Gov 2001: Problem Set 2
Assessment Problem Solutions

Instructions

You should submit your answers and R code to the problems below using the Quizzes section on Canvas.
Remember that you should treat this assessment problem as you would treat a final exam. You are not allowed to discuss the problem with classmates, the teaching staff, or any other people. You also may not post questions about the assessment problem to the Canvas discussion boards.
You may consult any readings, notes, or R code from the class, and you can also use the internet as a resource, but remember that all answers and code must be entirely your own.
Also don't forget that you only have one opportunity to submit your answers to the assessment problem on Canvas. Be sure you check your work before you click the submit button.
Remember that the purpose of these assessment problems is for you to get an honest sense of how well you understand the content from the class. If you struggle with an assessment problem one week, you should spend the following week being sure that you catch up on the topics you didn't understand.

Using simulation to decide your Friday night plans

Imagine that you're a member of the United States Senate. The US Capitol building has one bar inside that exclusively serves to US Senators. It's the hip place to be on a Friday night, and all of your distinguished colleagues want to be there. The problem is that the bar is very small inside, and it's very miserable to be there when there's a filibuster proof majority inside (more than 60 people inside). On the other hand, it's very boring to be at the bar when there's fewer than 10 noted gentlemen or gentlewomen there.

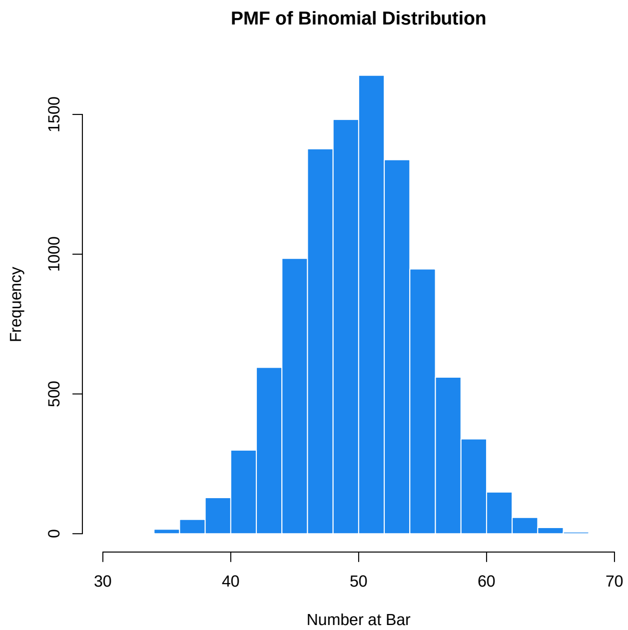
In this problem, we're going to use simulation to help you to decide whether you'll be happier going out to the bar or whether you should stay home.

A

You decide that you're going to go to the bar no matter what. The other 99 Senators independently decide whether to go to the bar, and each has a 50% probability of going to the bar. First, set the seed to 12345 and simulate the total number of Senators in the bar on 10,000 Friday nights. Plot a histogram of the PMF.

Hint: The `rbinom()` function should be helpful.

```
sims <- 10000
set.seed(12345)
pdf("assessment2_a.pdf")
hist(rbinom(n = sims, size = 99, prob = .5) + 1,
     xlab = "Number at Bar",
     ylab = "Frequency",
     main = "PMF of Binomial Distribution",
     col = "dodgerblue2",
     border = "white",
     bty = "l")
graphics.off()
```



B

Now, in R, use simulation to determine the probability that you have a good time at the bar. In other words, what is the probability that between 9 and 59 of the other 99 Senators join you at the bar? Again set the seed to 12345.
Below are three different ways you could simulate this. Each gives you a probability of .9762.

```
## You can simulate the number at the bar all at once and calculate
## the probability from there
```

```
set.seed(12345)
number.at.bar <- rbinom(n = sims, size = 99, prob = .5) + 1
mean(number.at.bar >= 10 & number.at.bar <= 60) #0.9762


## Or you can simulate Friday nights, one at a time

friday.fun <- function(){
  # Simulate the number of people going to the bar and add one
  # because you're definitely going
  going.to.bar <- rbinom(n = 1, size = 99, prob = .5) + 1
  # Check if the number going is within the right bounds
  going.to.bar >= 10 & going.to.bar <= 60
}

set.seed(12345)
results <- c()
for(i in 1:sims){
  results[i] <- friday.fun()
}
mean(results) #0.9762

## Or:

set.seed(12345)
mean(replicate(sims, friday.fun())) #0.9762
```

C

Briefly discuss and critique one of the assumptions that underlies this model. Do you think that assumption is a reasonable one to make?
There's several possibilities here. Probably the most obvious is that you're probably violating the independence assumption of your model. Each Senator probably isn't making the decision to go independently of all other Senators.
You might also think that the assumption that the Senators go to the bar with probability 0.5 is flawed.

D

Now let's assume that you're a member of the US House of Representatives, instead of the Senate, and that their bar is a little bit bigger in size. Continue to assume that each of your colleagues has a 50% probability of going to the bar. Write a function in R that takes, as arguments, the total number of Representatives (or Senators) and the maximum number of people who can comfortably fit into the bar.

```
friday.fun.2 <- function(population, capacity){
  number.at.bar <- rbinom(n = sims, size = population - 1, prob = .5) + 1
  mean(number.at.bar >= 10 & number.at.bar <= capacity)
}


## Or if you plan to use a for loop...

friday.fun.3 <- function(population, capacity){
  # Simulate the number of people going to the bar and add one
  # because you're definitely going
  going.to.bar <- rbinom(n = 1, size = population - 1, prob = .5) + 1
  # Check if the number going is within the right bounds
  going.to.bar >= 10 & going.to.bar <= capacity
}
```

E

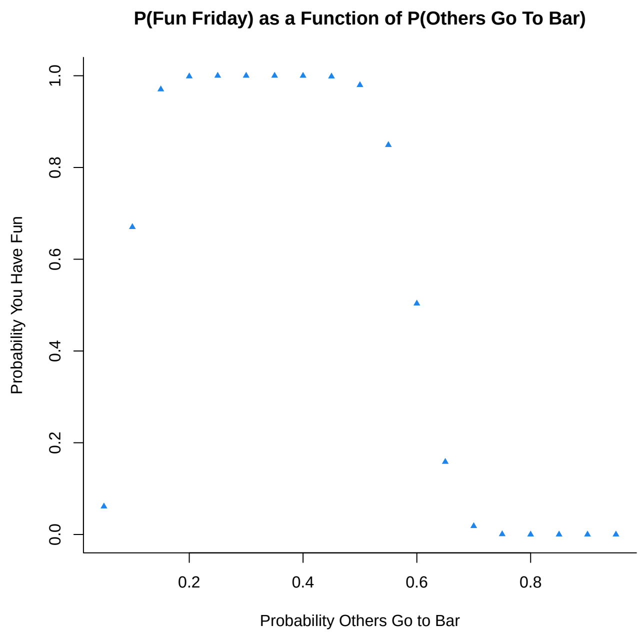
Use the function to estimate the probability that you have a fun time at the bar, given that the House has 435 members[1] (instead of 100) and the their bar's capacity is 225 (instead of 60). Set the seed to 54321.
A few different ways of writing the code all result with a probability of 0.7625.

```
## Using the vectorized version of our function:
set.seed(54321)
friday.fun.2(population = 435, capacity = 225) #0.7625


## Or using a loop with the non-vectorized version of the function:

set.seed(54321)
results <- c()
for(i in 1:sims){
  results[i] <- friday.fun.3(population = 435, capacity = 225)
}
mean(results) #0.7625


## Or:

set.seed(54321)
mean(replicate(sims, friday.fun.3(population = 435, capacity = 225))) #0.7625
```

[1]The Constitution says that only the 435 voting members of the House are allowed into the bar. Sorry Eleanor Holmes Norton.

F

Assume again that you're in the Senate (with 99 colleagues and a bar that fits 60 people). Alter your function so that it also takes as an argument the probability that the other Senators go to the bar. Vary this parameter (which was originally 50%) at 5% intervals from 5% to 95% (i.e 5%, 10%, 15%...95%) and store the results. Set the seed to 1776. Create a graph showing the probability of having a fun Friday night at each level of the parameter that you varied.




```
## Notice below how much faster the vectorized code is than the for loop!

## If we vectorized our code:

friday.fun.4 <- function(population, capacity, probability){
  number.at.bar <- 1 + rbinom(n = sims,
                              size = population - 1,
                              prob = probability)
  mean(number.at.bar >= 10 & number.at.bar <= capacity)
}

set.seed(1776)
probabilities <- seq(.05, .95, .05)
results.by.prob <- c()

for(j in 1:length(probabilities)){
  results.by.prob[j] <- mean(friday.fun.4(population = 100,
```

```
                                                capacity = 60,
                                                probability = probabilities[j]))
}
```

```
## If we don't vectorize the function and use a for-loop instead:

friday.fun.5 <- function(population, capacity, probability){
  # Simulate the number of people going to the bar and add one
  # because you're definitely going
  going.to.bar <- 1 + rbinom(n = 1,
                             size = (population - 1),
                             prob = probability)

  # Check if the number going is within the right bounds
  going.to.bar >= 10 & going.to.bar <= capacity
}

set.seed(1776)
probabilities <- seq(.05, .95, .05)
results.by.prob <- c()

for(j in 1:length(probabilities)){
  results <- c()
  for(i in 1:sims){
    results[i] <- friday.fun.5(population = 100,
                               capacity = 60,
                               probability = probabilities[j])
  }
  results.by.prob[j] <- mean(results)
  print(probabilities[j])
}
```

```
## or
set.seed(1776)
probabilities <- seq(.05, .95, .05)
results.by.prob <- c()

for(j in 1:length(probabilities)){
```

```
  results.by.prob[j] <- mean(replicate(sims,
                                      friday.fun.5(population = 100,
                                                   capacity = 60,
                                                   probability = probabilities[j])))
  print(probabilities[j])
}


## Now create the graph!
results <- data.frame(prob.going.to.bar = probabilities,
                      prob.fun = results.by.prob)

plot(results$prob.going.to.bar,
     results$prob.fun,
     pch = 18,
     bty = "l",
     xlab = "Probability that others go to bar",
     ylab = "Probability you have fun")
```

G

The Constitution is changed so that members of the House of Representatives and Senators are allowed to go to either of the two bars inside the Capitol building. Assume that all the other 534 elected representatives (435 House members plus 99 Senators other than you) go to the Senate bar with probability 0.15, the House bar with probability 0.4, and stay home with probability 0.45. Also assume that you will definitely go to either the Senate bar or the House bar, each with equal probability. Set the seed to 429.

What is the probability that you have a fun Friday night?

Hint: The `rmultinom()` function will be useful. Also, this question is meant to be challenging to code, so don't worry too much if you struggle to get the right answer. The probability of having a fun Friday night is 0.4123.

```
# Write a function
# population: total number of House members + Senators (535)
# capacity: vector of the capacity of the two bars (60, 225)
# probability: vector of the probabilities that everyone else goes
#    to Senate bar, House bar, stays home (.15, .35, .5)
# my.bar.prob: vector of probabilities of going to Senate or House bar (.5, .5)


friday.two.bars <- function(population, capacity, probability, my.bar.prob){

  ## in others, row1 is number at Bar A, row2 is number at Bar B,
```

```
  ## row3 is number who stayed home:
  others <- rmultinom(n = sims, size = population - 1, prob = probability)


  # we could also use rbinom() here:
  me <- rmultinom(n = sims, size = 1, prob = my.bar.prob)

  # did I go to bar 1 (Senate) or bar 2 (House)?
  me <- apply(me, 2, function(x) which(x == 1))

  # This for loop certainly isn't the fastest way to write this code,
  # but it makes it the most clear to follow
  results <- c()
  for(i in 1:length(me)){

    # Figure out which bar I went to on Friday night i
    my.bar <- me[i]

    # Figure out how many other people were at the bar
    # I went to on Friday night i (add one to include me)
    total.at.my.bar <- others[my.bar,i] + 1

    # Figure out if the bar I was at on Friday night i
    # was too crowded, too empty, or just right
    results[i] <- total.at.my.bar >= 10 & total.at.my.bar <= capacity[my.bar]
  }

  return(results)
}

sims <- 10000
set.seed(429)
mean(friday.two.bars(population = 535,
                     capacity = c(60,225),
                     probability = c(.15,.4,.45),
                     my.bar.prob = c(.5, .5)))#0.4123
```

R code

Please submit all your code for this assessment problem as a .R file. Your code should be clean, commented, and executable without error.