Homework 2

Derien Weatherspoon

2023-02-01

```r
cars_multi <- read.csv("cars_multi.csv")
library(ggplot2)
library(tidyverse)
```

```
## ── Attaching packages ─────────────────────────────────── tidyverse 1.3.1 ──
```
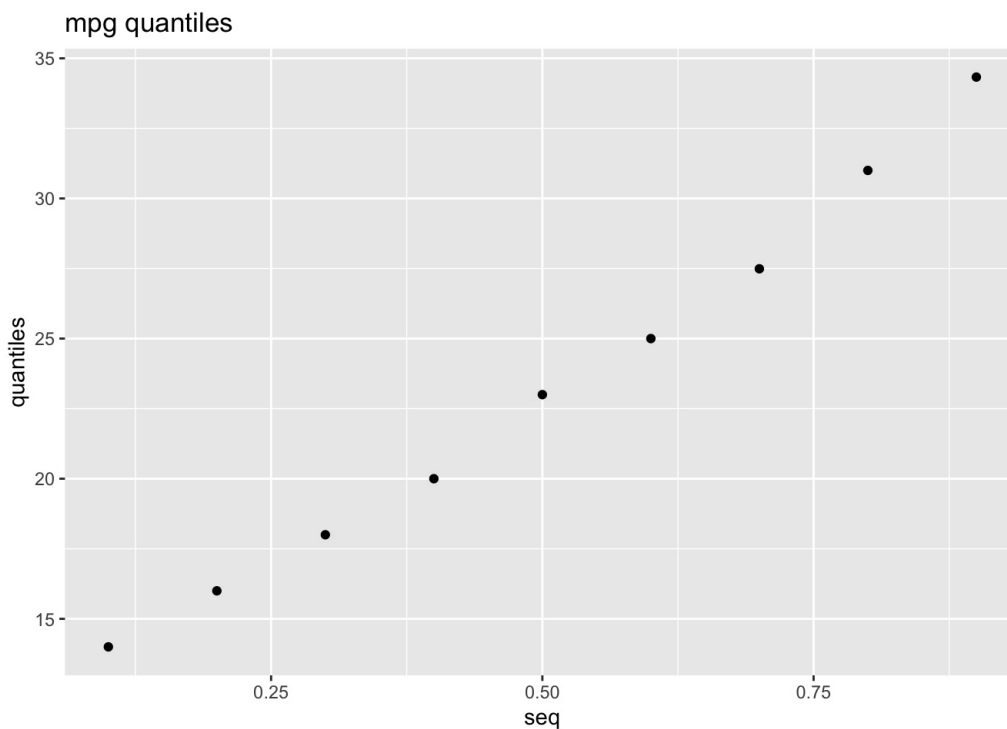
```
## ✔ tibble  3.1.3      ✔ dplyr   1.0.7
## ✔ tidyr   1.1.3      ✔ stringr 1.4.0
## ✔ readr   2.0.1      ✔ forcats 0.5.1
## ✔ purrr   0.3.4
```

```
## ── Conflicts ──────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
```

```r
#Quantile plotting. Recall that you can find quantiles with the quantile command. Take the values 0.1, 0.2, ...,
0.9, and find the corresponding quantiles for the mpg variable in the cars_multi dataset, and create a scatterplo
t with the quantiles on the y axis, and the decimal values 0.1, ..., 0.9 on the x.
seq_x <- seq(0.1,0.9,0.1)
seq_y <- quantile(cars_multi$mpg, probs = seq(0.1,0.9,0.1))
df <- data.frame(seq_x,seq_y)

ggplot(df, aes(seq_x,seq_y)) + geom_point() + labs(x="seq",y="quantiles", title = "mpg quantiles")
```



```r
#Repeat (1), but use a facet command to create separate plots by origin code.

#Check origin country assignments
origin_factored <- cars_multi$origin.factored <- factor(cars_multi$origin, labels = c("usa", "eu", "jp"))
with(cars_multi,table(origin.factored, origin))
```
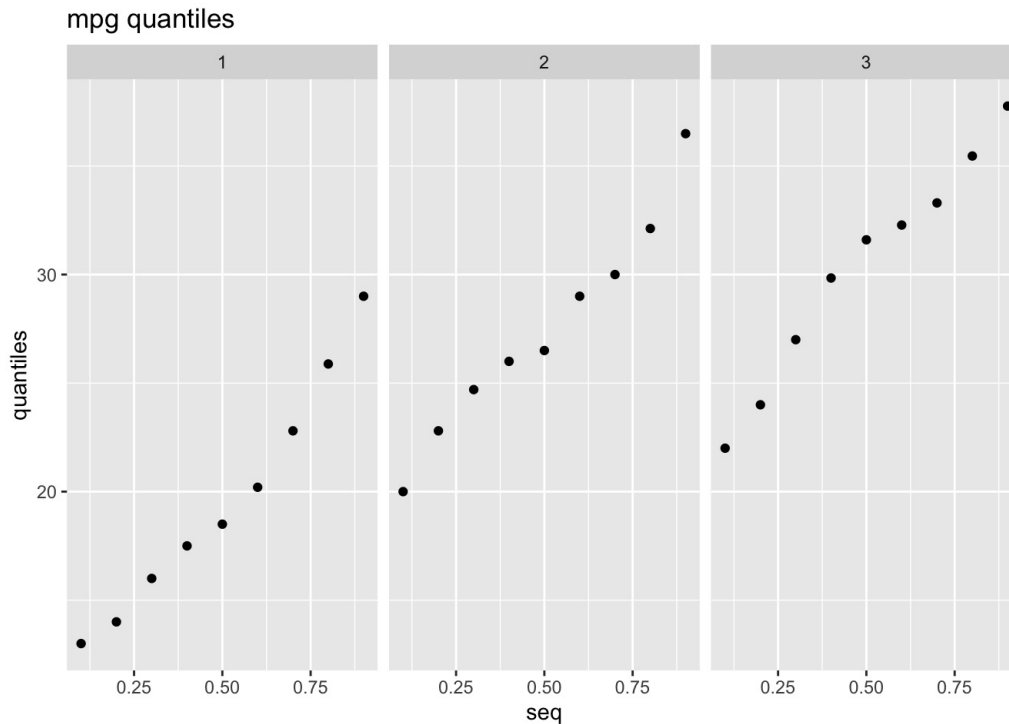
```
##                 origin
## origin.factored   1    2    3
##            usa  249    0    0
##            eu     0   70    0
##            jp     0    0   79
```
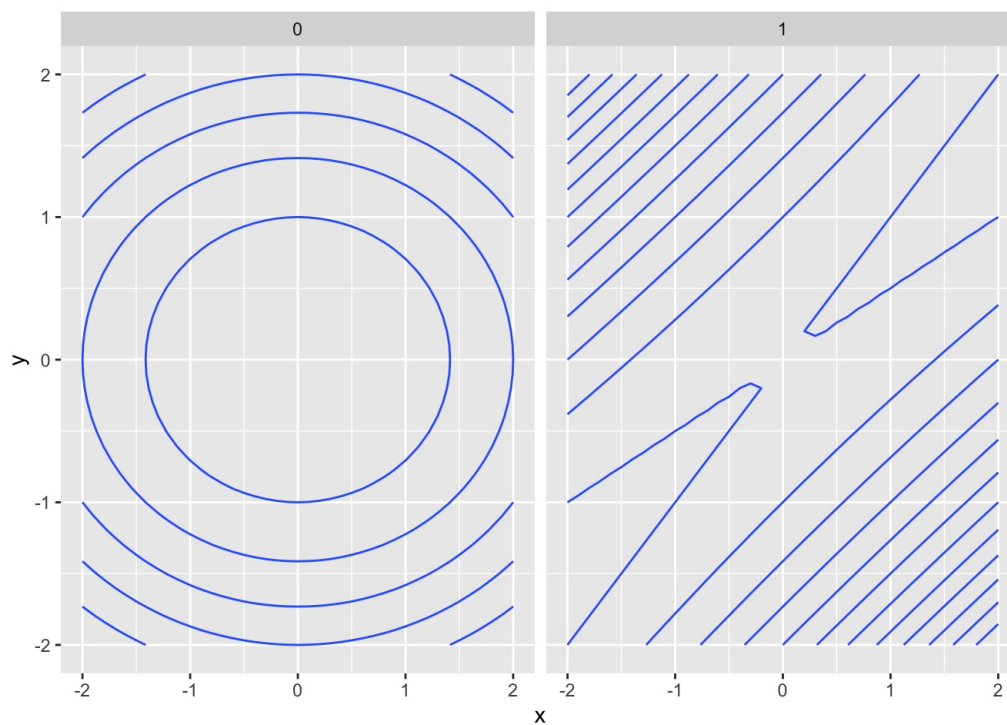
```
seq_y_2 <- list()
for (i in 1:3){
  seq_y_2[[i]] <- quantile(cars_multi$mpg[cars_multi$origin == i], probs = seq_x)
}

dat_2 <- data.frame(x = seq_x, y = seq_y_2[[1]], origin = 1)
for (i in 2:3){
  dat_2 <- rbind(dat_2, data.frame(x = seq_x, y = seq_y_2[[i]], origin = i))
}
ggplot(dat_2, aes(x,y)) + geom_point() + labs(x="seq",y="quantiles", title = "mpg quantiles") + facet_wrap(~as.fa
ctor(origin))
```


mpg quantiles

```
#For the function f(x, y, z) = x2 + 2y2 - 3xyz, z only takes the values 0 and 1. Create a contour plot with ggplo
t that does separate graphs for the 2 values of z (with faceting).

#initialize
x <- seq(-2, 2, by = 0.1)
y <- seq(-2, 2, by = 0.1)
z <- c(0, 1)
df_3 <- expand.grid(x = x, y = y, z= z)
df_3$f <- with(df_3, x^2 + 2*y^2 - 3*x*y*z)
#plotting contour plot now
ggplot(df_3, aes(x = x, y = y, z = f)) + geom_contour() + facet_wrap(~ z) +
labs (x = "x", y = "y", color = "f(x, y, z)")
```

```
#x and y are continuous from 0 to 1.
#don't make matrix
```

```
#A fair die is rolled 30 times.
#a. What is the probability that exactly half of the rolls are even numbers?
#b. What is the probability that more than 20 of the rolls are even numbers?
#c. What is the probability that less than 5 of the rolls are greater than 4?

a4 <- dbinom(15, 30, 0.5) #15 is the number we want, 30 rolls, and 3/6 (0.5) chance to get an even?
a4
```

```
## [1] 0.1444644
```

```
b4 <-  1 - pbinom(20, 30, 0.5) #20 because we want more than 20 success, 30 number of rolls, 0.5 chance to be even
b4
```

```
## [1] 0.02138697
```

```
c4 <- pbinom(4,30,1/3) #Less than 5 rolls are greater than 4(0.33), 30 trials
c4
```

```
## [1] 0.01222972
```

```
#A web server gets typically pinged according to a Poisson process with rate 30/second.
#a. Find the probability that the server gets pinged between 20 and 40 times in a particular second.
#b. Calculate the number of seconds in a year
#c. Use (b) to estimate the maximum number of pings in a single second over the course of a year
#d. Often a web server creates alerts when the ping rate is alarmingly high (typically, the sign of a Denial of Service attack by a hacker). What would be a good rate to create such an alarm (and why)?


a5 <- ppois(40,30) - ppois(20,30)
a5
```

```
## [1] 0.9324058
```

```
b5 <- 60 * 60 * 24 * 365
b5
```

```
## [1] 31536000
```

```
c5 <- max(rpois(b5,30))
c5
```

```
## [1] 65
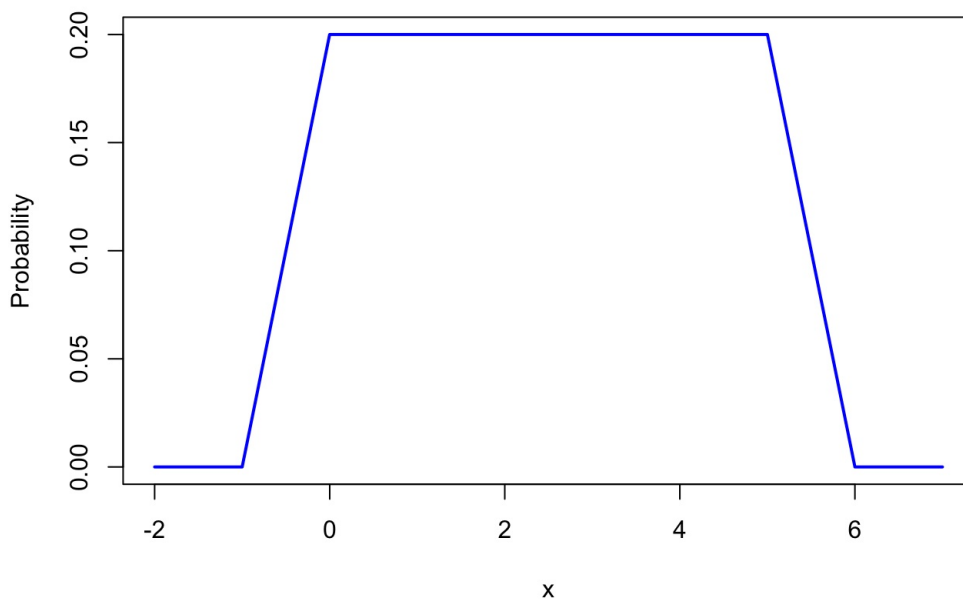```

```
#Ask about c and d
quantile(rpois(b5,30),seq(0.85,1, by =0.01))
```

```
##   85%   86%   87%   88%   89%   90%   91%   92%   93%   94%   95%   96%   97%   98%   99%  100%
##    36    36    36    36    37    37    37    38    38    39    39    40    41    42    43    63
```

```
#Based on the simulations ran, it looks like a ping rate of 65 would be enough to set an alarm because it's way h
igher than the median number of pings.
```

```
#The number of minutes that a bus is late is modeled by the Uniform density on the interval (0,5).
#a. Draw a picture of the density function.
#b. What is the probability that the bus is more than 1 minutes late?
#c. What is the conditional probability that the bus is more than 4 minutes late, given that it is already 3 minu
tes late?
#x <- 0:5 # Grid
#dunif(x, min = 1, max = 4)
x <- seq(-2, 7, length=10)
y <- dunif(x, min = 0, max = 5)
a1 <- plot(x, y, type = 'l', lwd = 2, ylim = c(0, .2), col='blue',
      xlab='x', ylab='Probability', main='Uniform Distribution Plot')
```



```
b6 <- punif(1,0,5)
b6
```

```
## [1] 0.2
```

```
c6 <- dunif(3,0,5)/punif(4,0,5)
c6
```

```
## [1] 0.25
```

```
#Using the normal density with μ = 50 cm and σ = 5 cm as a model for the length of catfish in a lake, answer the
following questions. Draw an appropriate picture of a normal density for each question.
#a. If a catfish is selected at random, what is the probability that it is more than 60 cm in length?
#b. What is the length x such that exactly 10% of catfish are shorter than x?
#c. What is the length y such that exactly 70% of catfish are longer than y?

a7 <- pnorm(60, 50, 5, lower.tail = F)
a7
```

```
## [1] 0.02275013
```

```
b7 <- qnorm(0.1,50,5, lower.tail = T)
b7
```

```
## [1] 43.59224
```

```
c7 <- qnorm(0.7,50,5, lower.tail = F)
c7
```

```
## [1] 47.378
```

```
#8. Let X be a random variable with μ = 80 and σ = 10.
#a. Compute P(μ − σ < X < μ + σ). Note that this is P(70 < X < 90).
#b. Compute P(60 < X < 100). Note that this can be written as P(μ − 2σ < X < μ + 2σ).
#c. Compute P(50 < X < 110). Note that this can be written as P(μ − 3σ < X < μ + 3σ).
#d. The relationships among normal densities suggest that you should get the same answers to these three question
s no matter what the values of μ and σ. Verify that this is true by calculating the same three quantities, but th
is time for a standard normal distribution, i.e., a normal distribution with μ = 0 and σ = 1.
a8 <- pnorm(90,80,10) - pnorm(70,80,10)
a8
```

```
## [1] 0.6826895
```

```
b8 <- pnorm(100,80,10) - pnorm(60,80,10)
b8
```

```
## [1] 0.9544997
```

```
c8 <- pnorm(110,80,10) - pnorm(50,80,10)
c8
```

```
## [1] 0.9973002
```

```
d8_1 <- pnorm(1,0,1) - pnorm(-1,0,1)
d8_2 <- pnorm(2,0,1) - pnorm(-2,0,1)
d8_3 <- pnorm(3,0,1) - pnorm(-3,0,1)
d8_1
```

```
## [1] 0.6826895
```

```
d8_2
```

```
## [1] 0.9544997
```

```
d8_3
```

```
## [1] 0.9973002
```

#https://blog.uvm.edu/tdonovan-vtcfwru/files/2020/07/binomial.pdf (https://blog.uvm.edu/tdonovan-vtcfwru/files/2020/07/binomial.pdf)
#http://stats4stem.weebly.com/r-binomial-distribution.html (http://stats4stem.weebly.com/r-binomial-distribution.html)
#https://cosmosweb.champlain.edu/people/stevens/WebTech/R/Chapter-6-R.pdf
(https://cosmosweb.champlain.edu/people/stevens/WebTech/R/Chapter-6-R.pdf)