STAT GU4206/GR5206 Sample Midterm

Gabriel

3/8/2019

The STAT GU4206/GR5206 Midterm is open notes, open book(s), open computer and online resources are allowed. Students are required to be physically present during the exam. The TA/instructor will be available to answer questions during the exam. Students are **not** allowed to communicate with any other people regarding the exam with the exception of the instructor (Gabriel Young) and course TAs. This includes emailing fellow students, using WeChat and other similar forms of communication. If there is any suspicion of one or more students cheating, further investigation will take place. If students do not follow the guidelines, they will receive a zero on the exam and potentially face more severe consequences. The exam will be posted on Canvas at 10:05AM. Students are required to submit both the .pdf (or .html) and .Rmd files on Canvas by 12:40AM. If students fail to knit the pdf or html file, the TA will take off a significant portion of the grade. Students will also be significantly penalized for late exams. If for some reason you are unable to upload the completed exam on Canvas by 12:40PM, then immediately email markdown file to the course TA.

Important: If you have a bug in your code then **RMarkdown** will not knit. I highly recommend that you comment out any non-working code. That way your file will knit and you will not be penalized for only uploading the Rmd file.

Part I - Character data and regular expressions

Consider the following toy dataset **strings_data.csv**. This dataset has 461 rows (or length 461 using readLines) and consists of random character strings.

```
char_data <- readLines("strings_data.csv")
head(char_data,8)
```

```
## [1] "\"strings\""
## [2] "\"rmJgFZUGKsBlvmuUOuWnFUyziiyWEEhiRROlJJXRXxOwp\""
## [3] "\"bacUqblSKDopCEAYWdgD\""
## [4] "\"qsPuSJdkmv\""
## [5] "\"RXAnEoHlliMllHMPFTcv\""
## [6] "\"SBolTFf0.2nMoQ9.454lKlgjQZGroup_IOMLFgXj\""
## [7] "\"rtoMgy0.36bRrnA9.454goQIJGroup_IMCRp\""
## [8] "\"CqdniznveOdQRhMyctjUEULimqmQjV\""
```

```
length(char_data)
```

```
## [1] 461
```

Among the 461 cases, several rows contain numeric digits and a specific string of the form "Group_Letter", where "Letter"" is a single uppercase letter. For example, the 6th element contains the symbols "0.2","9.454","Group_I".

```
char_data[6]
```

```
## [1] "\"SBolTFf0.2nMoQ9.454lKlgjQZGroup_IOMLFgXj\""
```

```
c("0.2","9.454","Group_I")
```

```
## [1] "0.2"     "9.454"   "Group_I"
```

Problem 1

Your task is to extract the numeric digits and the group variable from this character string vector. Notes:

1. The first number **x** is a single digit followed by a period and at least one digit. There are a few cases where the first number is only a single digit without a period.
2. The second number **y** is one or two digits followed by a period and at least one digit. Note that the second number can be negative or positive.
3. The group value is the string "Group_" followed by a single capital letter. For example "Group_I" and "Group_S" are both elements of the third string of interest.

Once you extract all three symbols, make sure to convert the numeric digits to a numeric mode (use **as.numeric()**) and organize the scrapped information in a dataframe. Your final dataframe should have 230 rows by 3 columns. The first three rows of your dataframe should look like the following output:

```
data.frame(x=c(0.20,0.36,0.56),
           y=c(9.454,9.454,9.454),
           Group=c("Group_I","Group_I","Group_I"))
```

```
##      x     y   Group
## 1 0.20 9.454 Group_I
## 2 0.36 9.454 Group_I
## 3 0.56 9.454 Group_I
```

Solution

```
## Code goes here ------
```

Problem 2

Use both **base R** and **ggplot** to construct a scatterplot of the variables **y** versus **x** and split the colors of the plot by the variable **Group**. Also include a legend, relabel the axes and include a title. Make sure the legend doesn't cover up the plot in base R.

Base R plot

```
## Code goes here ------
```

ggplot plot

```
library(ggplot2)
## Code goes here ------
```

Part II - Data proccessing and exploratory analysis

The data comprise of roughly 25,000 records for males between the age of 18 and 70 who are full time workers. A variety of variables are given for each subject: years of education and job experience, college graduate (yes, no), working in or near a city (yes, no), US region (midwest, northeast, south, west), commuting distance, number of employees in a company, and race (African America, Caucasian, Other). The response variable is weekly wages (in dollars). The data are taken many decades ago so the wages are low compared to current times.

```
salary_data <- read.csv("salary.txt",as.is=T,header=T)
head(salary_data)
```

```
##     wage edu exp city       reg  race deg  com emp
## 1 354.94   7  45  yes northeast white  no 24.3 200
## 2 370.37   9   9  yes northeast white  no 26.2 130
## 3 754.94  11  46  yes northeast white  no 26.4 153
## 4 593.54  12  36  yes northeast other  no  9.9  86
## 5 377.23  16  22  yes northeast white yes  7.1 181
## 6 284.90   8  51  yes northeast white  no 11.4  32
```

Below I am defining a new variable in the **salary_data** dataframe which computes the natural logarithm of wages.

```
salary_data$log_wage <- log(salary_data$wage)
```

Problem 3

Use the **summary()** function on the salary dataset to check if the variables make sense. Specifically, one of the continuous variables has some "funny" values. Remove the rows of the dataframe corresponding to these strange values. If you can't figure this question out, then move on because you can still solve Problem 4 & 5 without Problem 3.

Solution

```
## Code goes here ------
```

Problem 4

Using **ggplot**, plot **log_wages** against work experience, i.e., **x=exp** and **y=log_wages**. In this graphic, change the transparency of the points so that the scatterplot does not look so dense. **Note:** the **alpha** parameter changes the transparency. Also label the plot appropriately.

Solution

```
library(ggplot2)
## Code goes here ------
```

Notice that your graphic constructed from Problem 4 shows a quadratic or curved relationship between **log_wages** against **exp**. The next task is to plot three quadratic functions for each race level "black", "white" and "other". To estimate the quadratic fit, you can use the following function **quad_fit**:

```
quad_fit <- function(data_sub) {
  return(lm(log_wage~exp+I(exp^2),data=data_sub)$coefficients)
}
quad_fit(salary_data)
```

```
##  (Intercept)          exp     I(exp^2)
##  5.680659297  0.061220716 -0.001103711
```

The above function computes the least squares quadratic fit and returns coefficients $\hat{a}_1$,$\hat{a}_2$ and $\hat{a}_3$, where

$$\hat{Y} = \hat{a}_1 + \hat{a}_2 x + \hat{a}_3 x^2$$

and $\hat{Y} = \log(\text{wage})$ and $x = \text{exp}$.

Use **ggplot** to accomplish this task or use base R graphics for partial credit. Make sure to include a legend and appropriate labels.

Solution

```
## Code goes here ------
```

Part III - The Bootstrap

Data and model description

Consider a study that assesses how a drug affects someone's resting heart rate. The study consists of $n = 60$ respondents. The researcher randomly places the respondents into three groups; control group and two dosage groups (20 each). The first drug group is given 200 mg (x_1) and the second drug group is given 500 mg (x_2). She then measures each respondent's resting heart rate 1 hour after the drug was administered (Y). She also measures other characteristics of each respondent; age (x_3), weight (x_4), height (x_5), gender (x_6) and initial resting heart rate before the drug was administered (x_7). The statistical linear regression model is:

$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4 + \beta_5 x_5 + \beta_6 x_6 + \beta_7 x_7 + \epsilon, \quad \epsilon \overset{iid}{\sim} N(0, \sigma^2).$$

There are three dummy variables for this model:

$$x_1 = \begin{cases} 1 & \text{if 200mg} \\ 0 & \text{otherwse} \end{cases} \qquad x_2 = \begin{cases} 1 & \text{if 500mg} \\ 0 & \text{otherwse} \end{cases} \qquad x_6 = \begin{cases} 1 & \text{if male} \\ 0 & \text{otherwse} \end{cases}$$

Based on the above variable coding, the control group is described through the intercept β_0.

Exploratory analysis

The dataset **drugstudy.csv** is read in below.

```
drugstudy <- read.table("drugstudy.txt",header=T)
head(drugstudy)
```

```
##   Final.HR Initial.HR Dose1 Dose2 Age Height Weight Gender
## 1     75.1       73.6     0     0  29   73.1 251.73      1
## 2     71.6       71.7     0     0  34   72.4 151.59      1
## 3     65.5       66.5     0     0  25   67.2 133.89      1
## 4     77.2       72.7     0     0  39   69.8 154.91      1
## 5     75.8       75.8     0     0  32   72.7 186.59      1
## 6     67.9       68.7     0     0  25   66.4 205.77      1
```

Problem 5

Compute the average final resting heart rate for each drug group. Also compute the average initial resting heart rate for each drug group. Display the results in dataframe or table.

Solution

```
## Code goes here ------
```

Problem 6

Construct a comparative boxplot of the respondents final resting heart rate for each drug group. Use base R or ggplot. Make sure to label the plot appropriately.

Solution

```
## Code goes here ------
```

Nonparametric analysis (bootstrap)

Consider a nonparametric approach to assess the drug's impact on final resting heart rate. More specifically, the researcher is going to perform a bootstrap procedure on the following parameters:

1. β_1
2. β_2
3. $\beta_1 - \beta_2$

The final bootstrap intervals incorporate the three testing procedures:

1. $H_0 : \beta_1 = 0 \quad \text{vs.} \quad H_A : \beta_1 \neq 0$
2. $H_0 : \beta_2 = 0 \quad \text{vs.} \quad H_A : \beta_2 \neq 0$
3. $H_0 : \beta_1 - \beta_2 = 0 \quad \text{vs.} \quad H_A : \beta_1 - \beta_2 \neq 0$

When testing $\beta_1 = 0$, we are investigating the impact of the 200mg dosage group versus the control group. Similarly, when testing $\beta_2 = 0$, we are investigating the impact of the 500mg dosage group versus the control group. The third test $\beta_1 - \beta_2 = 0$ is describing if the low dosage group has the same impact on resting heart rate as the high dosage group.

Problem 7

Perform the follwong tasks!

Run a bootstrap procedure on parameters β_1, β_2 and $\beta_1 - \beta_2$. (i) Construct a table or dataframe displaying the least squares estimators of $\hat{\beta}_1, \hat{\beta}_2$ and $\hat{\beta}_1 - \hat{\beta}_2$ of the original dataset, (ii) the bootstrapped standard errors, and (iii) the bootstrap 95% confidence intervals. Use the traditional bootstrap intervals with $B = 1000$ boot iterations. The table should look similar to the following output:

Parameter	Estimate	Boot SE	95% Boot L-Bound	95% Boot U-Bound
Beta1	#	#	#	#
Beta2	#	#	#	#
Beta1_Beta2	#	#	#	#

Solution

```
## Code goes here ------
```

Problem 8

Briefly interpret your results. More specifically, check if zero falls in the bootstrap intervals and conclude if we do or do not show statistical significance.