IST 772 Week 2 Breakout 3

Week 2 - Practice with Conditional Probabilities

Instructions:

The table below shows accident reports from three different factories over the past month. Four types of accidents are represented. Each cell contains a count of the number of accidents of the given type at the particular factory:

Accidents	Factory 1	Factory 2	Factory 3
Vehicle	0	6	4
Spill	6	0	6
Equipment	6	4	5
Injury	4	9	0

1. Manually add marginal totals to the table above for cross checking with your R results.

```
factory_1_accidents <- 14
factory_2_accidents <- 19
factory_3_accidents <- 15
```

2. Recreate the matrix in R using the following code:

```
accMatrix <- matrix(data=c(0,6,4, 6,0,6, 6,4,5, 4,9,0),
                    nrow=4,byrow=T,
                    dimnames=list(c("Vehicle","Spill","Equipment","Injury"),
                                  c("Factory 1", "Factory 2", "Factory 3")))
accMatrix
```

```
##           Factory 1 Factory 2 Factory 3
## Vehicle           0         6         4
## Spill             6         0         6
## Equipment         6         4         5
## Injury            4         9         0
```

3. Compute a copy of accMatrix that contains proportions instead of counts. One helpful function that can be called on the whole matrix is sum( ).

```
total_sum <- sum(accMatrix)
#copy of accMatrix with proportions
accMatrix_proportions <- accMatrix / total_sum
# View the matrix with proportions
accMatrix_proportions
```

```
##           Factory 1 Factory 2 Factory 3
## Vehicle        0.00      0.12      0.08
## Spill          0.12      0.00      0.12
## Equipment      0.12      0.08      0.10
## Injury         0.08      0.18      0.00
```

4. Calculate marginal totals for accMatrix. Two helpful functions that can be called are rowSums( ) and colSums( ). Another helpful function is addmargins( ).

```
a <- addmargins(accMatrix_proportions)
a
```

```
##           Factory 1 Factory 2 Factory 3  Sum
## Vehicle        0.00      0.12      0.08 0.20
## Spill          0.12      0.00      0.12 0.24
## Equipment      0.12      0.08      0.10 0.30
## Injury         0.08      0.18      0.00 0.26
## Sum            0.32      0.38      0.30 1.00
```

5. OSHA is auditing the factory that has the worst accident record. Which factory is that? (A helpful function is which.max.) For that factory, list the raw proportions of each type of accident on the console, using the [ ] subsetting technique. For example, you could show the first column of accMatrix with this command: accMatrix[,1]

```
#Factory 2
```

6. Putting your focus solely on accidents at that factory, what's the probability of vehicle accidents at that factory? Write a line of R code that displays the result and include a comment describing what you see.

```
a[1,2]
```

```
## [1] 0.12
```

7. The company that insures these factories wants to understand the most prevalent type of accident across all factories. Add a comment in your code indicating which type of accident that is. For that type of accident, list the raw proportions for each factory on the console, using the [ ] subsetting technique.

```
# Equipment accidents are the most common
```

8. Putting your focus solely on that kind of accident, what's the probability of that kind of accident at each factory? Write a line of R code that displays the result and include a comment describing what you see.

```
a[3,]
```

```
## Factory 1 Factory 2 Factory 3       Sum
##      0.12      0.08      0.10      0.30
```

9. Post your code and comments to the code share window: https://codeshare.io/aJDyRX

Alternative question:

To control the spread of the disease, testing for COVID has become common. However, the tests are not perfect. As a result, whenever a test is administered, there are four possible outcomes: the person either has COVID or not and the test result is positive or not:

```
|    Test result      |
```

True condition	Positive	Negative
Disease	True	False
Positive	Negative	
——————+——————-+——————-		
No disease	False	True
Positive	Negative	
——————+——————-+——————-		

Tests are rated for their performance with two numbers: sensitivity, how well the test detects the disease (i.e., the % of true positives when the person has the disease, i.e., the top row) and specificity, the ability to determine someone doesn't have the disease (i.e., the bottom row). Prevalance is the proportion of people who actually have the disease in the population. Note that a test can be high in both sensitivity and specificity and still have a large number of false positives or negatives depending on the prevalance.

You can find a list of approved COVID tests with sensitivity and specificity here: https://www.centerforhealthsecurity.org/resources/COVID-19/serology/Serology-based-tests-for-COVID-19.html. Note that the sensitivity may change over time, e.g., many tests are not sensitive to the initial days of an infection. You can find data about COVID cases per thousand here: https://datausa.io/coronavirus (among other sites). Pick a test and state and fill out the table above assuming you did 1000 tests total of the population choosen at random (you can use the calculator here: https://www.bmj.com/content/369/bmj.m1808) If a test comes back positive, what are the odds that the person actually has COVID? If a test comes back negative?