Math 151 - Probability Theory - Homework 1

your name here

Due: Friday, January 25, 2019, in class

Important Note:

You should work to turn in assignments that are clear, communicative, and concise. Part of what you need to do is not print pages and pages of output. Additionally, you should remove these exact sentences and the information about HW scoring below.

Click on the *Knit to PDF* icon at the top of R Studio to run the R code and create a PDF document simultaneously. [PDF will only work if either (1) you are using R on the network, or (2) you have LaTeX installed on your computer. Lightweight LaTeX installation here: https://yihui.name/tinytex/]

`You must first install both R and R Studio onto your computer (or use one of the College's computers). See: https://research.pomona.edu/johardin/math151s19/ for many resources.`

```
knitr::opts_chunk$set(message=FALSE, warning=FALSE, fig.height=3, fig.width=5,
                      fig.align = "center")
# install.packages("mosaic")
# I've commented out the install command, because you only use that command once,
# and you use it inside the console.
# The library command belongs in your markdown file, and you run it every time.
library(mosaic)
```

General notes on homework assignments (also see syllabus for policies and suggestions):

- please be neat and organized, this will help me, the grader, and you (in the future) to follow your work.
- be sure to include your name on your assignment
- it is strongly recommended that you start / write out the questions as soon as you get the assignment. This will help you to start thinking how to solve them!
- for R problems, it is required to use R Markdown
- in case of questions, or if you get stuck please don't hesitate to email me (though I'm much less sympathetic to such questions if I receive emails within 24 hours of the due date for the assignment).

Homework assignments will be graded out of 5 points, which are based on a combination of accuracy and effort. Below are rough guidelines for grading.

5 All problems completed with detailed solutions provided and 75% or more of the problems are fully correct.

4 All problems completed with detailed solutions and 50-75% correct; OR close to all problems completed and 75%-100% correct

3 Close to all problems completed with less than 75% correct

2 More than half but fewer than all problems completed and $> 75\%$ correct

1 More than half but fewer than all problems completed and $< 75\%$ correct; OR less than half of problems completed

0 No work submitted, OR half or less than half of the problems submitted and without any detail/work shown to explain the solutions.

Assignment

[Hint: To use this Markdown file, first remove all of the non-R code from above. Keep the R code for the `library` command(s). If you have LaTeX commands you will be able to compile using LaTeX. Either is great, but LaTeX is sometimes difficult to learn. Lightweight LaTeX installation here: https://yihui.name/tinytex/]

[4] DeGroot, section 1.4. Prove Theorem 1.4.11. Note: the intersection $\bigcap$ is often written as a product, i.e., AB is the same as A$\bigcap$B.

There are four things to prove: For two arbitrary events A and B, prove that AB and AB^c are disjoint. Also prove that $A = AB \cup AB^c$. In addition, B and AB^c are disjoint and $A \cup B = B \cup (AB^c)$.

[6] DeGroot, section 1.4. For example, a reasonable description of A^c would be "the event that an odd card is selected" (a description in words), or "the set of outcomes: {Red1, Red3, Red5, Red7, Red9, Blue1, Blue3, Blue5, Blue7, Blue9}" (a description as a subset of S).

Suppose that one card is to be selected from a deck of 20 cards that contain 10 red cards numbered from 1 to 10 and 10 blue cards numbered 1 to 10. Let A be the event that a card with an even number is selected; let B be the event that a blue card is selected; and let C be the event that a card with a number less than 5 is selected. Describe the sample space, S, and describe each of the following events both in words and as subsets of S.

(a) ABC

(b) BC^c

(c) $A \cup B \cup C$

(d) $A(B \cup C)$

(e) $A^cB^cC^c$

[9] DeGroot, section 1.4

Let S be a given sample space and let \$A_1, A_2, $\ldots$ \$ be an infinite sequence of events. For $n = 1, 2, \ldots$, let $B_n = \bigcup_{i=n}^{\infty} A_i$ and let $C_n = \cap_{i=n}^{\infty} A_i$.

(a) Show that $B_1 \supset B_2 \supset \cdots$ and that $C_1 \subset C_2 \subset \cdots$.

(b) Show that an outcome in S belongs to the event $\cap_{n=1}^{\infty} B_n$ if and only if it belongs to an infinite number of the events $A_1, A_2, \ldots$.

(c) Show that an outcome in S belongs to the event $\bigcup_{n=1}^{\infty} C_n$ if and only if it belongs to all the events $A_1, A_2, \ldots$ except possibly a finite number of those events.

[R1] In an upcoming national election for the President of the United States, a pollster plans to predict the winner of the popular vote by taking a random sample of 1000 voters and declaring that the winner will be the one obtaining the most votes in his sample. Suppose that 48 percent of the voters plan to vote for the Republican candidate and 52 percent plan to vote for the Democratic candidate. To get some idea of how reasonable the pollster's plan is,

(a) Write a program to make this prediction by simulation. Repeat the simulation 100 times and see how many times the pollster's prediction would come true. That is, use your simulation to answer the question *what is the probability that the pollster will correctly predict the election with 1000 observations?*

(b) Repeat your experiment, assuming now that 49 percent of the population plan to vote for the Republican candidate; first with a sample of 1000 and then with a sample of 3000. *what is the probability that the pollster will correctly predict the election with 3000 observations?*

n.b. The Gallup Poll uses about 3000. [Problem taken from Grinstead and Snell, Introduction to Probability]

The following R code will may help you complete the simulation. Try: `?rflip` to see how the function works.

```
n.flip = 20
p.heads = 0.5
n.reps = 3
myflips = rflip(n=n.flip, p=p.heads)
myflips
```

```
##
## Flipping 20 coins [ Prob(Heads) = 0.5 ] ...
##
## T T H H H H H H H H H H T H H T T T T T
##
## Number of Heads: 12 [Proportion Heads: 0.6]
```

```
as.numeric(myflips)     # convince yourself that you understand this line
```

```
## [1] 12
```

```
myflips > n.flip/2      # convince yourself that you understand this line
```

```
## [1] TRUE
```

```
manyflips = c()   # initializing a vector to hold our results
for(i in 1:n.reps){
  myflips = rflip(n=n.flip, p=p.heads)
  manyflips = c(manyflips, as.numeric(myflips))  # concantenating the results from reps
  }
manyflips > n.flip/2      # convince yourself that you understand this line
```

```
## [1]  TRUE FALSE  TRUE
```

```
sum(manyflips > n.flip/2) # convince yourself that you understand this line
```

```
## [1] 2
```

Optional fun: You could also generate one flip at a time and plot the sequence using the `cumsum` function in R.