ST203: R for Data Science and Statistics

Rafael Moral

Assignment 4 – 2021

- Do all questions: only one randomly chosen question will be marked.
- Upload your script file via Moodle before 23:59 on Friday 10 December.
- You may include your code and your *commented* answers in the same script file.
- You may submit either an R script (‘.R‘) or an R Markdown file (‘.Rmd‘).
- Place your name and student number on the first line of your R script or in the YAML header in your R Markdown file.

Question 1

a) Construct the matrix.

$$A = \begin{bmatrix} 1 & 2 & 4 & 8 \\ 1 & 1 & 1 & 1 \\ 1 & 3 & 9 & 27 \\ 1 & 7 & 49 & 343 \\ 1 & 6 & 36 & 216 \end{bmatrix} = \begin{bmatrix} 1 & 2 & 2^2 & 2^3 \\ 1 & 1 & 1^2 & 1^3 \\ 1 & 3 & 3^2 & 3^3 \\ 1 & 7 & 7^2 & 7^3 \\ 1 & 6 & 6^2 & 6^3 \end{bmatrix}$$

b) Write a function `powermat` which when given a vector x and a positive integer m (≥ 2) constructs a matrix where the first column is a vector of ones, the second column is x, the third column is x^2 and so on until the m-th column which is x^{m-1}.

c) Test the function with $x = (1,2,3,4,0)$ and $m = 5$.

Question 2

Create a vector `y` containing a total of $n = 20$ observations. The first 10 observations should be simulated from a normal distribution with mean 20 and standard deviation 3, and the remaining 10 observations from a normal distribution with mean 10 and standard deviation 3. Begin by assigning `gl(2, 10)` to the variable `x` and create the design matrix $\mathbf{X}$ using

```
X <- model.matrix(~ x)
```

a) Assign to object `beta.hat` the results of

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}$$

and compare to the results of

```
coef(lm(y ~ x))
```

b) Here, we have 18 residual degrees of freedom (d.f.). Obtain the residual mean square, via

$$\hat{\sigma}^2 = \frac{\mathbf{y}^\top \mathbf{y} - \mathbf{y}^\top \mathbf{X} \hat{\boldsymbol{\beta}} - \hat{\boldsymbol{\beta}}^\top \mathbf{X}^\top \mathbf{y} + \hat{\boldsymbol{\beta}}^\top \mathbf{X}^\top \mathbf{X} \boldsymbol{\beta}}{\text{residual d.f.}}$$

and compare it with

```
anova(lm(y ~ x))$"Mean Sq"[2]
```

c) Obtain the standard errors for the estimates $\hat{\beta}$, given by the square root of the diagonal elements of the matrix

$$\mathbf{H} = (\mathbf{X}^\top \mathbf{X})^{-1} \hat{\sigma}^2,$$

and compare the result with

```
summary(lm(y ~ x))$coef[,2]
```

Question 3

For both code chunks below, rewrite them using pipes and check that you get the same answer:

a)

```
dat <- c(3,5,7,9,11,13,15)
median(exp((log(dat)^2)/5)*0.4)
```

```
## [1] 1.050502
```

b)

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
```

```
## v ggplot2 3.3.1     v purrr   0.3.4
## v tibble  3.0.4     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
glimpse(mpg)
```

```
## Rows: 234
## Columns: 11
## $ manufacturer <chr> "audi", "audi", "audi", "audi", "audi", "audi", "audi"...
## $ model        <chr> "a4", "a4", "a4", "a4", "a4", "a4", "a4", "a4 quattro"...
## $ displ        <dbl> 1.8, 1.8, 2.0, 2.0, 2.8, 2.8, 3.1, 1.8, 1.8, 2.0, 2.0,...
## $ year         <int> 1999, 1999, 2008, 2008, 1999, 1999, 2008, 1999, 1999, ...
## $ cyl          <int> 4, 4, 4, 4, 6, 6, 6, 4, 4, 4, 4, 6, 6, 6, 6, 6, 6, 8, ...
## $ trans        <chr> "auto(l5)", "manual(m5)", "manual(m6)", "auto(av)", "a...
## $ drv          <chr> "f", "f", "f", "f", "f", "f", "f", "4", "4", "4", "4",...
## $ cty          <int> 18, 21, 20, 21, 16, 18, 18, 18, 16, 20, 19, 15, 17, 17...
## $ hwy          <int> 29, 29, 31, 30, 26, 26, 27, 26, 25, 28, 27, 25, 25, 25...
## $ fl           <chr> "p", "p", "p", "p", "p", "p", "p", "p", "p", "p", "p",...
## $ class        <chr> "compact", "compact", "compact", "compact", "compact",...
```

```
## Find the % of cars in the mtcars dataset where mpg is greater than 20
sum(mpg$cty > 20)/length(mpg$cty)
```

```
## [1] 0.1923077
```

Question 4

Construct a vector `x` of $n = 100$ values from the Normal distribution where $\mu = 3$ and $\sigma = 5$.

a) Construct a numeric vector `g` which consists of five 1s, followed by five 2s and so on up to five 20s (hint: use `rep`). Then use `tapply` to construct a vector `y` of length 20, where `y[1]` is the mean of the first 5 values in `x`, `y[2]` is the mean of the next 5 values in `x`, and so on.

b) Use a `for` loop to give an alternative construction for the vector `y`.

c) Plot y_i versus `index <- seq(2.5, 97.5, length = 20)` as points, and join the dots (use `ylim=range(x)*1.1`, `type="o"`, and the `"red"` colour). Superpose the x_i values using `points`, also use `cex = .8, pch = 21, bg = "lightgray"` as arguments.

d) Assign the upper control limit, `ul`, to $\mu + 1.96\sigma$ and the lower control limit, `ll`, to $\mu - 1.96\sigma$ and draw horizontal lines at the positions μ, `ul` and `ll`, with `lty = 2` (dashed lines).

e) Write code which finds the values in `x` outside the control limits.

f) Write code which finds the indices of the values in `x` outside the control limits (hint: use `which`) and paint them blue in the plot (hint: you can use `points` with arguments `pch = 21, bg = "blue"`).

g) Write a function called `test0` which when given a set of observations and the lower and upper control limits, returns the indices for the observations which are outside the control limits.

h) Use the function `test0` to find the indices for values in `x` outside the control limits.

i) Write a function `jump` which finds the biggest (absolute) difference between consecutive values in a sequence of numbers. (Hint: use `diff`, `max` and `abs`. Do not use a `for` loop). Test the function on the vector `x`.

Question 5

The `flights` data set seen in class is available in the package `nycflights13` as a tibble. It contains on-time data for all flights that departed New York City via its three main airports throughout the whole year of 2013. The airport from which each flight departed is recorded in the origin column.

a) Load the `tidyverse` suite of packages, as well as the `nycflights13` package and the `flights` data therein.

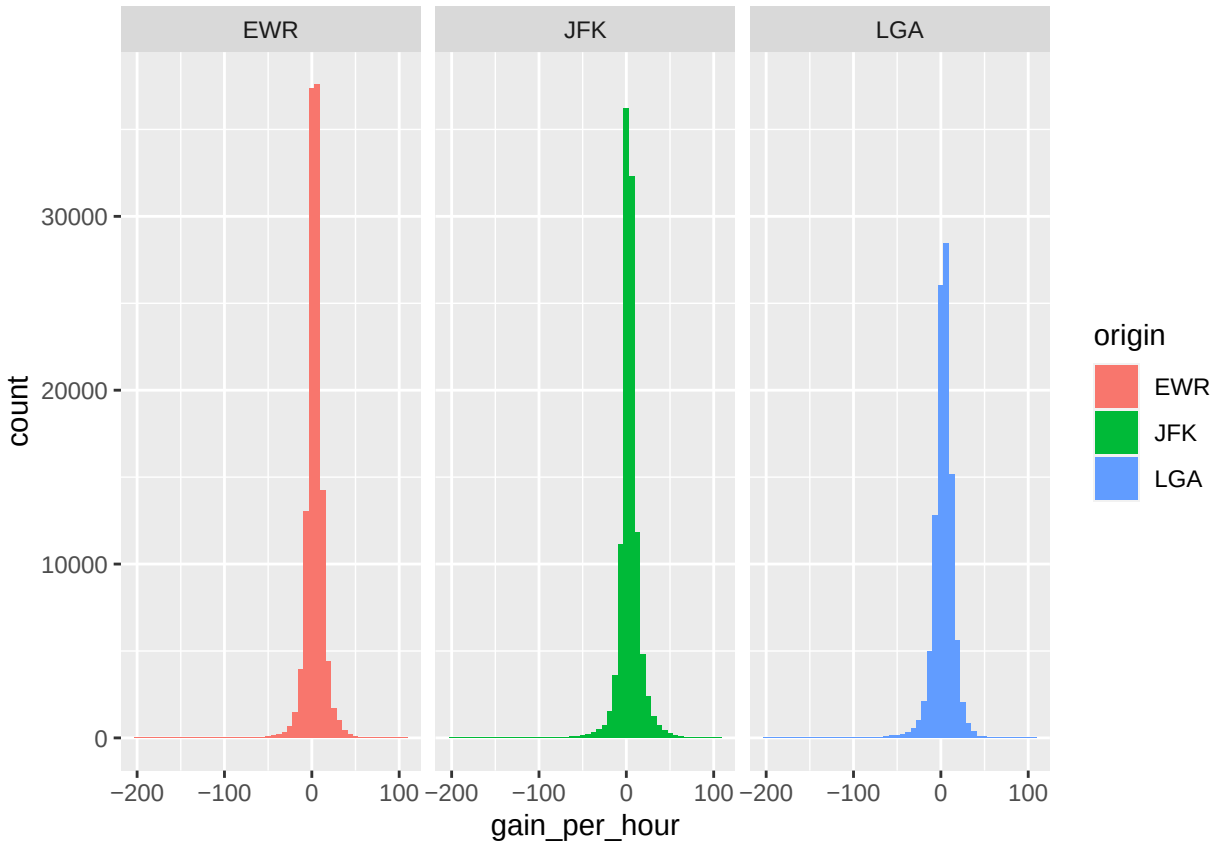
b) Cancelled flights are defined as those for which no departure or arrival took place. Write code using pipe operators and `filter()` to create a new data set `flights2` which removes any rows with missing values (i.e. `NA` entries) for the `dep_delay` or `arr_delay` variables (Hint: see `?is.na` and use the "and" operator `&` and the negation operator `!`).

c) Define the new variable `gain` as the difference between the departure delay and the arrival delay, the new variable `air_hour` as the `air_time` variable re-expressed in hours, and the variable `gain_per_hour` as the `gain` per `air_hour`. Write code using pipe operators and the functions `mutate()` and `select()` which creates a new tibble called `flights3` containing only the `gain_per_hour` and `origin` variables. (Hint: you may exploit the fact that `mutate()` allows you to refer to variables you have just created).

d) The `ggplot2` code below plots a kernel density estimate of the `gain_per_hour` for each origin airport. However, it overplots all three estimated density curves on the one graph. Modify this code so that it (i) partitions/facets the graph into a separate panel for each `origin` and (ii) produces histograms with 50 bins rather than density plots.

```
ggplot(flights3, aes(x = gain_per_hour, fill = origin)) +
  geom_density()
```

Your final plot should look like this:



NOTE: for parts (b) and (c) you may refer to the base R code below, which ultimately produces the same output `flights3`. This code is provided for you to check your results only, do not use this code as a solution to parts (b) and (c).

```
flights2 <- flights[!is.na(flights$dep_delay) & !is.na(flights$arr_delay),]
flights2$gain <- flights2$dep_delay - flights2$arr_delay
flights2$hours <- flights2$air_time / 60
flights2$gain_per_hour <- flights2$gain / flights2$hours
flights3 <- flights2[,c("gain_per_hour", "origin")]
```