Homework #5

Answer the following questions in a single R script called `hw05.R`. Answers must be given by R commands. You cannot simply look at the data set and answer the question via direct inspection. Use comments (#) to indicate which portion of your code answers which question. Be sure that you obtain the correct solutions to each question when you execute your script one line at a time from top to bottom.

Each question will be graded according to the following criteria:

- 0%: No attempts is made to answer the question.
- 25%: An attempt is made that, although unsuccessful, revealed some understanding of what the question was asking.
- 50%: Solution is incorrect, but with some modifications, could be corrected.
- 75%: Solution is incorrect, but easily resolved with minor modifications **OR** solution is correct, but obtained via convoluted reasoning or by avoiding standard approaches.
- 100%: Solution is correct and uses standard approaches.

#1) A fair 6-sided die is rolled repeatedly until it gives 3. Let X count the number of rolls it takes to get a 3.

(a) Write a function in R that performs this experiment and returns the corresponding value for X. (Hint: `sample(1:6,1)` replicates a dice roll.)

(b) Repeat this experiment 10,000 times. Store your results in a tibble. Display your results using a bar chart.

(c) Notice $P(X = 1) = \frac{1}{6}$ and $P(X = 2) = \frac{5}{6^2}$. Find a general formula for $P(X = k)$ (i.e. find the probability mass function for X). Then display these probabilities in a bar chart for $X = 1 \ldots 40$.

(d) Modify the plot in (c) to display the proportion of times each value of X occurs. Then plot this alongside the bar chart from (b) in a "dodge" format.

(e) Technically, X can take on any positive integer. Let's assume the largest X can be is 100. Compute $\mathrm{E}(X)$ under this assumption.

(f) Compute the sample mean from the data generated in part (b).

(g) Compute $\mathrm{Var}(X)$ under the assumptions in (e).

(h) Compute the sample variance from the data generated in part (b).

#2)

(a) Write a function `longest_run` that receives as input a vector of integers that are either 0 or 1 and returns the length of the longest run of 1's in the vector. Test your function against the following to ensure it performs as expected.

```r
longest_run(c(0,1,1,1,0,1)) # Should give 3
longest_run(c(1,0,1,0,0,1)) # Should give 1
longest_run(c(1,1,1,0,1,1)) # Should give 3
longest_run(c(1,1,1,1,1,1)) # Should give 6
longest_run(c(0,0,0,0,0,0)) # Should give 0
```

(b) The following code creates a tibble with 10,000 rows. Each row will contain a vector of 0's and 1's of length 15 under the `coin_flips` column. The probability of success (i.e. "heads") is $\theta = 0.5$.

```r
library(Rlab)

df_random <- tibble(index=1:10000) %>% rowwise() %>%
  mutate(coin_flips=lst(rbern(20,0.6)))
```

Let X count the number of 1's in each vector. Let Y count the longest run of 1's in each vector. Create columns in this tibble for X and Y.

(c) Give a bar graph that counts the number of times each Y occurs.

(d) Use `geom_tile` to display the number of times each pair of values for X and Y occurs in the tibble.

(e) Consider a data set where each row contains a measurement of X and a measurement of Y. When considering the i-th row, we denote the measurement of X by x_i and the measurement of Y by y_i. Suppose there are n rows. The **sample covariance** for these measurements is

$$\text{cov}_{x,y} = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})$$

where $\overline{x}$ is the sample mean of the x_i and $\overline{y}$ is the sample mean of the y_i. Compute the sample covariance for our tibble.

(f) The **Pearson correlation coefficient** is

$$r_{x,y} = \frac{\text{cov}_{x,y}}{s_x s_y}$$

where

$$s_x = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})^2}, \qquad s_y = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(y_i - \overline{y})^2}.$$

Recall s_x is the sample standard deviation for the x_i and s_y is the sample standard deviation for the y_i. Compute the Pearson correlation coefficient for our tibble.

(g) The following code snippet lists every possible outcome in this experiment.

```r
df_prob <- tibble(ind=1:2^15) %>% rowwise() %>%
  mutate(coin_flips=lst(as.integer(intToBits(ind))[1:15]))
```

Create columns for the random variables X and Y defined in part (b).

(h) Use `group_by` and `summarize` to give counts for the number of times each pair of values for X and Y occur in `df_prob`. Then add a column `prob` to this tibble that gives the probability for each pair of values for X and Y (recall that each outcome listed in `df_prob` has a probability of $\frac{1}{2^{15}}$). I highly recommend running `ungroup` after `summarize` since failing to do this will cause error in the subsequent calculations.

(i) Compute $\mathrm{E}(X)$ and $\mathrm{E}(Y)$.

(j) The **covariance** of random variables X and Y is

$$\mathrm{Cov}(X, Y) = \mathrm{E}((X - \mathrm{E}(X))(Y - \mathrm{E}(Y))).$$

Compute this value using the tibble in part (h). Your answer should be similar to what you computed in part (e).

(k) Compute $\sigma_X = \mathrm{SD}(X)$ and $\sigma_Y = \mathrm{SD}(Y)$.

(l) The **Pearson correlation coefficient** for random variables X and Y is

$$\rho_{X,Y} = \frac{\mathrm{Cov}(X, Y)}{\sigma_X \sigma_Y}.$$

Compute this value using the tibble in part (h). Your answer should be similar to what you computed in part (e).