TTC Delays

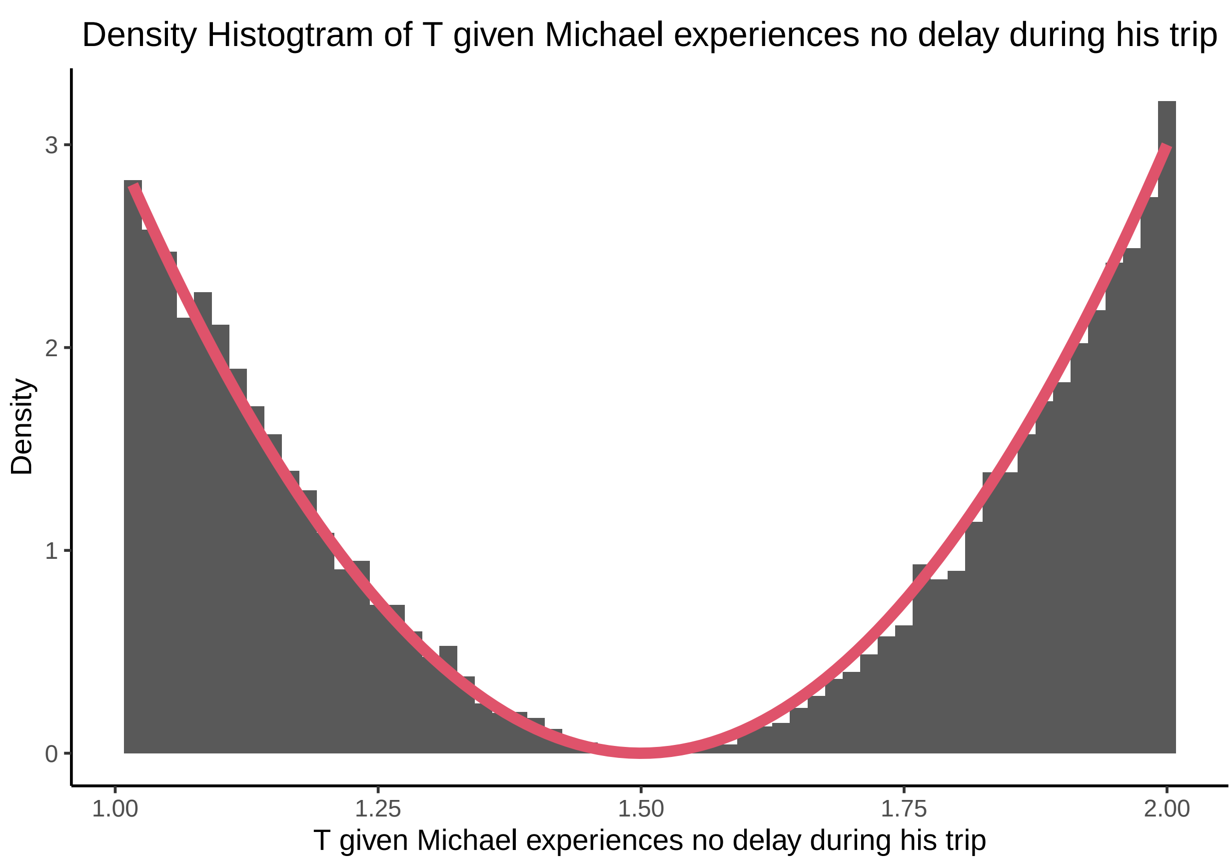
Nimit A. Bhanshali

6/16/2022

```r
# compute pdf of T given N = n
pdf_T_given_N <- function(t, n) {
  k <- 3 / ((n + 1)^3)

  if (t >= 1 && t <= n + 2) {
    return(k * (2 * t - n - 3)^2)
  }

  return(0)
}

# compute cdf of T given N = n
cdf_T_given_N <- function(t, n) {
  k <- 3 / ((n + 1)^3)

  if (t < 1) {
    return(0)
  }

  if (t >= 1 && t <= n + 2) {
    return(k * ((4 * t^3 / 3) - (2 * n * t^2) - (6 * t^2) + (t * n^2) +
                  (6 * n * t) + (9 * t) - (n^2) - (4 * n) - (13 / 3)))
  }

  return(1)
}

# 1.
set.seed(237)
N <- 10000 # number of simulations
U <- runif(N) # N simulations of U ~ U(0,1)
t <- numeric(N)

for(ind in 1:N) { # simulation loop for T | N = 0
  t_tmp <- 1 # starts at 1

  while(cdf_T_given_N(t_tmp, 0) < U[ind]){
  # while cdf_T_given_N is smaller than U, increases t_tmp by a minute
    t_tmp <- t_tmp + (1/60) # measurement precision is in minutes
  }

  t[ind] <- t_tmp # store the t_tmp value
```

```r
}

# pdf_T_given_N function when N = 0
pdft <- function(t) {
  pdf_T_given_N(t, 0)
}

# plot density histogram of N simulations of T | N = 0
# plot pdf_T_given_N when N = 0
library(ggplot2)
ggplot() +
   ggtitle("Density Histogtram of T given Michael experiences no delay during his trip") +
   theme_classic() +
   # From ggplot2 documentation: https://ggplot2.tidyverse.org/reference/element.html
   theme(plot.title = element_text(hjust = 0.5)) +
   geom_histogram(aes(x = t, y = after_stat(density)), binwidth = (1/60)) +
   geom_function(fun = pdft, colour =2, size = 2) +
   labs(x ="T given Michael experiences no delay during his trip",
        y = "Density")
```

```
## Warning in t >= 1 && t <= n + 2: 'length(x) = 101 > 1' in coercion to
## 'logical(1)'

## Warning in t >= 1 && t <= n + 2: 'length(x) = 101 > 1' in coercion to
## 'logical(1)'
```



```r
set.seed(237)
# 2.
N <- 10000 # number of simulations
ET0 <- sum(t)/N
print("2. Estimation of the expected value of T given Michael experiences no delay during his trip:")
```

```
## [1] "2. Estimation of the expected value of T given Michael experiences no delay during his trip:"
```

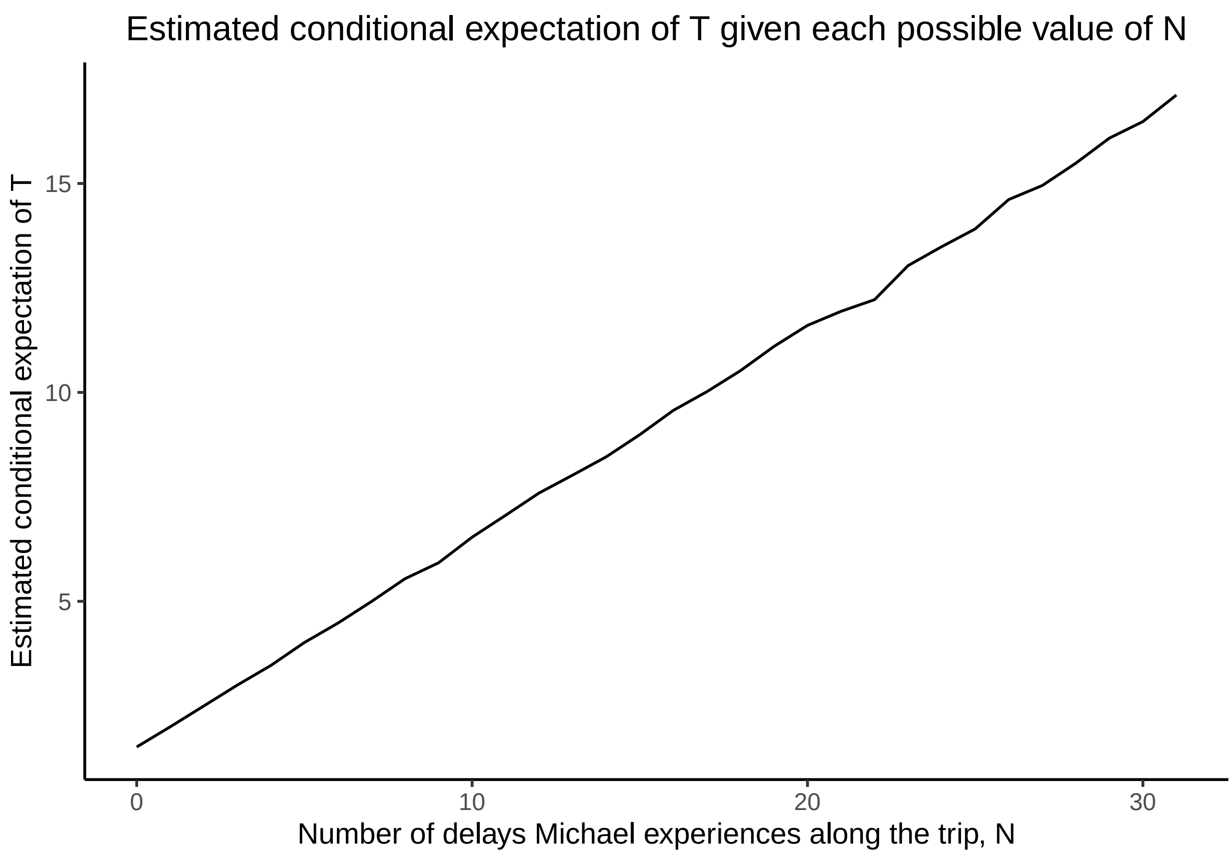
```r
round(ET0, digits = 3)
```

```
## [1] 1.514
```

The expected value of T given Michael experiences no delay during his trip computed in Question 1.4 is 1.500. Our estimate of the expected value of T given Michael experiences no delay during his trip is approximately 1.514. Since, the estimated and computed values are so similar we can conclude that our estimate is a good estimate of the expected value of T given Michael experiences no delay during his trip.

```r
# 3.
set.seed(237)
N <- 10000 # number of simulations

tn <- matrix(0, N, 32)

for(delay in 1:32){ # looping through each possible value of N

  U <- runif(N) # N simulations of U ~ U(0,1)

  for(ind in 1:N) { # simulation loop for T | N = delay - 1
    tn_tmp <- 1 # starts at 1

    while(cdf_T_given_N(tn_tmp, delay - 1) < U[ind]){
    # while cdf_T_given_N is smaller than U, increases tn_tmp by a minute
      tn_tmp <- tn_tmp + (1/60) # measurement precision is in minutes
    }

    tn[ind, delay] <- tn_tmp # store the tn_tmp value
  }

}

tnvector <- c(tn) # converts the matrix into a vector
ETN <- numeric()

for (delay in 0:31) { # looping through each possible value of N
  samp <- tnvector[((delay * N) + 1) : ((delay + 1) * N) ]
  # slicing the samples for the particular value of N
  ETN[delay + 1] <- sum(samp) / N # computing the expectation
}

# plotting conditional expectation of T for each possible value of N
ggplot() +
   ggtitle("Estimated conditional expectation of T given each possible value of N") +
```

```r
  theme_classic() +
  # From ggplot2 documentation: https://ggplot2.tidyverse.org/reference/element.html
  theme(plot.title = element_text(hjust = 0.5)) +
  geom_line(aes(x = 0:31, y = ETN)) +
  labs(x ="Number of delays Michael experiences along the trip, N",
       y = "Estimated conditional expectation of T")
```



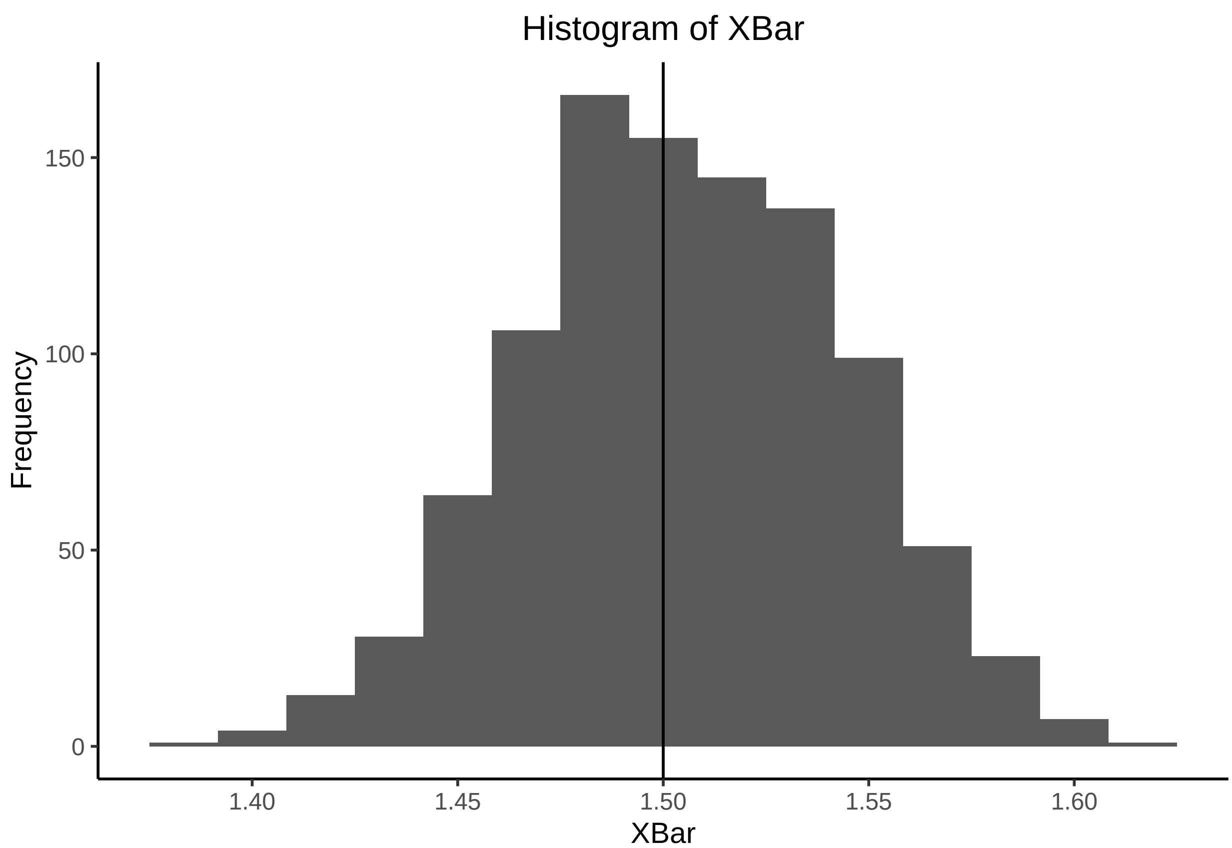

```r
# 4.
set.seed(237)
ET <- 0 # accumulator for marginal mean of T
for (delay in 0:31) { # looping through each possible value of N
  # taking the estimated conditional expectation for a given value of N and
  # multiplying it by P(N = n)
  expectation <- ETN[delay + 1] * dbinom(x = delay, size = 31, prob = (1/12))
  ET <- ET + expectation # Adding the weighted expectation to the accumulator
}
print("4. Estimated marginal mean, E[T]:")
```

```
## [1] "4. Estimated marginal mean, E[T]:"
```

```r
round(ET, digits = 3)
```

```
## [1] 2.785
```

```r
# 1.
set.seed(237)
n <- 0 # number of delays
N <- 1000 # number of simulations

xbar <- numeric(N)
for (samp in 1:N){ # N simulation loop

  U <- runif(100) # 100 simulations of U ~ U(0,1)
  x <- numeric(100)

  for(ind in 1:100) { # 100 simulations of T | N = 0
    x_tmp <- 1 # starts at 1

    while(cdf_T_given_N(x_tmp, n) < U[ind]){
      # while cdf_T_given_N is smaller than U, increases x_tmp by a minute
      x_tmp <- x_tmp + (1/60) # measurement precision is in minutes
    }

    x[ind] <- x_tmp # store the x_tmp value
  }

  xbar[samp] <- sum(x) / 100
}

# plot the histogram of xbar
# plot a vertical line at theoretical E(T|N = 0)
library(ggplot2)
ggplot() +
   ggtitle("Histogram of XBar") +
   theme_classic() +
   # From ggplot2 documentation: https://ggplot2.tidyverse.org/reference/element.html
   theme(plot.title = element_text(hjust = 0.5)) +
   geom_histogram(aes(x = xbar), binwidth = (1/60)) +
   geom_vline(xintercept = (3/2)) +
   labs(x = "XBar", y = "Frequency")
```




```
#2.
Exbar <- sum(xbar) / N # expectation of xbar
Exbar2 <- sum(xbar^2) / N # expectation of (xbar)^2
Varxbar <- Exbar2 - (Exbar^2) # variance of xbar

print("2a. Estimated expecation of Xbar")
```

```
## [1] "2a. Estimated expecation of Xbar"
```

```
round(Exbar, digits = 3)
```

```
## [1] 1.506
```

```
print("2b. Estimated variance of Xbar")
```

```
## [1] "2b. Estimated variance of Xbar"
```

```
round(Varxbar, digits = 3)
```

```
## [1] 0.001
```

```r
# 3.
# estimating E(T|N = 0) using 100 simulated samples
set.seed(237)
U <- runif(100)
x <- numeric(100)
for(ind in 1:100) {
  x_tmp <- 1
  while(cdf_T_given_N(x_tmp, 0) < U[ind]){
    x_tmp <- x_tmp + (1/60)
  }
  x[ind] <- x_tmp
}
ex100 <- sum(x)/100
```

```r
# comparing estimate of E(T|N = 0) using 100 simulated sample with xbar
set.seed(237)
optimistic <- xbar < ex100
print("3. Probability that estimate of E(T|N = 0) using 100 simulated samples is overly optimistic:")
```

```
## [1] "3. Probability that estimate of E(T|N = 0) using 100 simulated samples is overly optimistic:"
```

```r
round(sum(optimistic)/1000, digits = 3)
```

```
## [1] 0.664
```

This probability will be very similar if not exactly the same with a different set of simulated Xbar samples. Since, the variance of Xbar, 0.001, is so small, the concentration of a different set of Xbar samples is very tight around its expectation. Therefore, we can expect a different set of simulated Xbar samples to approximately yield the same values. Therefore, the probability that the estimate is overly optimistic is measured against almost the same set of Xbar samples. Hence, we can conclude that the probability that the estimate is overly optimistic will remain approximately, if not exactly, the same even with a different set of simulated Xbar samples.