S&DS 220: Homework 2

Due Friday January 26th, 11:59 pm

Braeden Cullen

Instructions

Complete the questions below. Upload your knitted PDF solutions to Gradescope by January 26th, 11:59PM.

Question 1: Exercise 1.4

In this exercise, you will graph the function $f(p) = p(1 - p)$ for $p \in [0, 1]$.

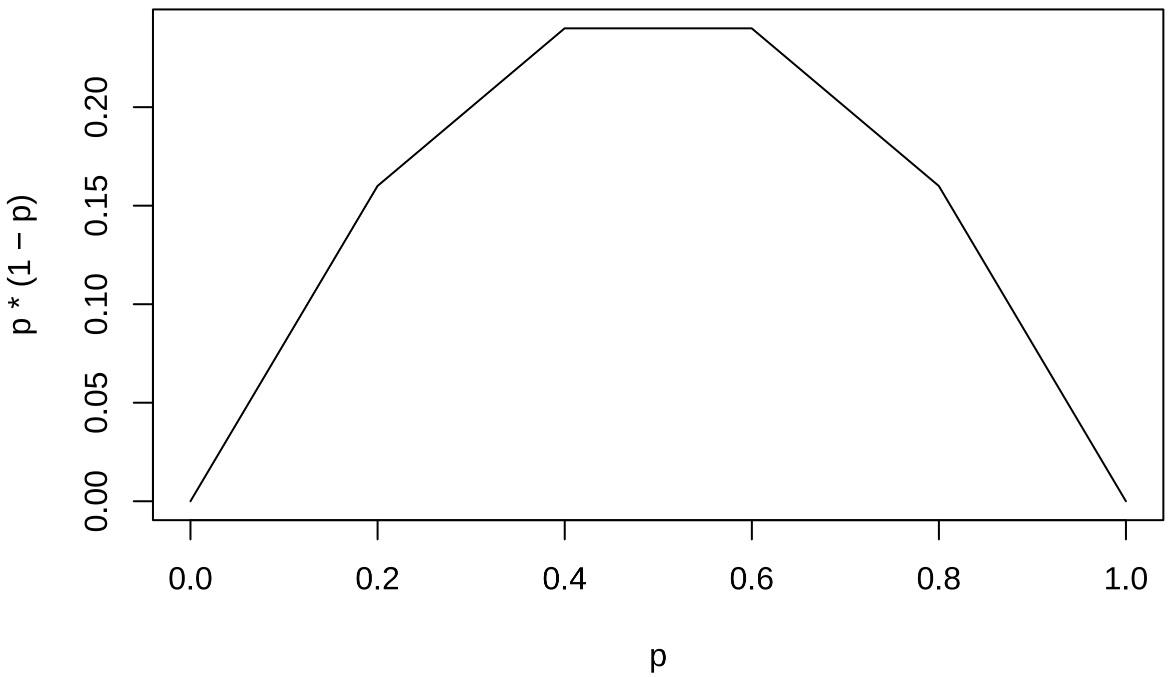
(a) Use seq to create a vector `p` of numbers from 0 to 1 spaced by 0.2.

```r
# your code here
p <- seq(0, 1, by = 0.2)
p
```

```
## [1] 0.0 0.2 0.4 0.6 0.8 1.0
```

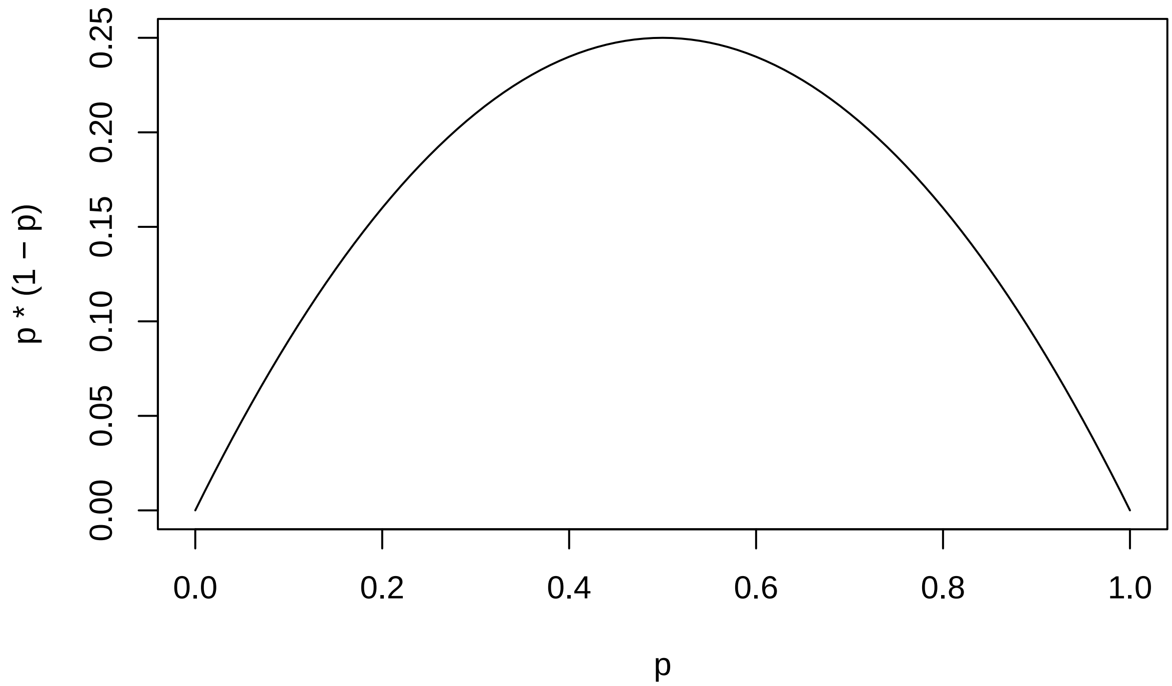
(b) Use plot to plot `p` in the `x` coordinate and `p(1-p)` in the `y` coordinate. Read the help page for `plot` and experiment with the `type` argument to find a good choice for this graph.

```r
# your code here
plot(p, p*(1-p), type = "l")
```



(c) Repeat, but with creating a vector `p` of numbers from 0 to 1 spaced by 0.01.

```
# your code here
p <- seq(0, 1, by = 0.01)
plot(p, p*(1-p), type = "l")
```


0.00
0.05
0.10
0.15
0.20
0.25
p * (1 − p)
0.0
0.2
0.4
0.6
0.8
1.0
p

Question 2: Exercsie 1.7

R has a built-in vector `rivers` which contains the lengths of major North American rivers.

(a) Use `?rivers` to learn about the data set.

```
?rivers
```

(b) Find the mean and standard deviation of the rivers data using the base R functions `mean` and `sd`.

```
# your code here
mn <- mean(rivers)
s <- sd(rivers)
mn
```

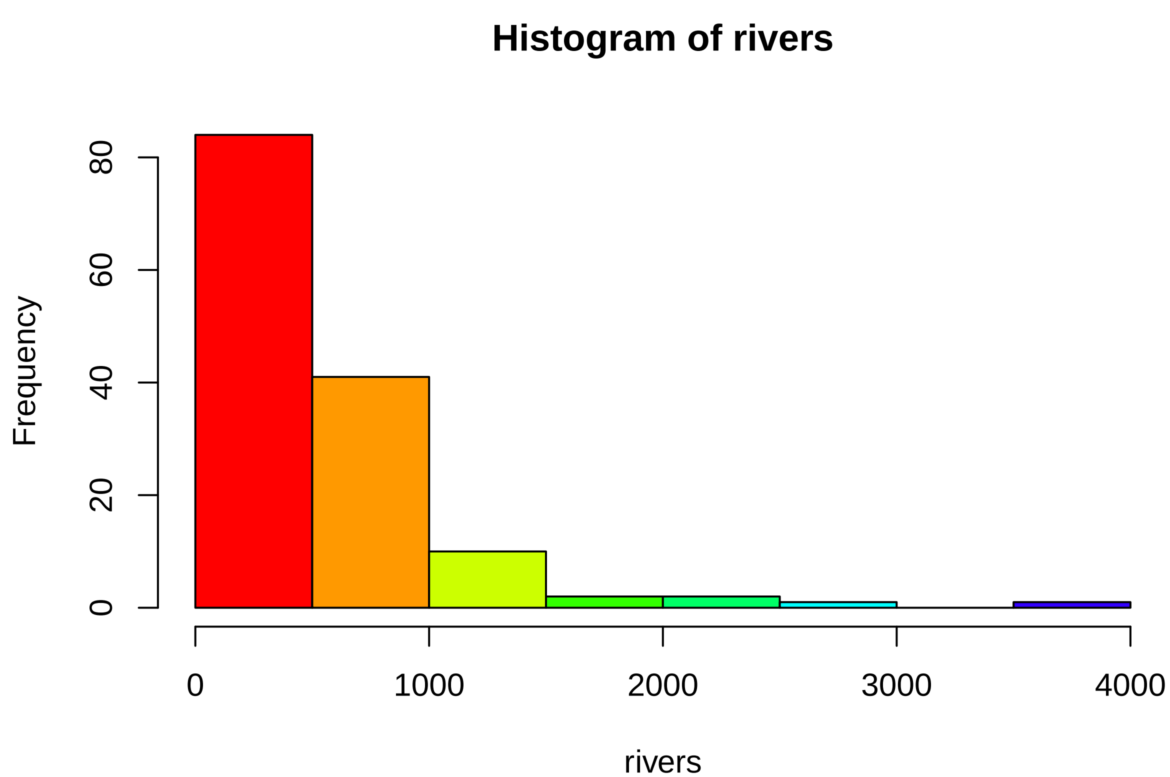
```
## [1] 591.1844
```

```
s
```

```
## [1] 493.8708
```

(c) Make a histogram (`hist`) of the rivers data.

```
# your code here
hist(rivers, col = rainbow(10))
```



(d) Get the five number summary (`summary`) of rivers data.

```
# your code here
summary(rivers)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   135.0   310.0   425.0   591.2   680.0  3710.0
```

(e) Find the longest and shortest lengths of rivers in the set.

```
# your code here
mx <- max(rivers)
mn <- min(rivers)
mx
```

```
## [1] 3710
```

```
mn
```

```
## [1] 135
```

(f) Make a list of all (lengths of) rivers longer than 1000 miles.

```
# your code here
river_lst <- rivers[rivers > 1000]
river_lst
```

```
##  [1] 1459 1450 1243 2348 1171 3710 2315 2533 1306 1054 1270 1885 1100 1205 1038
## [16] 1770
```

Question 3: Exercise 1.9

There is a built-in data set `state`, which is really seven separate variables with names such as

`state.name`, `state.region`, and `state.area`.

(a) What are the possible regions a state can be in? How many states are in each region?

```
# your code here
table(state.region)
```

```
## state.region
##     Northeast         South North Central          West
##             9            16            12            13
```

(b) Which states have area less than 10,000 square miles?

```
# your code here
less_than <- state.name[state.area < 10000]
less_than
```

```
## [1] "Connecticut"   "Delaware"      "Hawaii"        "Massachusetts"
## [5] "New Hampshire" "New Jersey"    "Rhode Island"  "Vermont"
```

(c) Which state's geographic center is furthest south? (Hint: use `which.min`)

```
# your code here
lat = state.center$y
min_index = which.min(lat)
state.name[min_index]
```

```
## [1] "Florida"
```

Question 4: Exercise 1.11

Consider the `mtcars` data set.

(a) Convert the am variable to a factor with two levels, auto and manual, by typing the following:

`mtcars$am <- factor(mtcars$am, levels = c(0, 1), labels = c("auto", "manual")).`

```
# your code here
data(mtcars)
mtcars$am <- factor(mtcars$am, levels = c(0, 1), labels = c("auto", "manual"))
head(mtcars$am)
```

```
## [1] manual manual manual auto   auto   auto
## Levels: auto manual
```

(b) How many cars of each type of transmission are there?

```
# your code here
num_auto <- table(mtcars$am)[1]
num_manual <- table(mtcars$am)[2]
# num_auto
num_auto
```

```
## auto
##   19
```

```
# num_manual
num_manual
```

```
## manual
##     13
```

```
# 19 auto, 13 manual
```

(c) How many cars of each type of transmission have gas mileage estimates greater than 25 mpg?

```
# your code here
num_auto <- mtcars$mpg[mtcars$am == "auto"]
num_auto_greater <- sum(num_auto > 25)
num_auto_greater
```

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

```
num_manual <- mtcars$mpg[mtcars$am == "manual"]
num_manual_greater <- sum(num_manual > 25)
num_manual_greater
```

```
## [1] 6
```

```
# 0 auto, 6 manual
```

Question 5: Exercise 1.12

This problem uses the data set `hot_dogs` from the package `fosdata`. See the section called **Libraries** in the Preface of the text under "Software Installation" (page xii).

> *Important: never install a package in an R script or R Markdown document. Always use the console!*

(a) How many observations of how many variables are there? What types are the variables?

```
# your code here
library(fosdata)
dim(hot_dogs)
```

```
## [1] 54  3
```

```
# 54 observations of 3 variables
str(hot_dogs)
```

```
## 'data.frame':    54 obs. of  3 variables:
##  $ type    : Factor w/ 3 levels "Beef","Meat",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ calories: int  186 181 176 149 184 190 158 139 175 148 ...
##  $ sodium  : int  495 477 425 322 482 587 370 322 479 375 ...
```

```
# the variables are type, calories, and sodium
```

(b) What are the three kinds of hot dogs in this data set?

```
# your code here
table(hot_dogs$type)
```

```
## 
##    Beef    Meat Poultry 
##      20      17      17
```

```
# Beef Meat Poultry
```

(c) What is the highest sodium content of any hot dog in this data set?

```
# your code here
max(hot_dogs$sodium)
```

```
## [1] 645
```

```
# 645
```

(d) What is the mean calorie content for Beef hot dogs?

```r
# your code here
mean(hot_dogs$calories[hot_dogs$type=="Beef"])
```

```
## [1] 156.85
```

```r
# 156.85
```