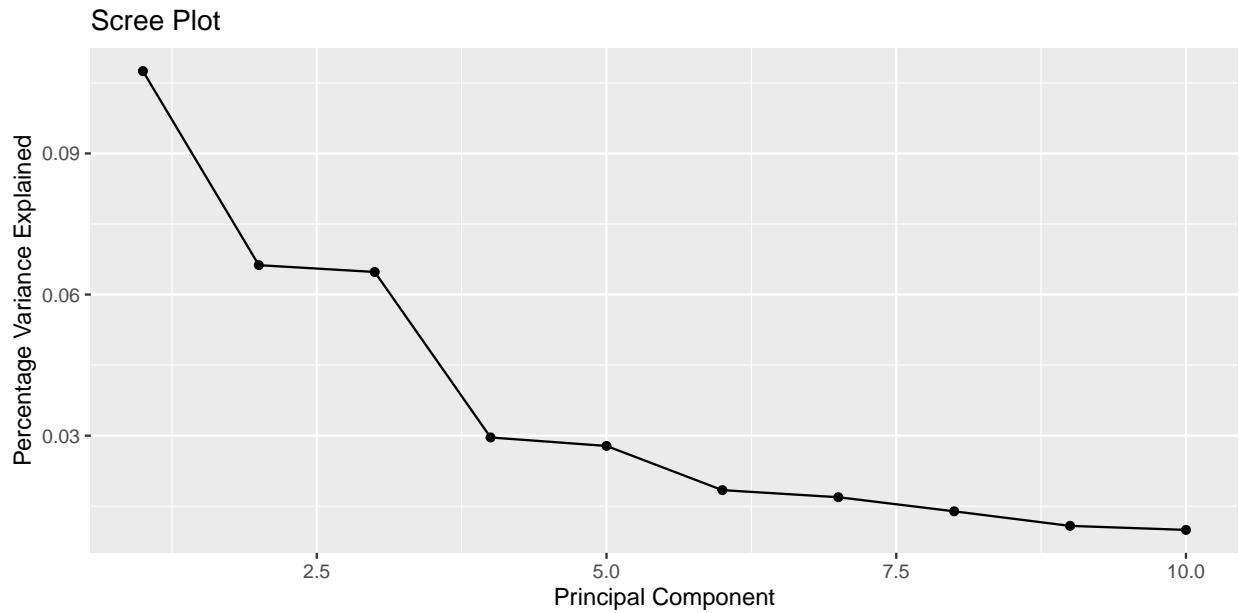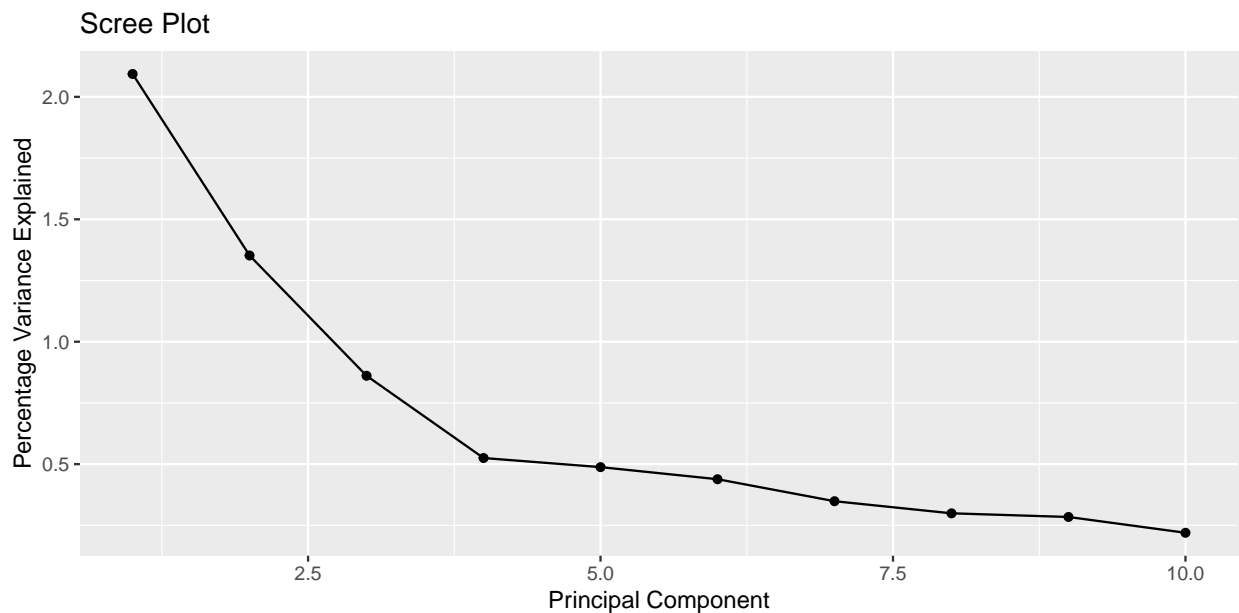
Scree Plot


```
qplot(c(1:10), pve_apx_us) +
  geom_line()+
  xlab("Principal Component") +
  ylab("Percentage Variance Explained") +
  ggtitle("Scree Plot")
```

Scree Plot

b) Plot PC1 vs PC2 of the centered and scaled data and PC1 vs PC2 of the centered but unscaled data side by side. Should we scale or not scale for clustering process? Why? (Hint: to put plots side by side, use `gridExtra::grid.arrange()` or `ggpubr::ggrrange()` or `egg::ggrrange()` for ggplots; use `fig.show="hold"` as chunk option for base plots)

```r
# pc scores
pc_score <- (svd_ret$u[, 1:10])*(svd_ret$d[1:10])
pc_score_us <- (svd_ret_us$u[, 1:10])*(svd_ret_us$d[1:10])

scaled_centered <- data.table(x = pc_score[,1],
                y = pc_score[,2],
                col = as.factor(brca_subtype)) %>%
  ggplot() +
  geom_point(aes(x = x, y = y, col = col)) +
  theme_bw() +
  labs(color = "Cancer SubType") +
  xlab("PC1") +
  ylab("PC2") +
  ggtitle("Scaled and Centered")

unscaled_centered <- data.table(x = pc_score_us[,1],
                y = pc_score_us[,2],
                col = as.factor(brca_subtype)) %>%
  ggplot() +
  geom_point(aes(x = x, y = y, col = col)) +
  theme_bw() +
  labs(color = "Cancer Subtype") +
  xlab("PC1") +
  ylab("PC2") +
  ggtitle("Unscaled and Centered")

grid.arrange(scaled_centered, unscaled_centered, ncol = 2)
```
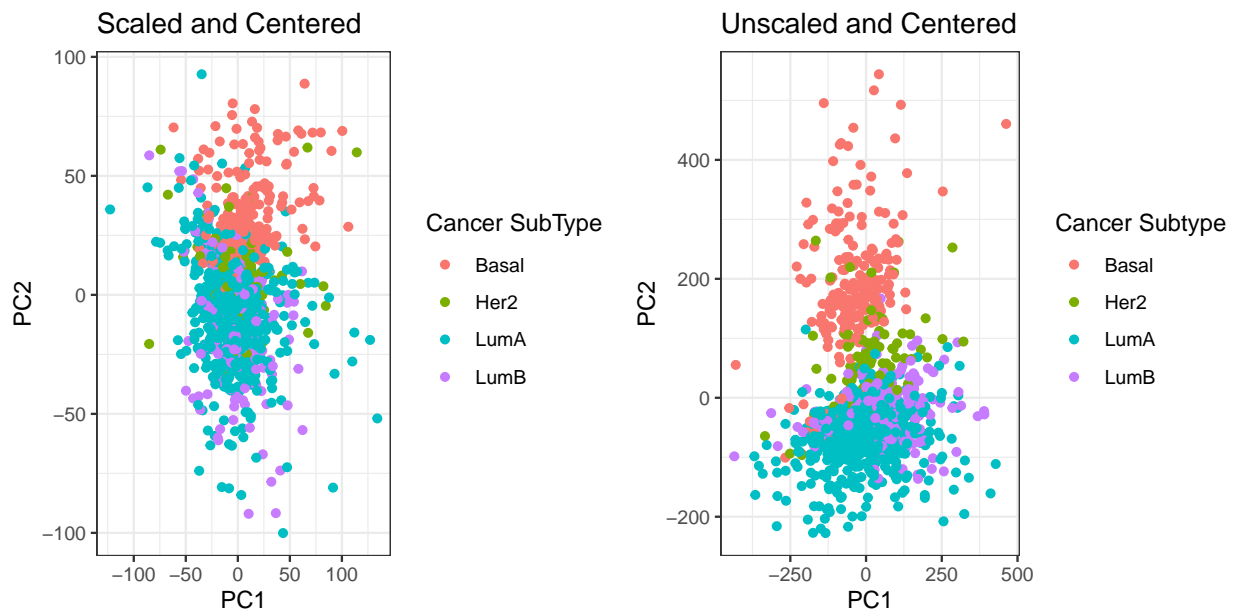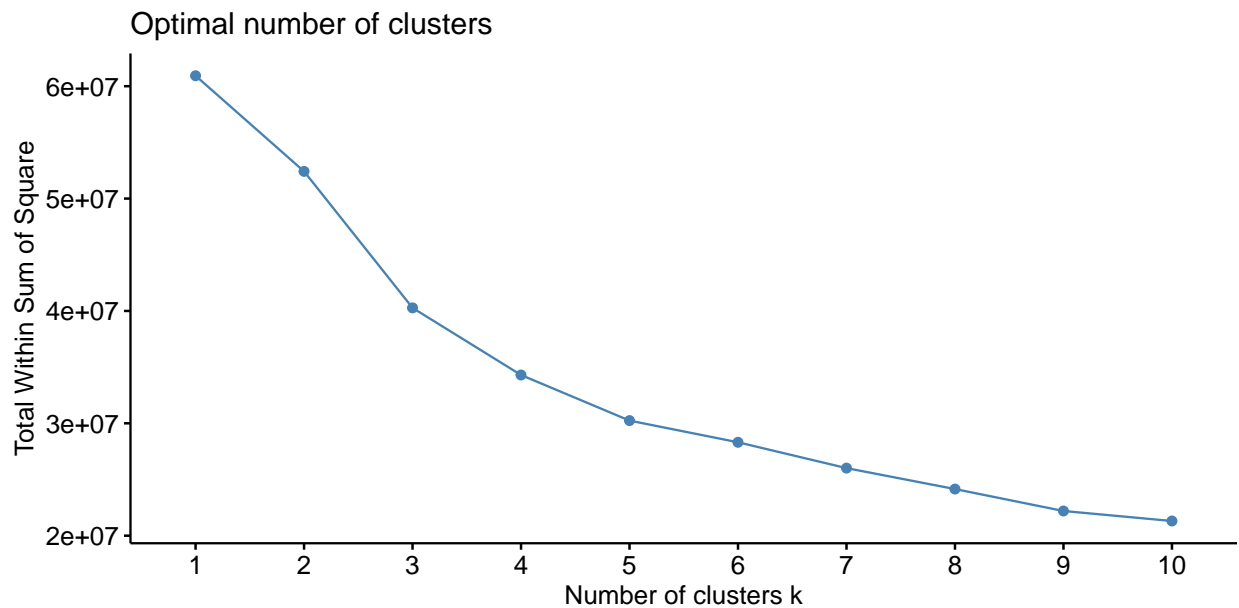

4. Spectrum clustering: center but do not scale the data

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.

10

```
# elbow methodfor optimal number of clusters
fviz_nbclust(pc_score_us[, 1:4], kmeans, method = "wss")
```



```
# 4 clusters
kmean_unscaled <- kmeans(x = pc_score_us[, 1:4], 3)
```

b) Choose an optimal cluster number and apply kmeans. Compare the real sub-type and the clustering
   label as follows: Plot scatter plot of PC1 vs PC2. Use point color to indicate the true cancer type and
   point shape to indicate the clustering label. Plot the kmeans centroids with black dots. Summarize
   how good is clustering results compared to the real sub-type.

```
centroids <- data.frame(x = kmean_unscaled$centers[,1],
                        y = kmean_unscaled$centers[,2],
                        col =levels(brca_subtype)[3],
                        cl = factor(1:3))

data.table(x = pc_score_us[,1],
           y = pc_score_us[,2],
           col = as.factor(brca_subtype),
           cl = as.factor(kmean_unscaled$cluster)) %>%
  ggplot(aes(x = x, y = y, col = col)) +
  geom_point(aes(shape = cl))+
  geom_point(data = centroids,color = "black", size = 3, stroke = 1)+
  theme_bw() +
  labs(color = "Cancer Subtype", shape = "Cluster") +
  xlab("PC1") +
  ylab("PC2") +
  ggtitle("Unscaled and Centered")
```

Unscaled and Centered
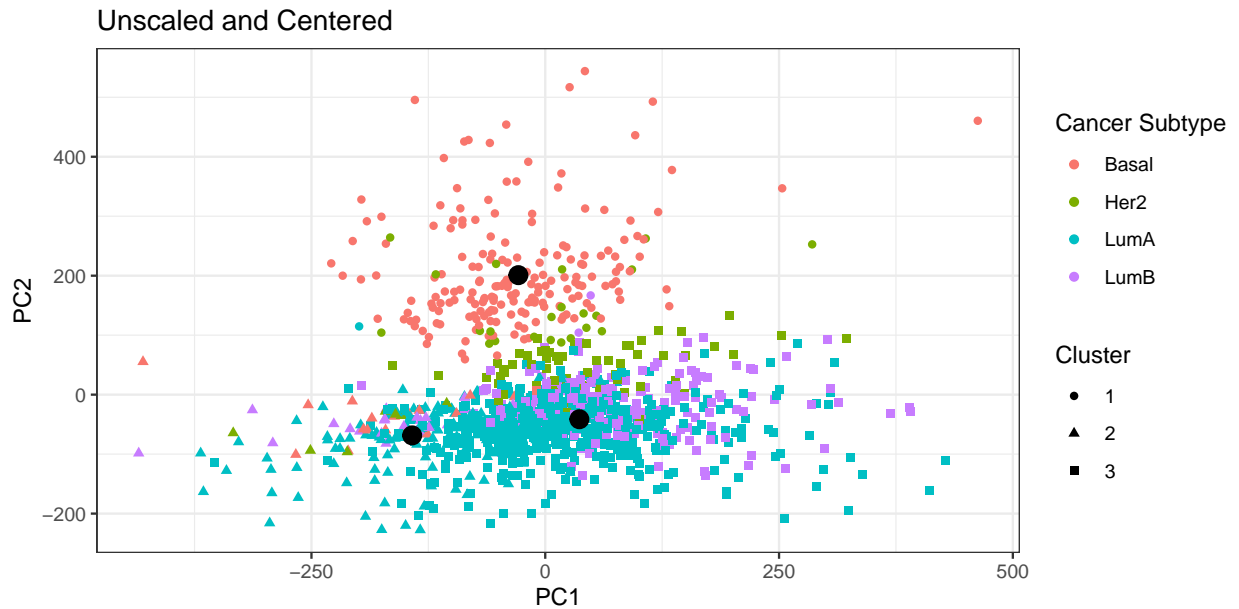
c) Compare the clustering result from applying kmeans to the original data and the clustering result from applying kmeans to 4 PCs. Does PCA help in kmeans clustering? What might be the reasons if PCA helps?

```r
kmean_brca <- kmeans(x = brca_sub, 4)

centroids_brca <- data.frame(x = kmean_brca$centers[,1],
                             y = kmean_brca$centers[,2],
                             col =levels(brca_subtype),
                             cl = factor(1:4))

# run kmeans on data but what x and y should you use??

data.table(x = pc_score_us[,1],
              y = pc_score_us[,2],
              col = as.factor(brca_subtype),
              cl = as.factor(kmean_brca$cluster)) %>%
  ggplot(aes(x = x, y = y, col = col)) +
  geom_point(aes(shape = cl))+
  geom_point(data = centroids_brca,color = "black", size = 3, stroke = 1)+
  theme_bw() +
  labs(color = "Cancer Subtype", shape = "Cluster") +
  xlab("PC1") +
  ylab("PC2") +
  ggtitle("Unscaled and Centered")
```

Unscaled and Centered
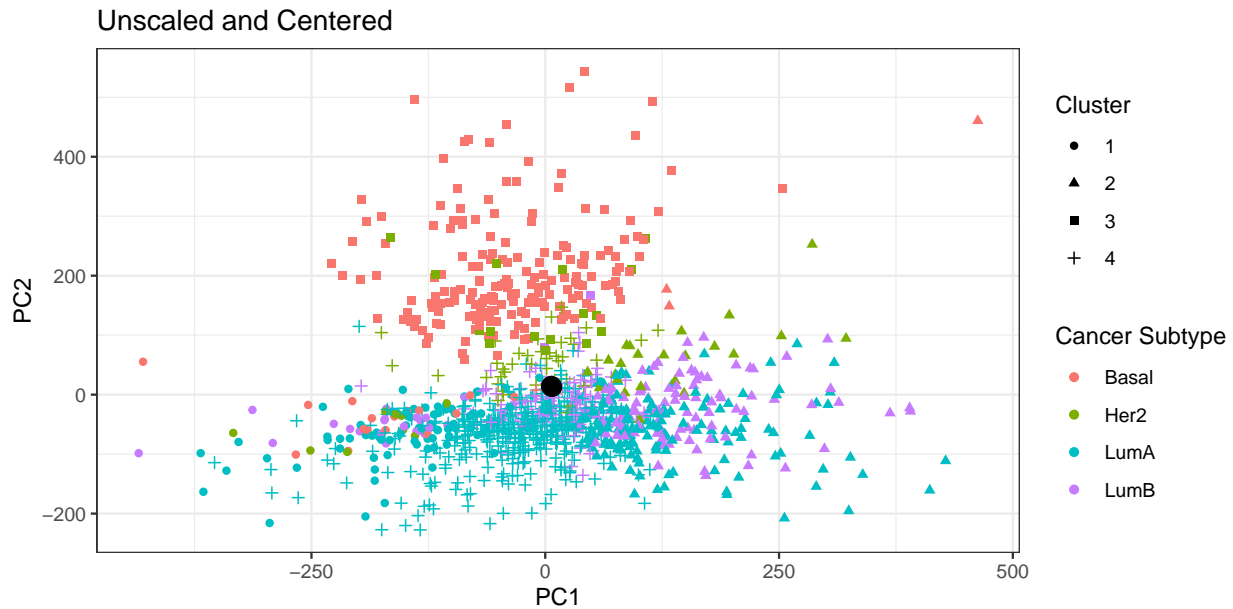
d) Now we have an x patient with breast cancer but with unknown sub-type. We have this patient's mRNA sequencing data. Project this x patient to the space of PC1 and PC2. (Hint: remember we remove some gene with no counts or no variablity, take log and centered) Plot this patient in the plot in iv) with a black dot. Calculate the Euclidean distance between this patient and each of centroid of the cluster. Can you tell which sub-type this patient might have?

```r
x_patient <- read.csv("data/brca_x_patient.csv.gz", header = T) %>%
  select(which(subset_bool))
x_patient <- log2(x_patient+1e-10)

scaled_x_patient<- scale(x_patient, center= colMeans(brca_sub))
x_patient_pred <- scaled_x_patient %*% svd_ret$v


x_patient_projection <- data.frame(x = x_patient_pred[,1],
                        y = x_patient_pred[,2],
                        col =levels(brca_subtype)[1],
                        cl = 1)

data.table(x = pc_score_us[,1],
           y = pc_score_us[,2],
           col = as.factor(brca_subtype),
           cl = as.factor(kmean_unscaled$cluster)) %>%
  ggplot(aes(x = x, y = y, col = col)) +
  geom_point(aes(shape = cl))+
  geom_point(data = x_patient_projection,color = "black", size = 3, stroke = 1)+
  theme_bw() +
  labs(color = "Cancer Subtype", shape = "Cluster") +
  xlab("PC1") +
  ylab("PC2") +
  ggtitle("Unscaled and Centered")
```
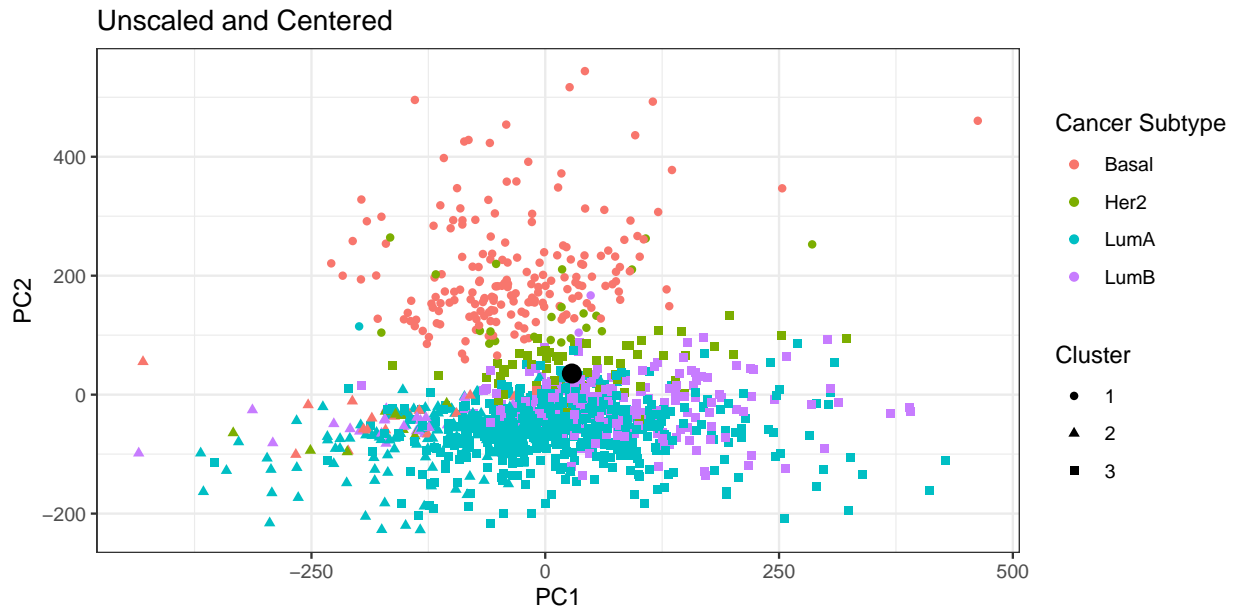
13

Unscaled and Centered

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.

```
?ISLR::Auto
```

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.

- mpg summary

```
summary(Auto$mpg)

mpg_box <- ggplot(Auto, aes(y=mpg)) +
  geom_boxplot()
```

- cylinders summary

```
summary(Auto$cylinders)

cyl_box <- ggplot(Auto, aes(y=cylinders)) +
  geom_boxplot()
```

- displacement summary

```
summary(Auto$displacement)

disp_box <- ggplot(Auto, aes(y=displacement)) +
  geom_boxplot()
```

- horsepower summary

```
summary(Auto$horsepower)

horse_box <- ggplot(Auto, aes(y=horsepower)) +
  geom_boxplot()
```

- weight summary

```
summary(Auto$weight)

weight_box <- ggplot(Auto, aes(y=weight)) +
  geom_boxplot()
```
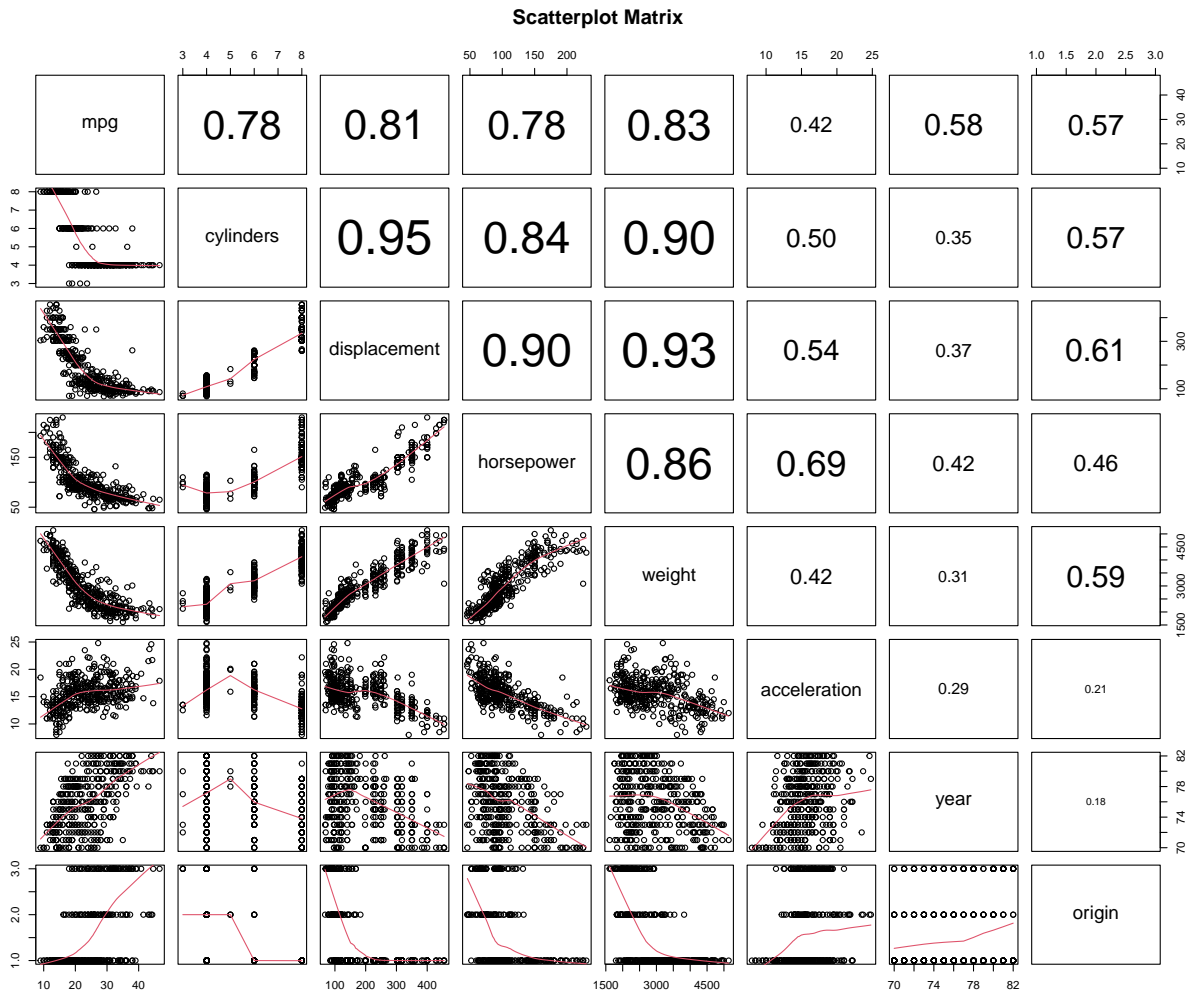
- acceleration summary
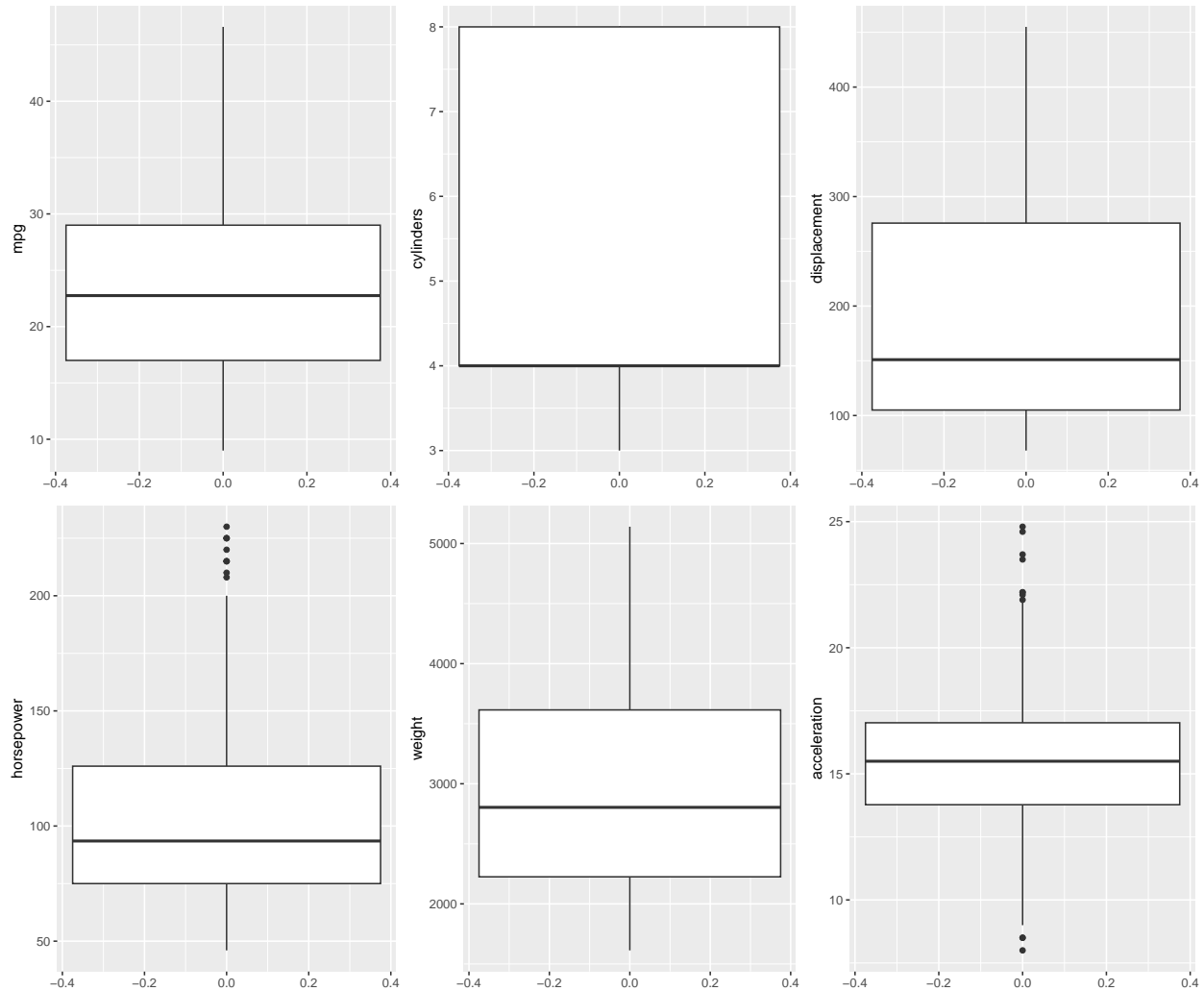
```
summary(Auto$acceleration)

acc_box <- ggplot(Auto, aes(y=acceleration)) +
  geom_boxplot()
```

- year summary
  total 13 years from 1970-1982

```
length(unique(Auto$origin))
summary(as.factor(Auto$origin))
```

- Origin of car American: 245 European: 68 Japanese: 79
- Auto names
  unique auto names: 301

Scatterplot Matrix

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)
stargazer(fit1, align=TRUE, ci = TRUE, type = "latex")
```

% Table created by stargazer v.5.2.3 by Marek Hlavac, Social Policy Institute. E-mail: marek.hlavac at gmail.com % Date and time: Sun, Feb 12, 2023 - 11:27:03 % Requires LaTeX packages: dcolumn

Year is significant at the 0.01 level. Our model is saying that for every year that goes by, there is about a 1.230 increase in the mpg of a car.

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. (Table 4)__

17

Table 2:

	Dependent variable:
	mpg
year	1.230***
	(1.060,1.400)
Constant	−70.000***
	(-83.000,-57.000)
Observations	392
R^2	0.337
Adjusted R^2	0.335
Residual Std. Error	6.360 (df = 390)
F Statistic	198.000*** (df = 1; 390)
Note:	*p<0.1; **p<0.05; ***p<0.01

```
fit2 <- lm(mpg ~ year + horsepower, data = Auto)
stargazer(fit2, align=TRUE, ci = TRUE, type = "latex")
```

% Table created by stargazer v.5.2.3 by Marek Hlavac, Social Policy Institute. E-mail: marek.hlavac at gmail.com % Date and time: Sun, Feb 12, 2023 - 11:27:03 % Requires LaTeX packages: dcolumn

Table 3:

	Dependent variable:
	mpg
year	0.657***
	(0.527,0.787)
horsepower	−0.132***
	(-0.144,-0.119)
Constant	−12.700**
	(-23.200,-2.250)
Observations	392
R^2	0.685
Adjusted R^2	0.684
Residual Std. Error	4.390 (df = 389)
F Statistic	424.000*** (df = 2; 389)
Note:	*p<0.1; **p<0.05; ***p<0.01

Year is significant at the 0.01 level. Our model is saying that for every year that passes by, there is about a .657 increase in the mpg of a car. This effect size decreases from the previous one since we added horsepower to the dataset. (Table 5)

   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 confidence intervals got a lot smaller going from (i) to (ii). Since we added more information to the model (`horspower`) this reduces some of the variability that we see when we examine year alone. This reduction in confidence interval means that we are likely getting more precise.

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).

```
fit2 <- lm(mpg ~ year * horsepower, data = Auto)
stargazer(fit2, align=TRUE, ci = TRUE, type = "latex")
```

% Table created by stargazer v.5.2.3 by Marek Hlavac, Social Policy Institute. E-mail: marek.hlavac at gmail.com % Date and time: Sun, Feb 12, 2023 - 11:27:03 % Requires LaTeX packages: dcolumn

Table 4:

	Dependent variable:
	mpg
year	2.190***
	(1.880,2.510)
horsepower	1.050***
	(0.820,1.270)
year:horsepower	−0.016***
	(-0.019,-0.013)
Constant	−127.000***
	(-150.000,-103.000)
Observations	392
R^2	0.752
Adjusted R^2	0.750
Residual Std. Error	3.900 (df = 388)
F Statistic	393.000*** (df = 3; 388)
Note:	*p<0.1; **p<0.05; ***p<0.01

All of the variables are significant at the 0.01 level. Year is an extremely significant variable. Our model is saying that for every year that passes by, there is about a 2.190 increase in the mpg of a car. This effect size increases dramatically from the previous models. (Table 6)

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?

```
fit3 <- lm(mpg ~ cylinders, data = Auto)
stargazer(fit3, align=TRUE, ci = TRUE, type = output_format)
```

Cylinders is significant at the 0.01 level. Our model is saying that for every 1 cylinder added, there is about a 3.560 increase in the mpg of a car. (Table 7)

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`.

```
fit4 <- lm(mpg ~ factor(cylinders), data = Auto)
stargazer(fit4, align=TRUE, ci = TRUE, type = output_format)
```

Only 4 Cylinders is significant at the 0.01 level. Our model is saying that for every 1 cylinder added, there is about a 3.560 increase in the mpg of a car. (Table 7)

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

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?

Yes you can using anova(H_0, H_1). There is strong evidence of rejecting the null hypothesis that fit0: `mpg` is linear in `cylinders` vs. fit1: `mpg` relates to `cylinders` as a categorical variable

```
test_anova <- anova(fit3, fit4)
test_anova
```

```
## Analysis of Variance Table
##
## Model 1: mpg ~ cylinders
## Model 2: mpg ~ factor(cylinders)
##   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
```

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.

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

b) Summarize the effects found.

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 give a 95% CI for your prediction.

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 $\mathbf{x}$ and $\mathbf{y}$ 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 $\mathbf{x}$ 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 $\mathbf{y}$ 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 $\mathbf{y}$-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.

```r
# 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){I l
  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 β_1 (stored in `results$b1`). Does the sampling distribution agree with theory?

ii. How many of your 95% confidence intervals capture the true β_1? Display your confidence intervals graphically.