C1M2_peer_reviewed

June 10, 2023

1 Module 2: Peer Reviewed Assignment

1.0.1 Outline:

The objectives for this assignment:

1. Mathematically derive the values of $\hat{\beta}_0$ and $\hat{\beta}_1$
2. Enhance our skills with linear regression modeling.
3. Learn the uses and limitations of RSS, ESS, TSS and R^2.
4. Analyze and interpret nonidentifiability.

General tips:

1. Read the questions carefully to understand what is being asked.
2. This work will be reviewed by another human, so make sure that you are clear and concise in what your explanations and answers.

[1]:
```
# Load Required Packages
library(RCurl) #a package that includes the function getURL(), which allows for␣
 ↪reading data from github.
library(tidyverse)
```

```
  Attaching packages                                     tidyverse
1.3.0

 ggplot2 3.3.0      purrr   0.3.4
 tibble  3.0.1      dplyr   0.8.5
 tidyr   1.0.2      stringr 1.4.0
 readr   1.3.1      forcats 0.5.0

  Conflicts
tidyverse_conflicts()
 tidyr::complete() masks
RCurl::complete()
 dplyr::filter()   masks
stats::filter()
 dplyr::lag()      masks stats::lag()
```

1.1 Problem 1: Maximum Likelihood Estimates (MLEs)

Consider the simple linear regression model $Y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$ for $i = 1, ..., n$, $\varepsilon_i \sim N(0, \sigma^2)$. In the videos, we showed that the least squares estimator in matrix-vector form is $\widehat{\beta} = (\beta_0, \beta_1)^T = (X^T X)^{-1} X^T \mathbf{Y}$. In this problem, you will derive the least squares estimators for simple linear regression without (explicitly) using linear algebra.

Least squares requires that we minimize

$$f(\mathbf{x}; \beta_0, \beta_1) = \sum_{i=1}^{n} \Big(Y_i - [\beta_0 + \beta_1 x_i] \Big)^2$$

over β_0 and β_1.

1. (a) Taking Derivatives Find the partial derivative of $f(\mathbf{x}; \beta_0, \beta_1)$ with respect to β_0, and the partial derivative of $f(\mathbf{x}; \beta_0, \beta_1)$ with respect to β_1. Recall that the partial derivative with respect to x of a multivariate function $h(x, y)$ is calculated by taking the derivative of h with respect to x while treating y constant.

$$\frac{\partial f}{\partial \beta_0} = -2 \sum_{i=1}^{n} \Big(Y_i - [\beta_0 + \beta_1 x_i] \Big)$$

$$\frac{\partial f}{\partial \beta_1} = -2 \sum_{i=1}^{n} \Big(Y_i - [\beta_0 + \beta_1 x_i] \Big) x_i$$

1. (b) Solving for $\hat{\beta}_0$ and $\hat{\beta}_1$ Use **1. (a)** to find the minimizers, $\widehat{\beta}_0$ and $\widehat{\beta}_1$, of f. That is, set each partial derivative to zero and solve for β_0 and β_1. In particular, show

$$\widehat{\beta}_1 = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(Y_i - \bar{Y})}{\sum_{i=1}^{n} (x_i - \bar{x})^2} \qquad \text{and} \qquad \widehat{\beta}_0 = \bar{Y} - \widehat{\beta}_1 \bar{x}$$

Solving for β_0

$$-2 \sum_{i=1}^{n} \Big(Y_i - [\beta_0 + \beta_1 x_i] \Big) = 0$$

$$-\sum_{i=1}^{n} \beta_0 = -\sum_{i=1}^{n} \Big(Y_i - \beta_1 x_i \Big)$$

$$n\beta_0 = \sum_{i=1}^{n} \Big(Y_i - \beta_1 x_i \Big)$$

$$\hat{\beta}_0 = \frac{1}{n} \sum_{i=1}^{n} Y_i - \frac{\hat{\beta}_1}{n} \sum_{i=1}^{n} x_i$$

$$\hat{\beta}_0 = \bar{Y} - \hat{\beta}_1 \bar{x}_i$$

Solving for β_1

$$-2\sum_{i=1}^{n}\Big(Y_i - [\beta_0 + \beta_1 x_i]\Big)x_i = 0$$

$$\sum_{i=1}^{n}\Big(Y_i - [\beta_0 + \beta_1 x_i]\Big)x_i = 0$$

$$\sum_{i=1}^{n} Y_i x_i - \sum_{i=1}^{n}\beta_0 x_i - \sum_{i=1}^{n}\beta_1 x_i^2 = 0$$

$$\sum_{i=1}^{n}\beta_1 x_i^2 = \sum_{i=1}^{n} Y_i x_i - \sum_{i=1}^{n}\beta_0 x_i$$

$$\beta_1\sum_{i=1}^{n} x_i^2 = \sum_{i=1}^{n} Y_i x_i - \beta_0\sum_{i=1}^{n} x_i$$

$$\beta_1\sum_{i=1}^{n} x_i^2 = \sum_{i=1}^{n} Y_i x_i - (\bar{Y} - \beta_1\bar{x})\sum_{i=1}^{n} x_i$$

$$\beta_1\sum_{i=1}^{n} x_i^2 - \beta_1\bar{x}\sum_{i=1}^{n} x_i = \sum_{i=1}^{n} Y_i x_i - \bar{Y}\sum_{i=1}^{n} x_i$$

$$\beta_1\Big(\sum_{i=1}^{n} x_i^2 - \bar{x}\sum_{i=1}^{n} x_i\Big) = \sum_{i=1}^{n} Y_i x_i - \bar{Y}\sum_{i=1}^{n} x_i$$

Now

$$\sum_{i=1}^{n}(x_i - \bar{x})(Y_i - \bar{Y}) = \sum_{i=1}^{n}(x_i Y_i - x_i\bar{Y} - \bar{x}Y_i + \bar{x}\bar{Y})$$

$$\sum_{i=1}^{n}(x_i - \bar{x})(Y_i - \bar{Y}) = \sum_{i=1}^{n}(x_i Y_i - x_i\bar{Y}) - \bar{x}\sum_{i=1}^{n} Y_i + \sum_{i=1}^{n}\bar{x}\bar{Y}$$

$$\sum_{i=1}^{n}(x_i - \bar{x})(Y_i - \bar{Y}) = \sum_{i=1}^{n}(x_i Y_i - x_i\bar{Y}) - \bar{x}n\bar{Y} + n\bar{x}\bar{Y}$$

$$\sum_{i=1}^{n}(x_i - \bar{x})(Y_i - \bar{Y}) = \sum_{i=1}^{n}(x_i Y_i - x_i\bar{Y})$$

And

$$\sum_{i=1}^{n}(x_i - \bar{x})^2 = \sum_{i=1}^{n}(x_i^2 - 2x_i\bar{x} + \bar{x}^2)$$

$$\sum_{i=1}^{n}(x_i - \bar{x})^2 = \sum_{i=1}^{n} x_i^2 - 2\bar{x}\sum_{i=1}^{n} x_i + \sum_{i=1}^{n}\bar{x}^2$$

$$\sum_{i=1}^{n}(x_i - \bar{x})^2 = \sum_{i=1}^{n} x_i^2 - 2\bar{x}n\bar{x} + n\bar{x}^2$$

$$\sum_{i=1}^{n}(x_i - \bar{x})^2 = \sum_{i=1}^{n} x_i^2 - \bar{x}n\bar{x}$$

$$\sum_{i=1}^{n}(x_i - \bar{x})^2 = \sum_{i=1}^{n} x_i^2 - \bar{x}\sum_{i=1}^{n} x_i$$

By doing both substitutions

$$\beta_1\left(\sum_{i=1}^{n}(x_i - \bar{x})^2\right) = \sum_{i=1}^{n}(x_i - \bar{x})(Y_i - \bar{Y})$$

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(Y_i - \bar{Y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

1.2 Problem 2: Oh My Goodness of Fit!

In the US, public schools have been slowly increasing class sizes over the last 15 years [https://stats.oecd.org/Index.aspx?DataSetCode=EDU_CLASS]. The general cause for this is because it saves money to have more kids per teacher. But how much money does it save? Let's use some of our new regression skills to try and figure this out. Below is an explanation of the variables in the dataset.

Variables/Columns:
School
Per-Pupil Cost (Dollars)
Average daily Attendance
Average Monthly Teacher Salary (Dollars)
Percent Attendance
Pupil/Teacher ratio

Data Source: E.R. Enlow (1938). "Do Small Schools Mean Large Costs?," Peabody Journal of Educaltion, Vol. 16, #1, pp. 1-11

[2]:
```
school.data = read_table("school.dat")
names(school.data) = c("school", "cost", "avg.attendance", "avg.salary", "pct.
 ↪attendance", "pup.tch.ratio")
head(school.data)
dim(school.data)
```

```
Parsed with column specification:
cols(
  Adair = col_character(),
  `66.90` = col_double(),
  `451.4` = col_double(),
  `160.22` = col_double(),
  `90.77` = col_double(),
  `33.8` = col_double()
)
```

A tibble: 6 × 6

school <chr>	cost <dbl>	avg.attendance <dbl>	avg.salary <dbl>	pct.attendance <dbl>	pup.tch.ratio <dbl>
Calhoun	108.57	219.1	161.79	89.86	23.0
Capitol View	70.00	268.9	136.37	92.44	29.4
Connally	49.04	161.7	106.86	92.01	29.4
Couch	71.51	422.1	147.17	91.60	29.2
Crew	61.08	440.6	146.24	89.32	36.3
Davis	105.21	139.4	159.79	86.51	22.6

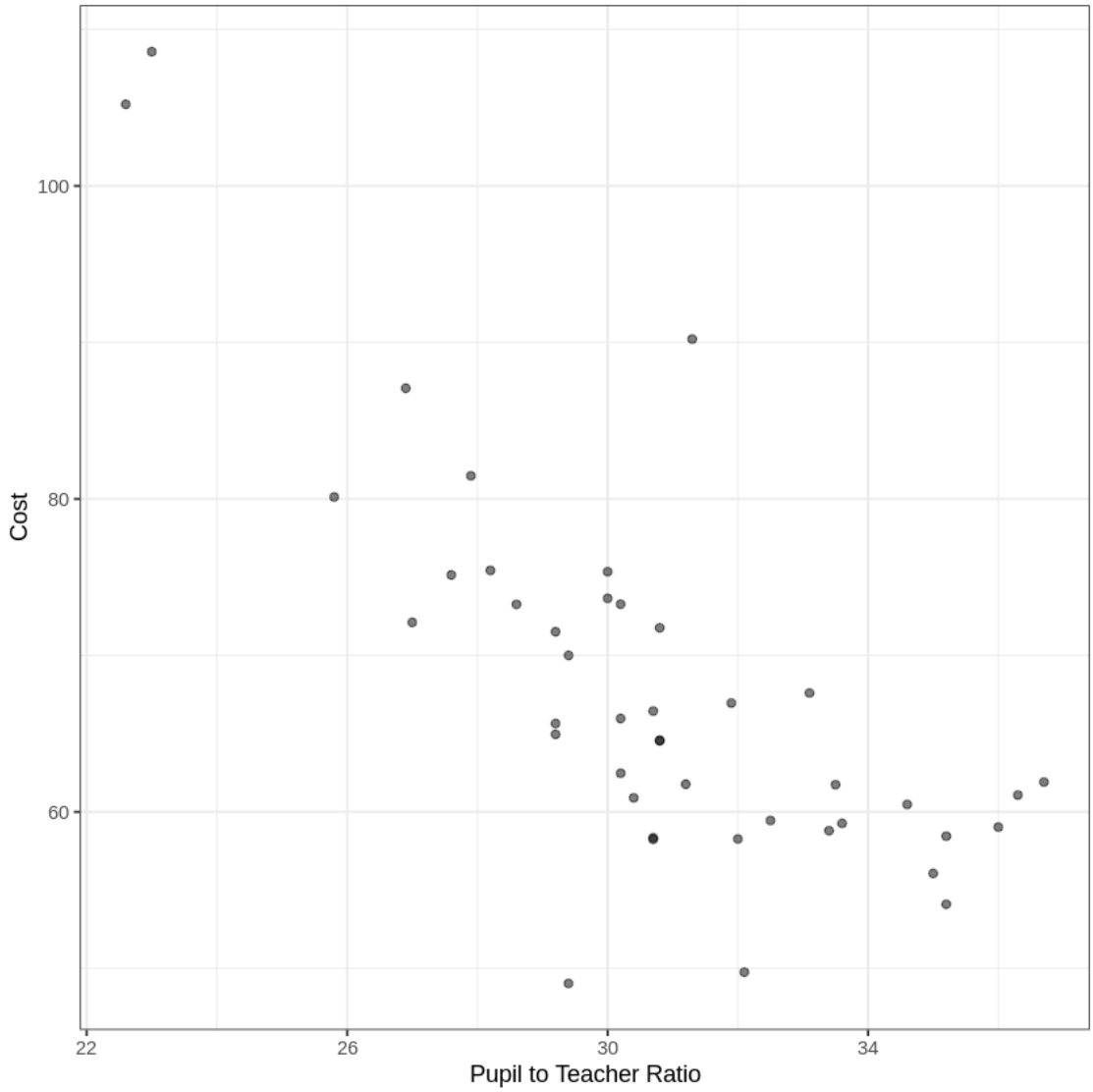
1. 43 2. 6

2. (a) Create a model Begin by creating two figures for your model. The first with `pup.tch.ratio` on the x-axis and `cost` on the y-axis. The second with `avg.salary` on the x-axis and `cost` on the y-axis. Does there appear to be a relation between these two predictors and the response.

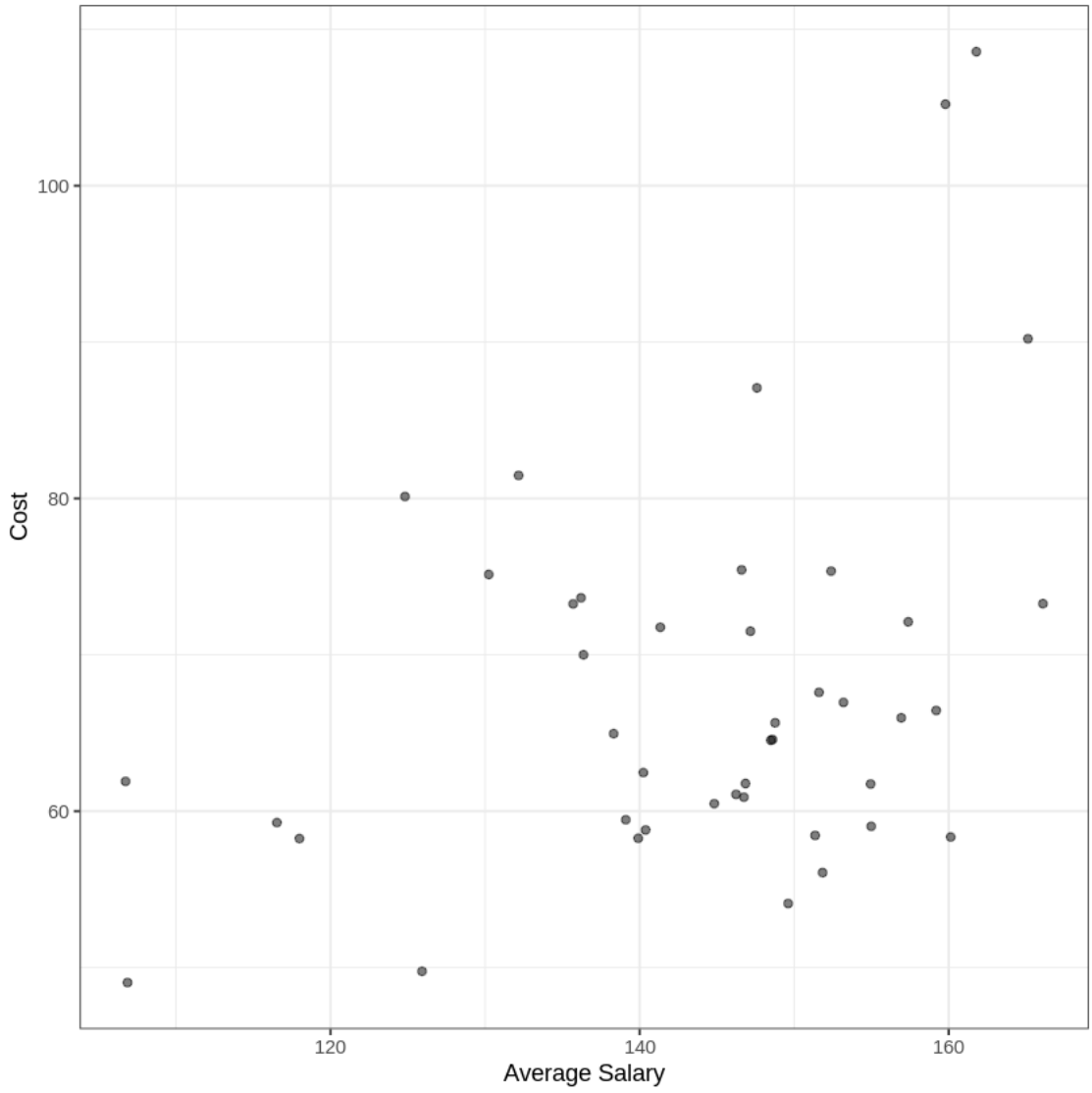
Then fit a multiple linear regression model with `cost` as the response and `pup.tch.ratio` and `avg.salary` as predictors.

[3]:
```
# Your Code Here

pup.tch.ratio <- school.data$pup.tch.ratio
avg.salary <- school.data$avg.salary

cost <- school.data$cost
```

[4]:
```
ggplot(school.data, aes(x = pup.tch.ratio, y = cost)) +
    geom_point( alpha = 0.5) +
    xlab("Pupil to Teacher Ratio") + ylab("Cost")+
    theme_bw()
```



[5]:
```
ggplot(school.data, aes(x = avg.salary, y = cost)) +
    geom_point( alpha = 0.5) +
    xlab("Average Salary") + ylab("Cost")+
    theme_bw()
```



[6]:
```
School.MLR <- lm(cost ~ pup.tch.ratio + avg.salary, school.data)
summary(School.MLR)
```

```
Call:
lm(formula = cost ~ pup.tch.ratio + avg.salary, data = school.data)

Residuals:
     Min       1Q   Median       3Q      Max 
-13.8290  -5.2752  -0.8332   3.8253  19.6986 

Coefficients:
               Estimate Std. Error t value Pr(>|t|)    
(Intercept)   120.23756   17.73230   6.781 3.79e-08 ***
```

```
pup.tch.ratio  -2.82585    0.37714  -7.493 3.90e-09 ***
avg.salary      0.24061    0.08396   2.866   0.0066 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 7.721 on 40 degrees of freedom
Multiple R-squared:  0.6372,Adjusted R-squared:  0.6191
F-statistic: 35.13 on 2 and 40 DF,  p-value: 1.559e-09
```

2. (b) RSS, ESS and TSS In the code block below, manually calculate the RSS, ESS and TSS for your MLR model. Print the results.

[7]:
```
# Your Code Here

beta0 <- School.MLR$coefficients[1]
beta1 <- School.MLR$coefficients[2]
beta2 <- School.MLR$coefficients[3]

costFunction = function(put.tch.ratio, avg.salary){
    return(beta0 + beta1*put.tch.ratio + beta2*avg.salary)
}

cost_hat <- costFunction(school.data$pup.tch.ratio, school.data$avg.salary)
cost_bar <- mean(school.data$cost)
```

[8]:
```
RSS <- sum((school.data$cost - cost_hat)^2)  #Difference between estimate and␣
 ↪data
ESS <- sum((cost_hat - cost_bar)^2)          #Difference between estimate and␣
 ↪mean of the data
TSS <- sum((school.data$cost - cost_bar)^2)  #Difference between data and mean␣
 ↪of the data
```

[9]:
```
cat("RSS is ", RSS, "\n")
cat("ESS is ", ESS, "\n")
cat("TSS is ", TSS, "\n")
```

```
RSS is  2384.597
ESS is  4188.568
TSS is  6573.165
```

2. (c) Are you Squared? Using the values from **2.b**, calculate the R^2 value for your model. Check your results with those produced from the `summary()` statement of your model.

In words, describe what this value means for your model.

[10]:
```
# Your Code Here

R2 <- 1 - RSS/TSS
cat("R Squared is ", R2)
```

```
R Squared is  0.6372224
```

It measures the fit of the model to the data. The closer the value of R^2 is to 1 means a better fit. However it is necessary to remember that R^2 does not prove causation between the predictors and the response, and this means that the model might be completely wrong even if we have a high coefficient of determination.

2. (d) Conclusions Describe at least two advantages and two disadvantages of the R^2 value.

Advantages - It is easy to see if the data fits to the model with just one number. - It is easy to calculate and understand.

Disadvantages - It is misleading in some cases to think that a low value of R^2 indicates a bad model, due to natural noise in the data. - It doesn't explain causality.

2 Problem 3: Identifiability

This problem might require some outside-of-class research if you haven't taken a linear algebra/matrix methods course.

Matrices and vectors play an important role in linear regression. Let's review some matrix theory as it might relate to linear regression.

Consider the system of linear equations

$$Y_i = \beta_0 + \sum_{j=1}^{p} \beta_j x_{i,j} + \varepsilon_i, \tag{1}$$

for $i = 1, ..., n$, where n is the number of data points (measurements in the sample), and $j = 1, ..., p$, where

1. $p + 1$ is the number of parameters in the model.
2. Y_i is the i^{th} measurement of the *response variable*.
3. $x_{i,j}$ is the i^{th} measurement of the j^{th} *predictor variable*.
4. ε_i is the i^{th} *error term* and is a random variable, often assumed to be $N(0, \sigma^2)$.
5. β_j, $j = 0, ..., p$ are *unknown parameters* of the model. We hope to estimate these, which would help us characterize the relationship between the predictors and response.

3. (a) MLR Matrix Form Write the equation above in matrix vector form. Call the matrix including the predictors X, the vector of Y_is $\mathbf{Y}$, the vector of parameters β, and the vector of error terms ε. (This is more LaTeX practice than anything else...)**

$$\mathbf{Y} = X\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$

3. (b) Properties of this matrix In lecture, we will find that the OLS estimator for β in MLR is $\widehat{\boldsymbol{\beta}} = (X^T X)^{-1} X^T \mathbf{Y}$. Use this knowledge to answer the following questions:

1. What condition must be true about the columns of X for the "Gram" matrix $X^T X$ to be invertible?
2. What does this condition mean in practical terms, i.e., does X contain a deficiency or redundancy?
3. Suppose that the number of measurements (n) is less than the number of model parameters ($p+1$). What does this say about the invertibility of $X^T X$? What does this mean on a practical level?
4. What is true about about $\widehat{\boldsymbol{\beta}}$ if $X^T X$ is not invertible?

1. The columns of X must be linearly independent.

2. The meaning of the first point is that the columns must not contain redundant information. A column shouldn't be a multiple of another one.
3. If the number of measurements (n) is less than the number of model parameters ($p+1$), then $X^T X$ is not invertible. On a practical level it means that we require more measurements.
4. If $X^T X$ is non-invertible there is not a unique $\widehat{\boldsymbol{\beta}}$ that satisfies $\widehat{\boldsymbol{\beta}} = (X^T X)^{-1} X^T \mathbf{Y}$. This is what is called the problem of non-identifiability.

2.1 Problem 4: Downloading...

The following data were collected to see if time of day madea difference on file download speed. A researcher placed a file on a remote server and then proceeded to download it at three different time periods of the day. They downloaded the file 48 times in all, 16 times at each Time of Day (`time`), and recorded the Time in seconds (`speed`) that the download took.

4. (a) Initial Observations The `downloading` data is loaded in and cleaned for you. Using `ggplot`, create a boxplot of `speed` vs. `time`. Make some basic observations about the three categories.

[11]:
```
# Load in the data and format it
downloading = read.csv("downloading.txt", sep="\t")
names(downloading) = c("time", "speed")
# Change the types of brand and form to categories, instead of real numbers
downloading$time = as.factor(downloading$time)
summary(downloading)
```

```
                time         speed
 Early (7AM)      :16   Min.   : 68.0
 Evening (5 PM)   :16   1st Qu.:129.8
```

```
 Late Night (12 AM):16   Median :198.0
                         Mean   :193.2
                         3rd Qu.:253.0
                         Max.   :367.0
```

[12]:
```
head(downloading)
```

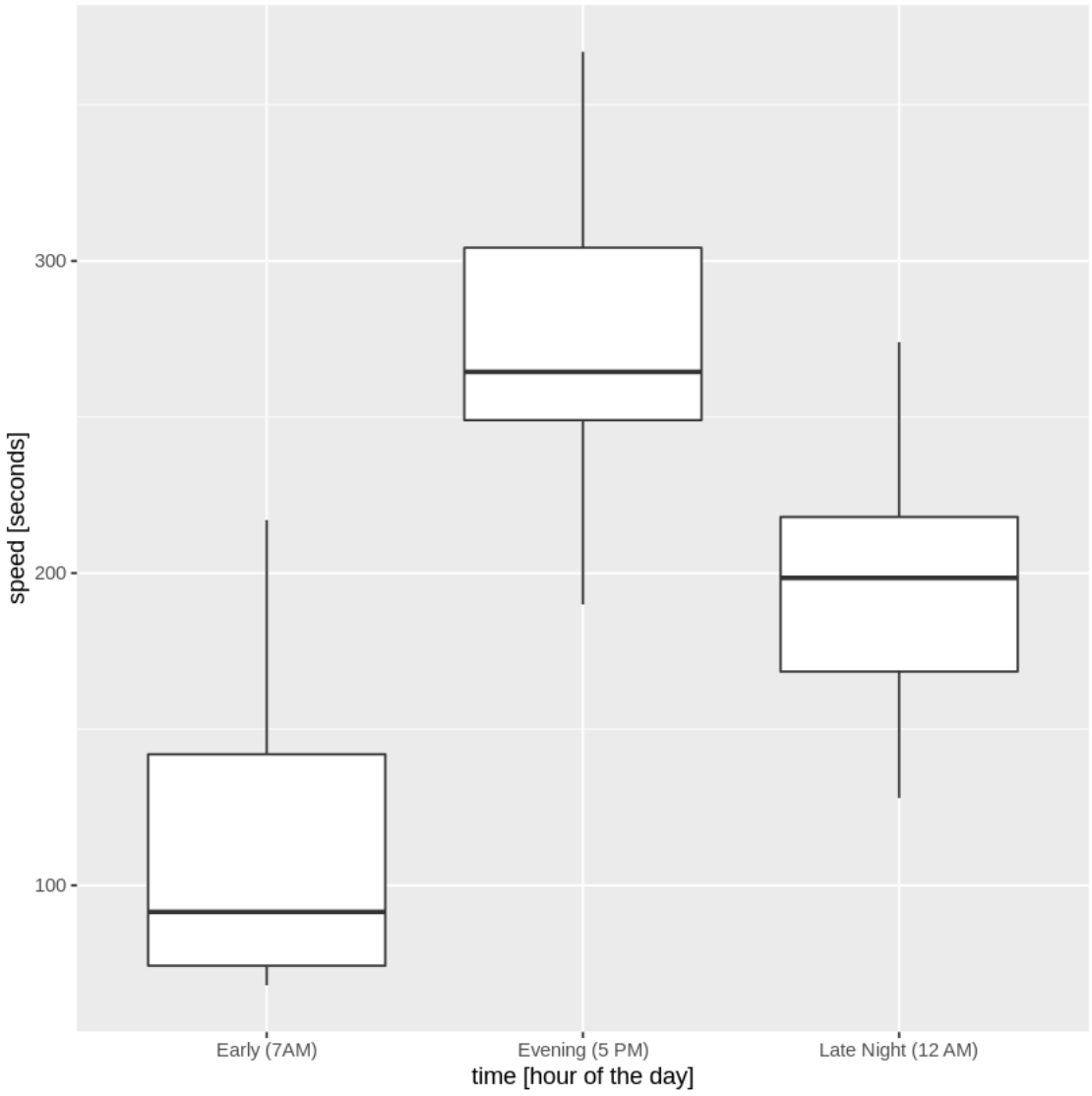
A data.frame: 6 × 2

	time <fct>	speed <int>
1	Early (7AM)	68
2	Early (7AM)	138
3	Early (7AM)	75
4	Early (7AM)	186
5	Early (7AM)	68
6	Early (7AM)	217

[13]:
```
summary(lm(speed ~ time, data = downloading))
```

```
Call:
lm(formula = speed ~ time, data = downloading)

Residuals:
    Min      1Q  Median      3Q     Max
-83.312 -34.328  -5.187  26.250 103.625

Coefficients:
                      Estimate Std. Error t value Pr(>|t|)
(Intercept)             113.37      11.79   9.619 1.73e-12 ***
timeEvening (5 PM)      159.94      16.67   9.595 1.87e-12 ***
timeLate Night (12 AM)   79.69      16.67   4.781 1.90e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 47.15 on 45 degrees of freedom
Multiple R-squared:  0.6717,Adjusted R-squared:  0.6571
F-statistic: 46.03 on 2 and 45 DF,  p-value: 1.306e-11
```

[14]:
```
ggplot(data = downloading, aes(x = time, y = speed)) +
  geom_boxplot() +
  labs(x = "time [hour of the day]", y = "speed [seconds]")
```



- You get lower download times early in the day most of the time.
- There might be an outlier for the "Early (7AM)" category, since the distribution of the values is too skewed down.
- The maximum speed is approximately 8 times bigger than the minimum one.
- The medians for the different categories are approximately:
  - "Early (7AM)": 100
  - "Evening (5PM)": 250
  - "Late Night (12 AM)": 200

4. (b) How would we model this? Fit a regression to these data that uses `speed` as the response and `time` as the predictor. Print the summary. Notice that the result is actually *multiple* linear regression, not simple linear regression. The model being used here is:

$$Y_i = \beta_0 + \beta_1 X_{i,1} + \beta_2 X_{i,2} + \varepsilon_i$$

where

1. $X_{i,1} = 1$ if the i^{th} download is made in the evening (5 pm).
2. $X_{i,2} = 1$ if the i^{th} download is made at night (12 am).

Note: If $X_{i,1} = 0$ and $X_{i,2} = 0$, then the i^{th} download is made in the morning (7am).

To confirm this is the model being used, write out the explicit equation for your model - using the parameter estimates from part (a) - and print out it's design matrix.

[15]:
```
# Your Code Here

downloading.MLR = lm(speed ~ time, data = downloading)
summary(downloading.MLR)
```

```
Call:
lm(formula = speed ~ time, data = downloading)

Residuals:
    Min      1Q  Median      3Q     Max
-83.312 -34.328  -5.187  26.250 103.625

Coefficients:
                       Estimate Std. Error t value Pr(>|t|)
(Intercept)              113.37      11.79   9.619 1.73e-12 ***
timeEvening (5 PM)       159.94      16.67   9.595 1.87e-12 ***
timeLate Night (12 AM)    79.69      16.67   4.781 1.90e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 47.15 on 45 degrees of freedom
Multiple R-squared:  0.6717,Adjusted R-squared:  0.6571
F-statistic: 46.03 on 2 and 45 DF,  p-value: 1.306e-11
```

Explicit Model Equation

$$Y = 113.37 + 159.94 \cdot \text{timeEvening} + 79.69 \cdot \text{timeLateNight}$$

Design Matrix

[40]:
```
designMatrix1 = model.matrix(downloading.MLR)
designMatrix1
```

A matrix: 48 × 3 of type dbl

	(Intercept)	timeEvening (5 PM)	timeLate Night (12 AM)
1	1	0	0
2	1	0	0
3	1	0	0
4	1	0	0
5	1	0	0
6	1	0	0
7	1	0	0
8	1	0	0
9	1	0	0
10	1	0	0
11	1	0	0
12	1	0	0
13	1	0	0
14	1	0	0
15	1	0	0
16	1	0	0
17	1	1	0
18	1	1	0
19	1	1	0
20	1	1	0
21	1	1	0
22	1	1	0
23	1	1	0
24	1	1	0
25	1	1	0
26	1	1	0
27	1	1	0
28	1	1	0
29	1	1	0
30	1	1	0
31	1	1	0
32	1	1	0
33	1	0	1
34	1	0	1
35	1	0	1
36	1	0	1
37	1	0	1
38	1	0	1
39	1	0	1
40	1	0	1
41	1	0	1
42	1	0	1
43	1	0	1
44	1	0	1
45	1	0	1
46	1	0	1
47	1	0	1
48	1	0	1

4. (c) Only two predictors? We have three categories, but only two predictors. Why is this the case? To address this question, let's consider the following model:

$$Y_i = \beta_0 + \beta_1 X_{i,1} + \beta_2 X_{i,2} + \beta_2 X_{i,3} + \varepsilon_i$$

where

1. $X_{i,1} = 1$ if the i^{th} download is made in the evening (5 pm).
2. $X_{i,2} = 1$ if the i^{th} download is made at night (12 am).
3. $X_{i,3} = 1$ if the i^{th} download is made in the morning (7 am).

Construct a design matrix to fit this model to the response, speed. Determine if something is wrong with it. Hint: Analyze the design matrix.

[34]:
```
colNames = colnames(designMatrix1)
colNames[4] = 'timeMorning (7 AM)'
colNames
```

1. '(Intercept)' 2. 'timeEvening (5 PM)' 3. 'timeLate Night (12 AM)' 4. 'timeMorning (7 AM)'

[38]:
```
# Your Code Here

designMatrix2 <- matrix(0, nrow=nrow(designMatrix1), ncol=ncol(designMatrix1)+1)
designMatrix2[,1] <- designMatrix1[,1]
designMatrix2[,2] <- designMatrix1[,2]
designMatrix2[,3] <- designMatrix1[,3]
designMatrix2[,4] <- designMatrix2[,1] - designMatrix2[,2] - designMatrix2[,3]
colnames(designMatrix2) <- colNames
print(designMatrix2)
```

```
      (Intercept) timeEvening (5 PM) timeLate Night (12 AM) timeMorning (7 AM)
 [1,]           1                   0                      0                   1
 [2,]           1                   0                      0                   1
 [3,]           1                   0                      0                   1
 [4,]           1                   0                      0                   1
 [5,]           1                   0                      0                   1
 [6,]           1                   0                      0                   1
 [7,]           1                   0                      0                   1
 [8,]           1                   0                      0                   1
 [9,]           1                   0                      0                   1
[10,]           1                   0                      0                   1
[11,]           1                   0                      0                   1
[12,]           1                   0                      0                   1
[13,]           1                   0                      0                   1
[14,]           1                   0                      0                   1
[15,]           1                   0                      0                   1
[16,]           1                   0                      0                   1
[17,]           1                   1                      0                   0
[18,]           1                   1                      0                   0
```

[19,]	1	1	0	0
[20,]	1	1	0	0
[21,]	1	1	0	0
[22,]	1	1	0	0
[23,]	1	1	0	0
[24,]	1	1	0	0
[25,]	1	1	0	0
[26,]	1	1	0	0
[27,]	1	1	0	0
[28,]	1	1	0	0
[29,]	1	1	0	0
[30,]	1	1	0	0
[31,]	1	1	0	0
[32,]	1	1	0	0
[33,]	1	0	1	0
[34,]	1	0	1	0
[35,]	1	0	1	0
[36,]	1	0	1	0
[37,]	1	0	1	0
[38,]	1	0	1	0
[39,]	1	0	1	0
[40,]	1	0	1	0
[41,]	1	0	1	0
[42,]	1	0	1	0
[43,]	1	0	1	0
[44,]	1	0	1	0
[45,]	1	0	1	0
[46,]	1	0	1	0
[47,]	1	0	1	0
[48,]	1	0	1	0

The fourth column of the design matrix is a linear combination of the first three columns, then $X^T X$ is non-invertible.

4. (d) Interpretation Interpret the coefficients in the model from **4.b**. In particular:

1. What is the difference between the mean download speed at 7am and the mean download speed at 5pm?
2. What is the mean download speed (in seconds) in the morning?
3. What is the mean download speed (in seconds) in the evening?
4. What is the mean download speed (in seconds) at night?

[49]:
```
meanMorning = mean(downloading$speed[downloading$time == 'Early (7AM)'])
meanEvening = mean(downloading$speed[downloading$time == 'Evening (5 PM)'])
meanNight = mean(downloading$speed[downloading$time == 'Late Night (12 AM)'])
cat("1. Difference between the mean download speed at 7am and the mean download␣
 ↪speed at 5pm: ", meanEvening-meanMorning, "\n")
```

```
cat("2. Mean download speed (in seconds) in the morning", meanMorning, "\n")
cat("3. Mean download speed (in seconds) in the evening", meanEvening, "\n")
cat("4. Mean download speed (in seconds) in the night", meanNight, "\n")
```

```
1. Difference between the mean download speed at 7am and the mean download speed
at 5pm:  159.9375
2. Mean download speed (in seconds) in the morning 113.375
3. Mean download speed (in seconds) in the evening 273.3125
4. Mean download speed (in seconds) in the night 193.0625
```

[51]:
```
cat("Difference between the mean download speed at 7am and the mean download␣
↪speed at 12am: ", meanNight-meanMorning, "\n")
```

```
Difference between the mean download speed at 7am and the mean download speed at
12am:  79.6875
```

$$Y = 113.37 + 159.94 \cdot \text{timeEvening} + 79.69 \cdot \text{timeLateNight}$$

- The intercept is equal to the mean download speed in the morning.
- The beta parameter for the evening time is the difference between the mean of the evening download speed and the mean of the morning download speed.
- The beta parameter for the night time is very close to the difference between the mean of the night download speed and the mean of the morning download speed.

[ ]:
```

```