Coding Assignment 1

Team N

Due: 2021-09-29 23:59

Table of Contents

# Put any packages you want here  
library(gt)  
library(tidyverse)

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.2 ──  
## ✔ ggplot2 3.3.6 ✔ purrr 0.3.4  
## ✔ tibble 3.1.8 ✔ dplyr 1.0.9  
## ✔ tidyr 1.2.0 ✔ stringr 1.4.1  
## ✔ readr 2.1.2 ✔ forcats 0.5.2  
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()

library(gtsummary)  
library(plotly)

##   
## Attaching package: 'plotly'  
##   
## The following object is masked from 'package:ggplot2':  
##   
## last\_plot  
##   
## The following object is masked from 'package:stats':  
##   
## filter  
##   
## The following object is masked from 'package:graphics':  
##   
## layout

library(readxl)  
library(plotly)  
library(corrplot)

## corrplot 0.92 loaded

sessionInfo()

## R version 4.2.1 (2022-06-23 ucrt)  
## Platform: x86\_64-w64-mingw32/x64 (64-bit)  
## Running under: Windows 10 x64 (build 19044)  
##   
## Matrix products: default  
##   
## locale:  
## [1] LC\_COLLATE=English\_United States.utf8   
## [2] LC\_CTYPE=English\_United States.utf8   
## [3] LC\_MONETARY=English\_United States.utf8  
## [4] LC\_NUMERIC=C   
## [5] LC\_TIME=English\_United States.utf8   
##   
## attached base packages:  
## [1] stats graphics grDevices utils datasets methods base   
##   
## other attached packages:  
## [1] corrplot\_0.92 readxl\_1.4.1 plotly\_4.10.0 gtsummary\_1.6.1  
## [5] forcats\_0.5.2 stringr\_1.4.1 dplyr\_1.0.9 purrr\_0.3.4   
## [9] readr\_2.1.2 tidyr\_1.2.0 tibble\_3.1.8 ggplot2\_3.3.6   
## [13] tidyverse\_1.3.2 gt\_0.7.0   
##   
## loaded via a namespace (and not attached):  
## [1] lubridate\_1.8.0 assertthat\_0.2.1 digest\_0.6.29   
## [4] utf8\_1.2.2 R6\_2.5.1 cellranger\_1.1.0   
## [7] backports\_1.4.1 reprex\_2.0.2 evaluate\_0.16   
## [10] httr\_1.4.4 pillar\_1.8.1 rlang\_1.0.4   
## [13] lazyeval\_0.2.2 googlesheets4\_1.0.1 rstudioapi\_0.14   
## [16] data.table\_1.14.2 rmarkdown\_2.16 googledrive\_2.0.0   
## [19] htmlwidgets\_1.5.4 munsell\_0.5.0 broom\_1.0.1   
## [22] compiler\_4.2.1 modelr\_0.1.9 xfun\_0.32   
## [25] pkgconfig\_2.0.3 htmltools\_0.5.3 tidyselect\_1.1.2   
## [28] viridisLite\_0.4.1 fansi\_1.0.3 crayon\_1.5.1   
## [31] tzdb\_0.3.0 dbplyr\_2.2.1 withr\_2.5.0   
## [34] grid\_4.2.1 jsonlite\_1.8.0 gtable\_0.3.0   
## [37] lifecycle\_1.0.1 DBI\_1.1.3 magrittr\_2.0.3   
## [40] scales\_1.2.1 cli\_3.3.0 stringi\_1.7.8   
## [43] broom.helpers\_1.8.0 fs\_1.5.2 xml2\_1.3.3   
## [46] ellipsis\_0.3.2 generics\_0.1.3 vctrs\_0.4.1   
## [49] tools\_4.2.1 glue\_1.6.2 hms\_1.1.2   
## [52] fastmap\_1.1.0 yaml\_2.3.5 colorspace\_2.0-3   
## [55] gargle\_1.2.0 rvest\_1.0.3 knitr\_1.40   
## [58] haven\_2.5.1

A Florida health insurance company wants to predict annual claims for individual clients. The company pulls a random sample of 50 customers. The owner wishes to charge an actuarially fair premium to ensure a normal rate of return. The owner collects all of their current customer’s health care expenses from the last year and compares them with what is known about each customer’s plan.

The data on the 50 customers in the sample is as follows:

* Charges: Total medical expenses for a particular insurance plan (in dollars)
* Age: Age of the primary beneficiary
* BMI: Primary beneficiary’s body mass index (kg/m2)
* Female: Primary beneficiary’s birth sex (0 = Male, 1 = Female)
* Children: Number of children covered by health insurance plan (includes other dependents as well)
* Smoker: Indicator if primary beneficiary is a smoker (0 = non-smoker, 1 = smoker)
* Cities: Dummy variables for each city with the default being Sanford

Answer the following questions using complete sentences and attach all output, plots, etc. within this report.

**For this assignment, ignore the categorical variables (gender, smoker, cities)**

# Question 1

Perform univariate analyses on the quantitative variables (center, shape, spread). Include descriptive statistics, and histograms. Be sure to use terms discussed in class such as bimodal, skewed left, etc.

# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* CHARGES \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
# Adding the Charges column to the "charges" variable  
Charges <- (insurancegroup7$Charges)  
# This generates the mean of the Charges column using the "charges" variable  
mean(Charges)

## [1] 15132.46

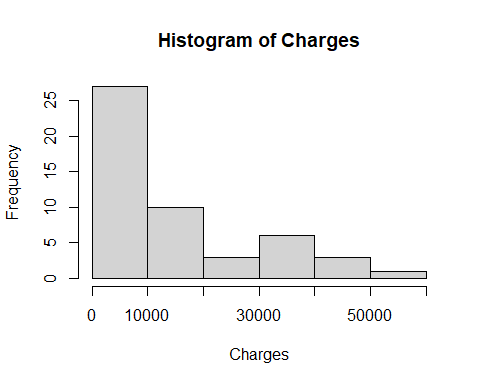
# This generates the median of the Charges column using the "charges" variable  
median(Charges)

## [1] 9819.649

# Custom function created by professor to generate the mode using the "charges" variable  
getModes <- function(x) {  
 ux <- unique(x)  
 tab <- tabulate(match(x, ux))  
 ux[tab == max(tab)]  
}  
  
# Using the custom function from the professor to get the mode of the Charges column using the "charges" variable  
getModes(Charges)

## [1] 8965.796 2850.684 38511.628 9850.432 37270.151 47055.532 27218.437  
## [8] 1146.797 2523.169 39774.276 6555.070 5261.469 11658.115 4466.621  
## [15] 41661.602 19144.577 9880.068 7935.291 8930.935 14001.287 34828.654  
## [22] 17128.426 2689.495 3277.161 18903.491 9788.866 6548.195 12957.118  
## [29] 4058.116 4402.233 12347.172 6059.173 12333.828 12323.936 58571.074  
## [36] 1261.859 36189.102 8671.191 1837.237 1149.396 2494.022 9144.565  
## [43] 22412.648 2138.071 2719.280 40273.645 20781.489 34303.167 10231.500  
## [50] 2136.882

# Generate a histogram of the Charges column using the "charges" variable  
hist(Charges)



# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Age \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
Age <- (insurancegroup7$Age)  
  
mean(Age)

## [1] 37.56

median(Age)

## [1] 36

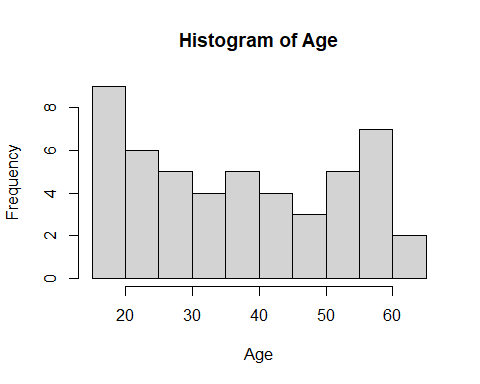
getModes <- function(x) {  
 ux <- unique(x)  
 tab <- tabulate(match(x, ux))  
 ux[tab == max(tab)]  
}  
  
  
getModes(Age)

## [1] 19

summary(Age)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 18.00 24.00 36.00 37.56 51.75 63.00

hist(Age)



# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* BMI \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
BMI <- (insurancegroup7$BMI)  
  
mean(BMI)

## [1] 32.3258

median(BMI)

## [1] 32.11

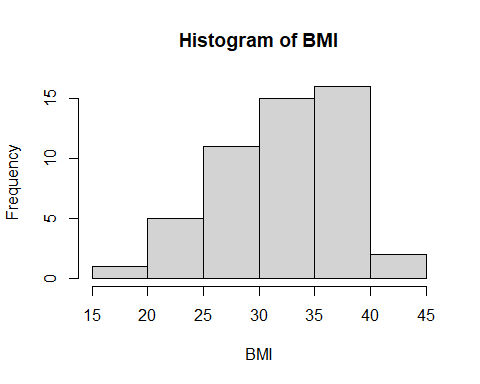
getModes <- function(x) {  
 ux <- unique(x)  
 tab <- tabulate(match(x, ux))  
 ux[tab == max(tab)]  
}  
  
  
getModes(BMI)

## [1] 36.670 28.595 37.100 36.575

summary(BMI)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 19.95 28.69 32.11 32.33 36.65 43.01

hist(BMI)



# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Children \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
Children <- (insurancegroup7$Children)  
  
mean(Children)

## [1] 1.06

median(Children)

## [1] 1

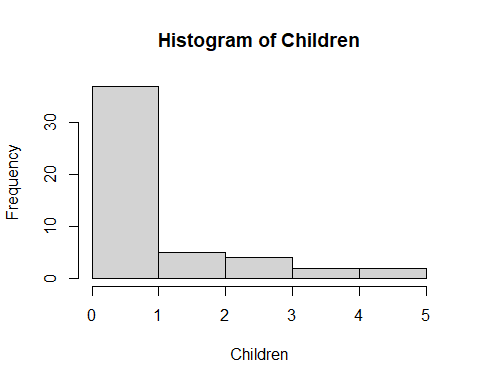
getModes <- function(x) {  
 ux <- unique(x)  
 tab <- tabulate(match(x, ux))  
 ux[tab == max(tab)]  
}  
  
  
getModes(Children)

## [1] 0

summary(Children)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.00 0.00 1.00 1.06 1.75 5.00

hist(Children)



# Question 2

Perform bivariate analyses on the quantitative variables (direction, strength and form). Describe the linear association between all variables.

# Question 3

Generate a regression equation in the following form:

also write out the regression cleanly in this document.

# Question 4

An eager insurance representative comes back with a potential client. The client is 40, their BMI is 30, and they have one dependent. Using the regression equation above, predict the amount of medical expenses associated with this policy. (Provide a 95% confidence interval as well)