RExersice_W2

Zhiyu Li

2021/10/10

Q1

```
#Load necessary package(s)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.4     v dplyr   1.0.7
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   2.0.1     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(ggplot2)
library(ggpubr)
invisible()
```

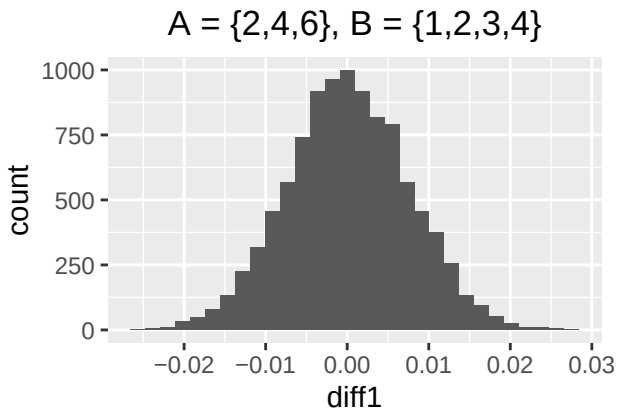
```
#Define a function to generate n times tossing
tossing <- function(n){
  return(ceiling(runif(n, 0, 6)))
}
```

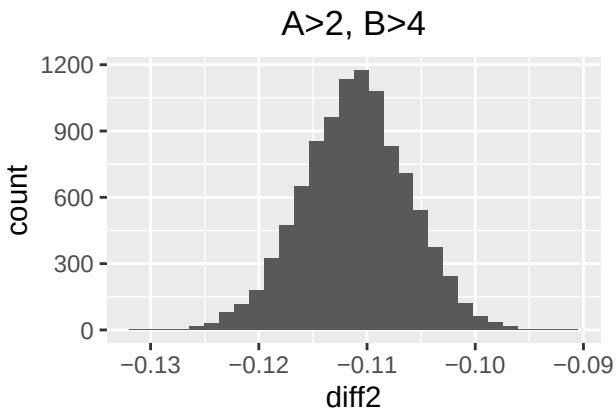
```
#Define a function for calculating P(A),P(B),P(AB)
Prob_A_B_AB <- function(n, A, B){
  #assign tossing result to a local variable
  n_times_tossing <- tossing(n)

  Prob_A <- sum(n_times_tossing %in% A)/n

  Prob_B <- sum(n_times_tossing %in% B)/n

  Prob_AB <- sum((n_times_tossing %in% intersect(A, B)))/n

  return(c(Prob_A, Prob_B, Prob_AB))
}
```

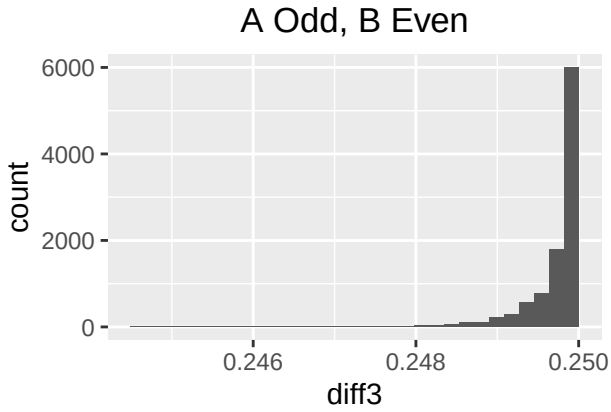
```r
#Define a function for calculating the difference between
#P(A)*P(B) and P(AB)
Diff_Between_A_B_AB <- function(n, A = c(2,4,6), B = c(1,2,3,4)){
  Prob <- Prob_A_B_AB(n, A, B)
  ProbA <- Prob[1]
  ProbB <- Prob[2]
  ProbAB <- Prob[3]
  Diff <- ProbA * ProbB - ProbAB
  return(Diff)
}
```

```r
#Default setting is A = {2,4,6}, B = {1,2,3,4}
diff1 <- replicate(10000, Diff_Between_A_B_AB(1000))
diff1_df <- as.data.frame(diff1)
d1<- ggplot(data = diff1_df, aes(x = diff1))+
  geom_histogram()+
  labs(title = 'A = {2,4,6}, B = {1,2,3,4}')+
  theme(plot.title = element_text(hjust = 0.5))

#Let A > 2, that A = {3,4,5,6}, and B > 4, that B = {5,6}.
#A and B are not independent.
A = c(3,4,5,6)
B = c(5,6)
diff2 <- replicate(10000, Diff_Between_A_B_AB(1000, A, B))
diff2_df <- as.data.frame(diff2)
d2<- ggplot(data = diff2_df, aes(x = diff2))+
  geom_histogram()+
  labs(title = 'A>2, B>4')+
  theme(plot.title = element_text(hjust = 0.5))

#Let A is odd, that A = {1,3,5}, and B is even, that B = {2,4,6}.
#A and B are not independent.
A = c(1,3,5)
B = c(2,4,6)
diff3 <- replicate(10000, Diff_Between_A_B_AB(1000, A, B))
diff3_df <- as.data.frame(diff3)
d3<- ggplot(data = diff3_df, aes(x = diff3))+
  geom_histogram()+
  labs(title = 'A Odd, B Even')+
  theme(plot.title = element_text(hjust = 0.5))

ggarrange(d1,d2,d3)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```








```
#Means for Differences of 3 Events
c(
  Independent_Events = diff1 %>% mean(),
  Dependent_Events_I = diff2 %>% mean(),
  Dependent_Events_II = diff3 %>% mean()
)
```

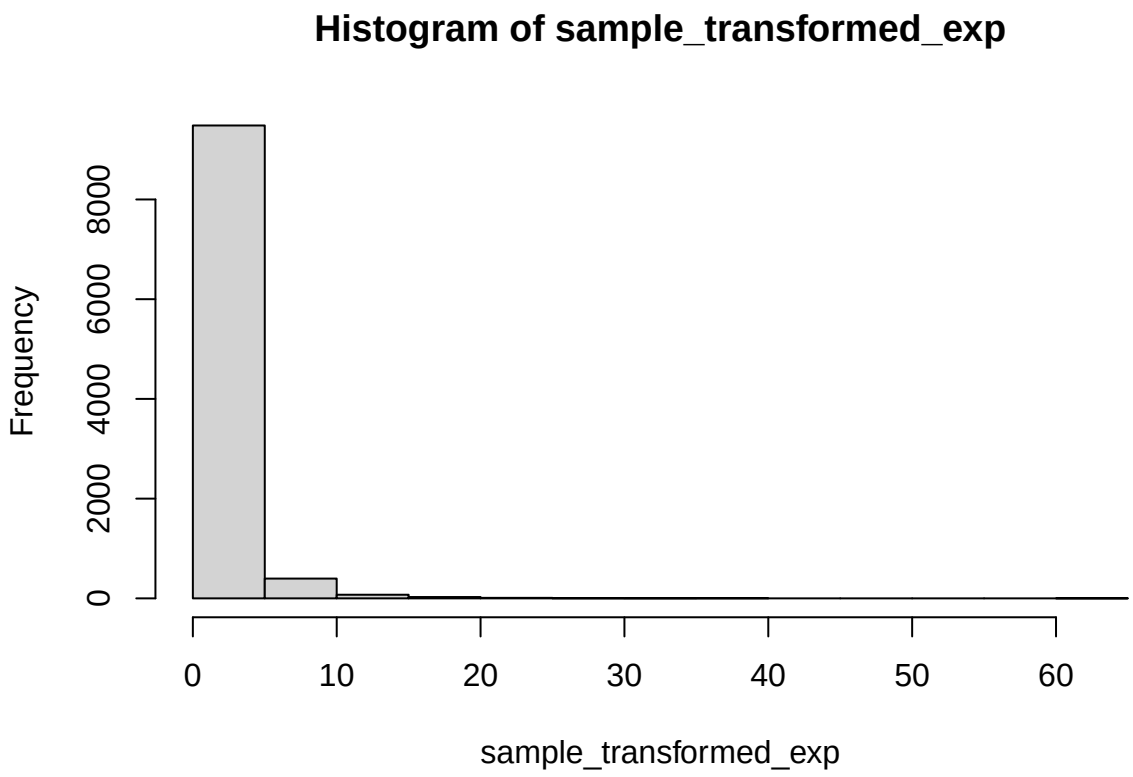
```
##  Independent_Events  Dependent_Events_I Dependent_Events_II
##        0.0001266412       -0.1110667955        0.2497407580
```

Q5(b)

```
#Draw a sample of 10,000 from N(0,1)
sample_normal <- rnorm(10000, mean = 0, sd = 1)
sample_transformed_exp <- exp(sample_normal)
mean_sample_transformed <- mean(sample_transformed_exp)
sd2_sample_transformed <- sd(sample_transformed_exp)^2
cat(' sample mean:', mean_sample_transformed,'\n','sample variance',sd2_sample_transformed)
```

```
##  sample mean: 1.651046
##  sample variance 5.21628
```

```r
#Histogram for Y
hist(sample_transformed_exp)
```




```r
#Define function for calculating mean of y
mean_of_y <- function(mu, sigma){
  mean <- exp((sigma^2)/2 + mu)
  return(mean)
}
#Define function for calculating variance of y
sd_square_of_y <- function(mu, sigma){
  std2 <- exp(sigma^2 + 2*mu)*(exp(sigma^2)-1)
  return(std2)
}
```

```r
cat(' sample mean,expectation',mean_sample_transformed,mean_of_y(0,1),'\n','sample variance,Variance',s
```

```
##  sample mean,expectation 1.651046 1.648721
##  sample variance,Variance 5.21628 4.670774
```