Freddie Perez
MATH 611
HW1
9/8/2019

1 HW ω_i is the wait time for the ith Person

1a) Determine $P(\omega_2 \geq C)$ for $C > 0$. $T_i \sim Exp(\lambda)$ iid, $S_i \sim Exp(\mu)$ iid

$\omega_i = D_{i-1} - A_i$, we A_i is the Arrival time of ith Person and D_i is the Departure time of the ith person

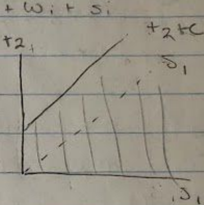
We know $\omega_2 = \begin{cases} 0, & \text{if } A_2 > D_1 \\ D_1 - A_2, & \text{o/w} \end{cases}$

$P(D_1 - A_2 > C)$ since $A_i = \Sigma T_i$, $D_i = A_i + \omega_i + S_i$

$= P(A_1 + S_1 + \omega_1 - A_2 \geq C)$, we Assume $\omega_1 = 0$

$= P(\cancel{T_1} + S_1 + 0 - \cancel{T_1} - T_2 \geq C)$

$= P(S_1 - T_2 \geq C)$



Joint pdf is $f(t, s_1) = \lambda e^{-\lambda t_2} \mu e^{-\mu s_1}$

$\therefore P(S_1 \geq C + T_2) = \int_0^{\infty} \int_{C+T_2}^{\infty} \lambda e^{-\lambda t_2} \mu e^{-\mu s_1} \, ds_1 \, dt_2$

$= \int_0^{\infty} \lambda e^{-\lambda t_2} \int_{C+T_2}^{\infty} \mu e^{-\mu s_1} \, ds_1 \, dt_2$

$= \int_0^{\infty} \lambda e^{-\lambda t_2} \left[-e^{-\mu s_1} \right] \Big|_{C+T_2}^{\infty} dt_2 = \int_0^{\infty} \lambda e^{-\lambda t_2} \left[0 - (-e^{-\mu(C+T_2)}) \right] dt_2$

$= \int_0^{\infty} \lambda \exp\{-\lambda t_2 - \mu c - \mu t_2\} \, dt_2$

$= \int_0^{\infty} \lambda \exp\{-t_2(\mu + \lambda)\} \exp\{-\mu c\} \, dt_2$

$= \lambda \exp\{-\mu c\} \int_0^{\infty} \exp\{-t_2(\mu+\lambda)\} \, dt_2$

$= \lambda \exp\{-\mu c\} \left\{ \left(\frac{-1}{\mu+\lambda} \right) \exp\{t_2(\mu+\lambda)\} \right\} \Big|_0^{\infty}$

$= \dfrac{\lambda \exp\{-\mu c\}}{(\mu+\lambda)} = P(\omega_2 \geq c), \quad \forall c > 0$

Freddie Perez
MATH 611
HW1
9/8/2019

1b) $P(\omega_3 \geq c)$, where $\omega_3 = D_2 - A_3$, $A_3 = T_1 + T_2 + T_3$

$D_2 = A_2 + S_2 + \omega_2$, $A_2 = T_1 + T_2$

$\omega_2 = D_1 - A_2$, $A_2 = T_1 + T_2$

$D_1 = A_1 + S_1 + \cancel{\omega_1}^{\,0}$, $A_1 = T_1$

$\Rightarrow D_2 - A_3 = A_2 + S_2 + \omega_2 - A_3 = \cancel{T_1} + \cancel{T_2} + S_2 + \omega_2 - \cancel{T_1} - \cancel{T_2} - T_3$

$= S_2 + \omega_2 - T_3$, if $A_2 > D_1$, then $\omega_2 = 0$ such that

$P(\omega_3 \geq c) = P(S_2 - T_3 \geq c,\ A_2 \geq D_1) = P(S_2 - T_3 \geq c,\ T_2 > S_1)$ (1)

Else if $A_2 < D_1$, then $\omega_2 = D_1 - A_2$ such that

$\Rightarrow S_2 + \omega_2 - T_3 = S_2 + D_1 - A_2 - T_3 = S_2 + A_1 + S_1 + 0 - A_2 - T_3$

$= S_2 + \cancel{T_1} + S_1 - \cancel{T_1} - T_2 - T_3 = S_2 + S_1 - T_2 - T_3$ st

$P(\omega_3 \geq c) = P(S_2 + S_1 - T_2 - T_3 \geq c,\ A_2 < D_1)$

$= P(S_2 + S_1 - T_2 - T_3 \geq c,\ T_2 < S_1)$ (2)

(1) $P(S_2 - T_3 > c,\ T_2 > S_1) = P(S_2 > c + T_3) \cdot P(T_2 > S_1)$ due to independence

$$= \left(\int_0^\infty \int_{c+T_3}^\infty \mu \exp(-\mu s_2)\, \lambda \exp(-\lambda t_3)\, ds_2\, dt_3\right)\left(\int_0^\infty \int_{s_1}^\infty \lambda \exp(-\lambda t_2)\, \mu \exp(-\mu s_1)\, dt_2\, ds_1\right)$$

(2) $P(S_2 + S_1 - T_2 - T_3 \geq c,\ T_2 < S_1) = P(S_2 + S_1 \geq c + T_2 + T_3,\ T_2 < S_1)$

$$= \int_0^\infty ds_1 \int_0^{s_1} dt_2 \int_0^\infty ds_2 \int_0^{s_2 + s_1 - c - t_2} dt_3$$

$$= \int_0^\infty \int_0^{s_1} \int_0^\infty \int_0^{s_2 + s_1 - c - t_2} \left(\mu e^{-\mu s_1}\right)\left(\lambda e^{-\lambda t_2}\right)\left(\mu e^{-\mu s_2}\right)\left(\lambda e^{-\lambda t_3}\right) ds_1\, dt_2\, ds_2\, dt_3$$

$\underline{P(\omega_3 \geq c) = (1) + (2)}$

Freddie Perez
MATH 611
HW1
9/8/2019

(1e)

Using the Monte Carlo Approach, we will Assume

$$V_n = \begin{cases} 0, & \text{if } W_n < 1 \\ 1, & \text{if } W_n \geq 1 \end{cases} = I(W_n \geq 1) \text{ As an indicater function}$$

such that $E(V_n) = 0 \cdot P(V_n < 1) + 1 \cdot P(V_n \geq 1) = P(V_n \geq 1)$

in order to calculate the $E(V_n)$, we will use the Law of Large Numbers such that the

$$P(W_n \geq 1) \cong \frac{1}{m} \sum_{i=1}^{m} I \cdot (w_n^{(i)} \geq 1) \approx E(V_n)$$

As we are only taking the number samples where $w_n^{(i)} \geq 1$ for the m^{th} person, and dividing By the total number of samples for the m^{th} person.

611 Freddie Perez HW1

Freddie Perez

September 3, 2019

1(c)

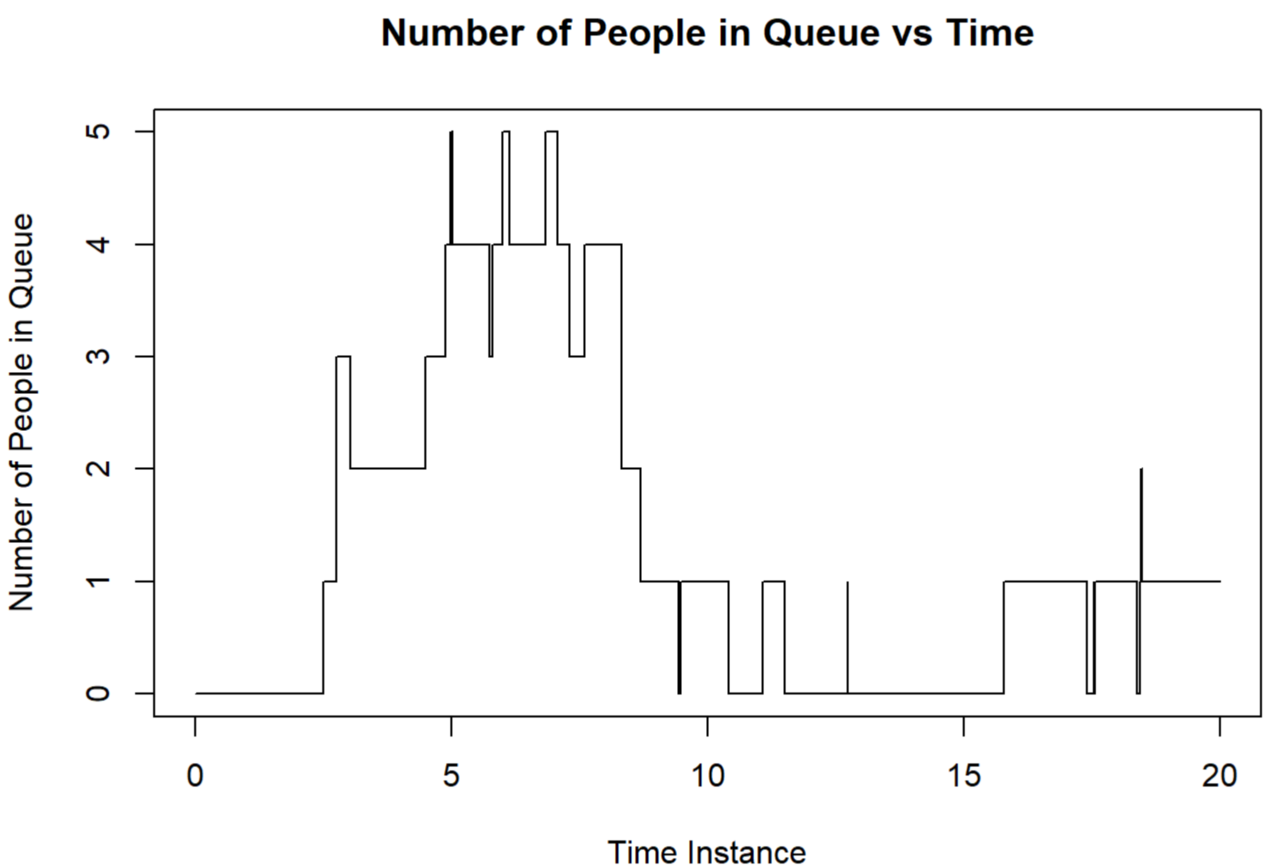
```
set.seed(1234)
WaitingTimes <- function(n,lam, mu){
  w <- rep(0,n)
  # do we set Ts and Ss one time or do we keep reseting
  # Ti samples
  ts <- c(0, rexp(n-1, rate = lam))
  # Si samples
  ss <- rexp(100, rate = mu)

  for (i in 2:n){
    # For loops checks Ai is less than the Departure time of (ith-1) person
    di_1 <- sum(ts[1:(i-1)]) + w[(i-1)] + ss[(i-1)]
    ai <- sum(ts[1:i])
    if (ai < di_1){

      # Stores sampled waiting time for the ith person
      w[i] <- di_1 - ai

    }else{
      # Otherwise stores 0 inplace for the ith sampled waiting time
      w[i] <- 0
    }
  }
  return(w)
}

WaitingTimes(10, 1, 1)
```

```
##  [1] 0.000000 0.000000 1.633318 3.222841 3.138758 5.804033 7.464764
##  [8] 6.672408 7.346750 6.523324
```

1(d)

```r
set.seed(1234)
plotQ <- function(t,lam, mu){

  # Assumes starting person
  n <- 1

  # Calculates Arrival time for person i=1
  ts <- c(rexp(1, rate=lam))

  # Checks if the total time is less than the time t input
  while(sum(ts)<t){
    n <- n + 1

    # Calculates 10 additional peoples times of arrival
    ts_i <- rexp(1, rate=lam)

    # Appends value to the vector ts
    ts <- c(ts, ts_i)

    # Re checks if the sum of all the values is less than the time t input
  }

  w <- rep(0,n)

  ss <- rexp(n, rate = mu)

  # Initializes arrival times for later records
  Ai <- c(ts[1], rep(0,n-1))

  # Intitalizes Departure times for later records
  Di <- c(ts[1] + ss[1], rep(0, n-1))

  for (i in 2:n){
    # Stores newest Arrival time for the ith customer
    Ai[i] <- sum(ts[1:i])

    # Checks the condition Ai is less than Di_1
    if (sum(ts[1:i]) < (sum(ts[1:(i-1)]) + w[(i-1)] + ss[(i-1)])){

      # Stores waiting time of the ith person in vector
      w[i] <- (sum(ts[1:(i-1)]) + w[(i-1)] + ss[(i-1)]) - (sum(ts[1:i]))

    }else{
      # IF condition is not met, assign 0 to ith waiting time.
      w[i] <- 0
    }

    # Stores departure time for the ith person
    Di[i] <- Ai[i] + ss[i] + w[(i)]
  }

  # Creates X-axis values
  xaxis = seq(0, t, .00001)
```

```
  # Initializes Y-axis
  yaxis = 1:length(xaxis)


  # Determines the number of people in Queue at time T based on the arrival and departure times
  for(i in 1:length(xaxis)){
    yaxis[i] = (length(Ai[Ai<=xaxis[i]]) - length(Di[Di<=xaxis[i]]))
  }

  return(plot(xaxis, yaxis, type='l', xlab = 'Time Instance', ylab = 'Number of People in Queue'
,
    main = 'Number of People in Queue vs Time'))
}

plotQ(20, 1, 1)
```



1(e, a)

```r
set.seed(1234)
MonteCarloApp <- function(n, i){

  # Takes in (n) the number of samples that will be taken for the (i)th person

  # Initializing count of when the W_i waiting time is greater than or equal to 1
  greater_1 <- 0

  for (trial in 1:n){

    # Taking a sample for the Ith person waiting time
    sampled_val <- WaitingTimes(i,1,1)[i]

    if (sampled_val >= 1){
      greater_1 = greater_1 + 1
    }else{
      greater_1 = greater_1 + 0
    }
  }

  prob <- greater_1/n

  return(cat('Monte Carlo Approach: \n', toString(prob), '\n\n'))
}

exact_prob <- function(c, lam, mu){

  # Calculates the exact probabilty of the second persons waiting time being greater than or equ
al to 1

  val = (lam * exp(-1 * mu * c))/(mu + lam)


  return(cat("Exact Probability:\n", toString(val), '\n'))
}

MonteCarloApp(100000, 2)
```

```
## Monte Carlo Approach:
##  0.18263
```

```r
exact_prob(1, 1, 1)
```

```
## Exact Probability:
##  0.183939720585721
```

1(e, b)

```r
set.seed(1234)
MonteCarloApp(100000, 100)
```

```
## Monte Carlo Approach:
##  0.88637
```