HW2.Zunqiu.Wang

Zunqiu Wang

9/12/2021

Q1

```r
## a
sum = 0
func.mean <- function(x) {
    ## a loop for a vector to calculate mean
    for (i in 1:length(x)) {
        sum = sum + x[i]
        mean = sum/length(x)
    }
    return(mean)
}
func.mean(c(5, 9, 11, 12))  # testing
```

```
## [1] 9.25
```

```r
## b nested conditional statement inside a loop and return
## as instructed
func.logic <- function(x) {
    for (i in 1:length(x)) {
        if (is.numeric(x[i]) == TRUE) {
            if (sum(x[1] + x[2]) > sum(x[3] + x[4])) {
                return(x)
            } else {
                return(0)
            }
        } else {
            return("Not numerical input")

        }

    }

}
func.logic(c(0, 1, "f", 4))
```

```
## [1] "Not numerical input"
```

```
func.logic(c(3, 4, 1, 2))
```

```
## [1] 3 4 1 2
```

```
func.logic(c(1, 2, 3, 4))
```

```
## [1] 0
```

```
## c
seq = c(1, 1)
fib <- function(n) {
    # construct a Fibonacci seq using loop
    for (i in 3:n) {
        seq[i] = seq[i - 2] + seq[i - 1]
    }
    return(seq)
}
fib(5)
```

```
## [1] 1 1 2 3 5
```

```
fib(10)
```

```
##  [1]  1  1  2  3  5  8 13 21 34 55
```

```
## d
mtx <- matrix(1:16, nrow = 4, ncol = 4)
mtx.row.mean <- apply(mtx, 1, func.mean)  # apply the function defined in a to calculate row mean
```

Q2

```
## a using aggregate() to calculate max for specified
## column group by month
agg.df <- aggregate(cbind(Wind, Ozone) ~ Month, airquality, max)

library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
group_by(airquality, Month) %>%
    summarise(max(Wind), max(Ozone, na.rm = TRUE))  # equivalent
```

```
## # A tibble: 5 x 3
##   Month `max(Wind)` `max(Ozone, na.rm = TRUE)`
##   <int>       <dbl>                      <int>
## 1     5        20.1                        115
## 2     6        20.7                         71
## 3     7        14.9                        135
## 4     8        15.5                        168
## 5     9        16.6                         96
```

```
## b
authors <- data.frame(surname = c("Tukey", "Venables", "Tierney",
    "Ripley", "McNeil"), nationality = c("US", "Australia", "US",
    "UK", "Australia"), stringsAsFactors = FALSE)
books <- data.frame(name = c("Tukey", "Venables", "Tierney",
    "Ripley", "Ripley", "McNeil", "R Core"), title = c("Exploratory Data Analysis",
    "Modern Applied Statistics ...", "LISP-STAT", "Spatial Statistics",
    "Stochastic Simulation", "Interactive Data Analysis", "An Introduction to R"),
    stringsAsFactors = FALSE)
authors
```

```
##    surname nationality
## 1    Tukey          US
## 2 Venables   Australia
## 3  Tierney          US
## 4   Ripley          UK
## 5   McNeil   Australia
```

```
books
```

```
##       name                         title
## 1    Tukey     Exploratory Data Analysis
## 2 Venables Modern Applied Statistics ...
## 3  Tierney                     LISP-STAT
## 4   Ripley            Spatial Statistics
## 5   Ripley         Stochastic Simulation
## 6   McNeil     Interactive Data Analysis
## 7   R Core          An Introduction to R
```

```
# have to include all rows even if one surname is missing
# in books df
merge(books, authors, by.x = "name", by.y = "surname", all.x = TRUE)
```

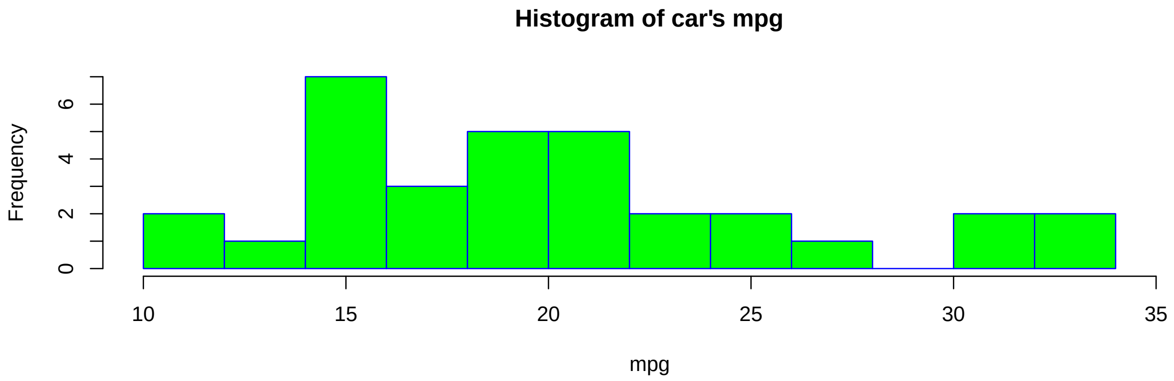
```
##       name                         title nationality
## 1   McNeil     Interactive Data Analysis   Australia
## 2   R Core          An Introduction to R        <NA>
## 3   Ripley            Spatial Statistics          UK
## 4   Ripley         Stochastic Simulation          UK
## 5  Tierney                     LISP-STAT          US
## 6    Tukey     Exploratory Data Analysis          US
## 7 Venables Modern Applied Statistics ...   Australia
## this code chunk adapted from:
## https://github.com/yihui/knitr-examples/blob/master/077-wrap-output.Rmd
## in order to produe a pdf without running out of page
library(knitr)
hook_output = knit_hooks$get("output")
knit_hooks$set(output = function(x, options) {
    # this hook is used only when the linewidth option is
    # not NULL
    if (!is.null(n <- options$linewidth)) {
        x = knitr:::split_lines(x)
        # any lines wider than n should be wrapped
        if (any(nchar(x) > n))
            x = strwrap(x, width = n)
        x = paste(x, collapse = "\n")
    }
    hook_output(x, options)
})
```

```
## c
options(width = 60)
gsub("[t|T]o", "2", "To be, or not to be -- that is the question:Whether
     'tis nobler in the mind to suffer The slings and arrows of outrageous
     fortune, Or to take arms against a sea of troubles, And by opposing end
     them. To die -- to sleep -- No more...")  # replace to or To to 2
```

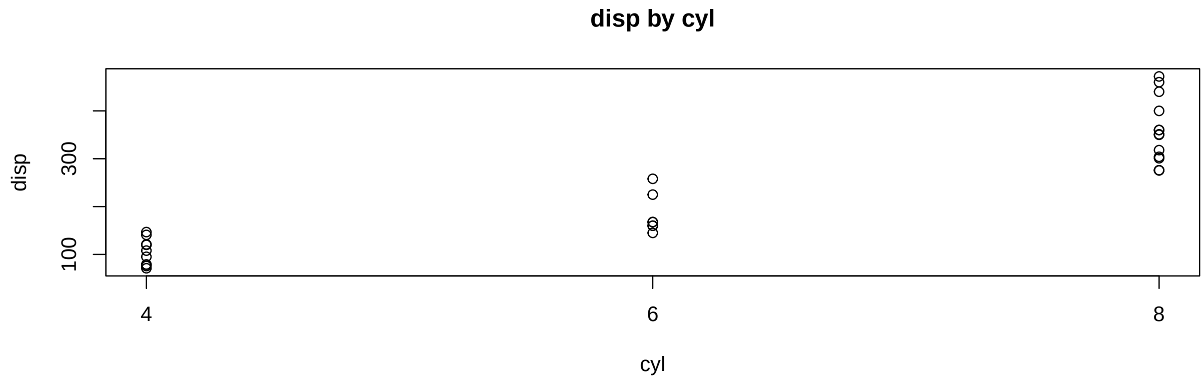
```
## [1] "2 be, or not 2 be -- that is the question:Whether
\n 'tis nobler in the mind 2 suffer The slings and arrows
of outrageous \n fortune, Or 2 take arms against a sea of
troubles, And by opposing end \n them. 2 die -- 2 sleep --
No more..."
```

Q3

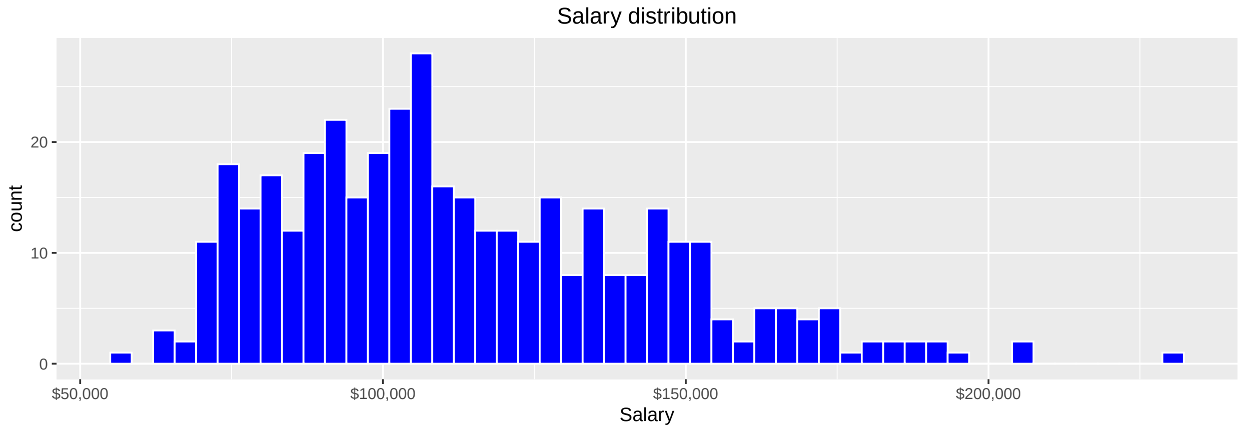
```
## a histogram
hist(mtcars$mpg, main = "Histogram of car's mpg", xlab = "mpg",
    border = "blue", col = "green", xlim = c(10, 35), breaks = 10)
```



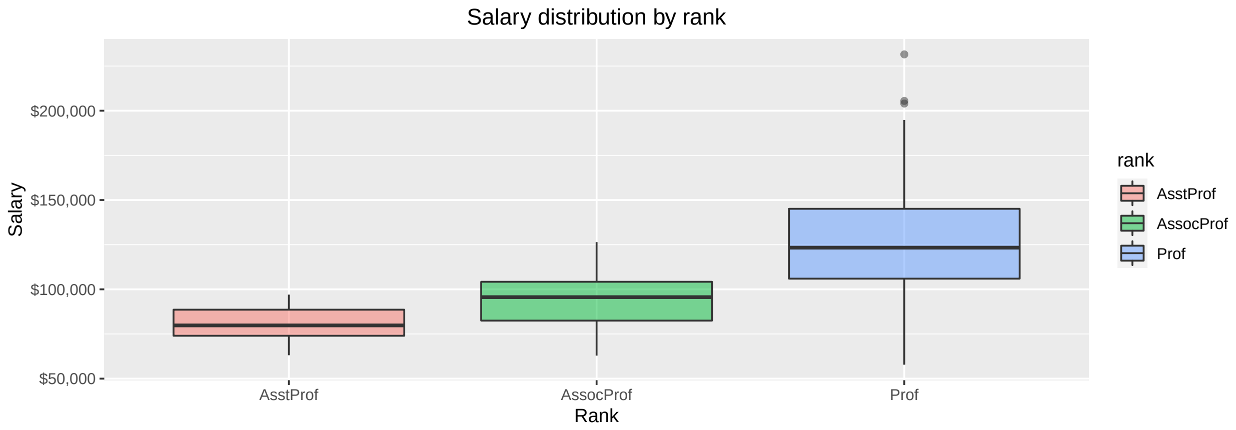
```r
## b scatter plot
plot(mtcars$cyl, mtcars$disp, main = "disp by cyl", xlab = "cyl",
    ylab = "disp", xaxt = "n")
axis(1, xaxp = c(4, 8, 2))

## c ggolot histogram
library(ggplot2)
```



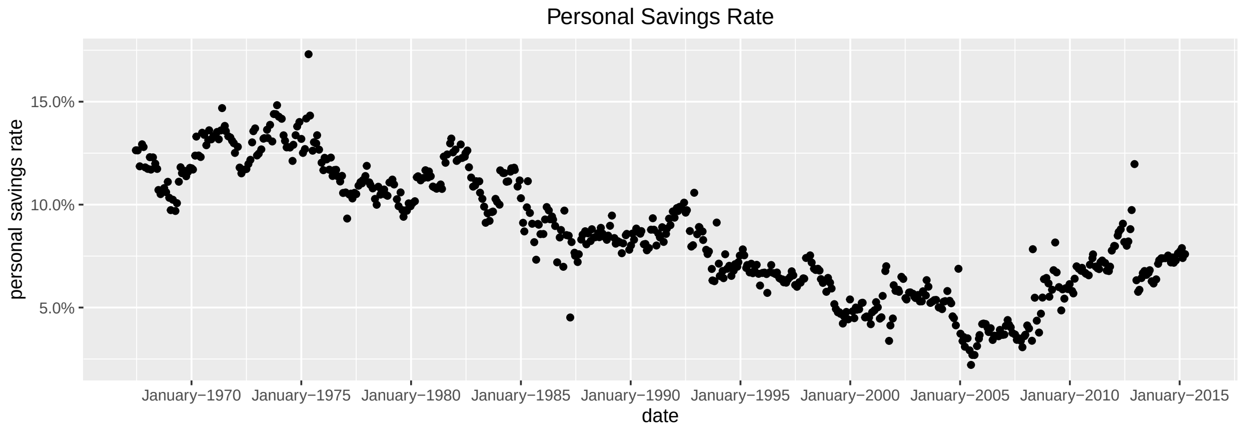
```r
# install.packages('carData')
data(Salaries, package = "carData")
# Salaries
ggplot(Salaries, aes(x = salary)) + geom_histogram(bins = 50,
    fill = "blue", color = "white") + labs(title = "Salary distribution",
    x = "Salary") + scale_x_continuous(label = scales::dollar) +
    theme(plot.title = element_text(hjust = 0.5))
```



```r
## d ggplot boxplot
ggplot(Salaries, aes(x = as.factor(rank), y = salary, fill = rank)) +
    geom_boxplot(alpha = 0.5) + labs(title = "Salary distribution by rank",
    x = "Rank", y = "Salary") + scale_y_continuous(label = scales::dollar) +
    theme(plot.title = element_text(hjust = 0.5))
```



```
## e install.packages('economics') economics
library(scales)
ggplot(economics, aes(x = date, y = psavert/100)) + geom_jitter() +
    labs(title = "Personal Savings Rate", x = "date", y = "personal savings rate") +
    scale_y_continuous(label = scales::percent) + scale_x_date(date_breaks = "5 year",
    labels = date_format("%B-%Y")) + theme(plot.title = element_text(hjust = 0.5))
```


Personal Savings Rate
personal savings rate
January−1970
January−1975
January−1980
January−1985
January−1990
January−1995
January−2000
January−2005
January−2010
January−2015
date