ECS 132 Spring 2024: Assignment 4
Solution

This problem set covers:

- Problem 1 deals with the coupon collector problem
- Problems 2-4 concern Poisson processes
- Problem 5 is about quantiles, boxplots and data analysis

1 Problem

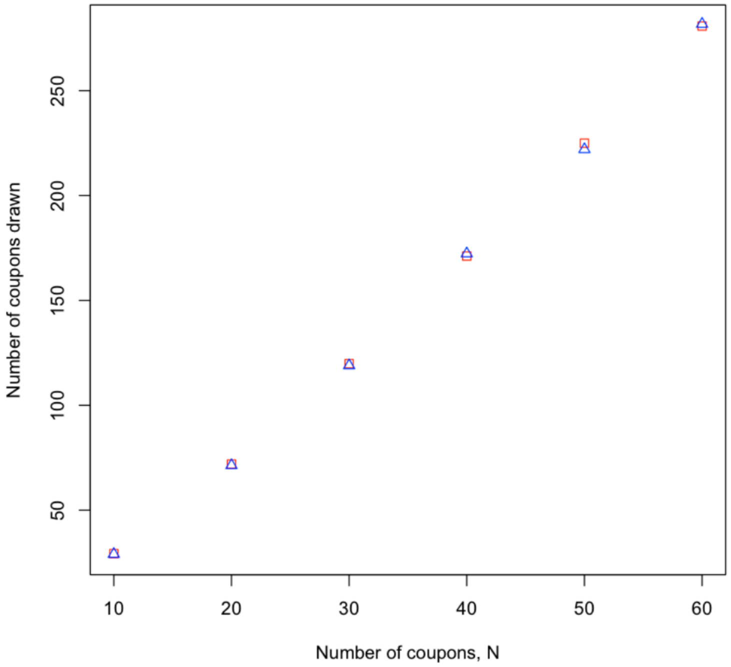
Recall the coupon collector problem discussed in class which has many applications in computer science. Consider a bag that contains N different types of coupons (say coupons numbered $1 \dots N$). There are infinite number of each type of coupon. Each time a coupon is drawn from the bag, it is independent of the previous selection and equally likely to be any of the N types. Since there is an infinite number of each type, one can view this as sampling with replacement. Let T correspond to the random variable that denotes the number of total coupons that needed to be collected in order to obtain a complete set of at least one of each type of coupon. Write a R simulation code to compute E(T) considering the following:

- N denotes the total number of coupons. Run your numerical simulation to develop an estimate of E(T) and plot E(T) for $N = 10, 20, 30, 40, 50, 60$. (Use 1000 trials or more, i.e, Nsim $\geq$ 1000.)
- We showed in class that for large N, $E(T)$ can be approximated by $N \log(N) + 0.577N + 0.5$. In the same plot show the theoretical value and summarize your observation regarding the accuracy of the approximation.

Answer

Here is the code.

```
N=c(10,20,30,40,50,60)     # different numbers of coupons to consider
T=rep(0,length(N))         # T[j]=mean number of draws needed for N[j] coupons
Theory=rep(0,length(N))    # Theory[j]=theoretical number for N[j] coupons
NSim=1000                  # The number of simulations to perform for each N value

for (j in 1:length(N)){
    num=rep(0,NSim)    # initialize fresh for each N[j] value
    for (i in 1:NSim) {
        trials=rep(0,0)
        while (length(unique(as.vector(trials))) < N[j]) {
            trials<-cbind(sample(1:N[j],1),trials)
            num[i]=num[i]+1
            }
        }
        T[j]=mean(num)
        Theory[j] = N[j]*log(N[j]) + 0.5771*N[j] + 0.5
}

plot(N,Theory,col="red",pch=0)
points(N,T,col="blue",pch=2)
```



Observations: There are two reasons why the simulated values differ from the theoretical value and they trade-off against each other: (1) the theoretical value is valid for large N, so the agreement should improve with N, but (2) the larger the sample size, greater the NSim value needed to converge to a stable value. (We will learn about error bars a little later.)

2 Problem

Consider that the entire human genome can be written in the form of a long book broken into pages. Suppose that the number of mutations on a single page of this book has a Poisson distribution with parameter $\lambda = 1$ (i.e., on average we expect 1 mutation per page).

1. For a given page, calculate the probability that there are at least 2 mutations on the page.
2. Now consider three pages, calculate the probability that there are no more than 2 mutations over all three of those pages.
3. For a given page, calculate the probability that there are at least 2 mutations on the page given that you have already spotted one mutation.

Answer

1. Let X denote the number of mutations on the page. Then we need to find $P(X \geq 2)$.
Recall the Poisson distribution

$$P(X = k) = \frac{\lambda^k e^{-\lambda}}{k!}$$

$$\begin{aligned} P(X \geq 2) &= 1 - P(X=0) - P(X=1) \\ &= 1 - e^{-1} - e^{-1} = 1 - 2e^{-1} \\ &= 0.264 \end{aligned}$$

2. Solve for $P(X \leq 2)$ over three pages. We were given that the expected number per page $\lambda = 1$, so the expected number over three pages $\lambda_3 = 3$ which is what we will use in the Poisson distribution. Recall the Poisson distribution:

$$P(X = k) = \frac{\lambda^k e^{-\lambda}}{k!}$$

So,

$$\begin{aligned} P(X \leq 2) &= P(0) + P(1) + P(2) \\ &= e^{-3} + 3e^{-3} + \frac{9e^{-3}}{2} \end{aligned}$$

3 . Solve for $P(X \geq 2 | X \geq 1)$.

$$\begin{aligned} P(X \geq 2 | X \geq 1) &= \frac{P(X \geq 2) P(X \geq 1 | X \geq 2)}{P(X \geq 1)} \\ &= \frac{P(X \geq 2)}{P(X \geq 1)} \\ &= \frac{1 - 2e^{-1}}{1 - e^{-1}} = 0.418 \end{aligned}$$

3 Problem

For the Poisson process explicitly show that $\sum_{k=0}^{\infty} p_k = 1$ where $p_k = P(X = k)$. Write out the first few terms explicitly and recall the formula for the Taylor series expansion for e^x.

Answer

$$\begin{aligned}\sum_{k=0}^{\infty} P(X=k) &= \sum_{k=0}^{\infty} \frac{\lambda^k e^{-\lambda}}{k!} \\ &= e^{-\lambda} \sum_{k=0}^{\infty} \frac{\lambda^k}{k!} \\ &= e^{-\lambda}\left[1+\lambda+\frac{\lambda^2}{2!}+\frac{\lambda^3}{3!}+\frac{\lambda^4}{4!}+\cdots\right] \\ &= e^{-\lambda}\cdot e^{\lambda} = 1.\end{aligned}$$

Recall the Taylor series expansion:

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \cdots$$

4 Problem

This question deals with the Poisson process and is formulated in terms of a subway station. It could as well be formulated in terms of a networking problem. A subway station where different train lines intersect is like switch/router in the communication network with the different train lines corresponding to what are called Labelled Switched Paths (LSPs). As for the trains, you can think of each one as a packet or a burst of packets. One can also draw this analogy in optical networks which forms the core of the Internet backbone.

Consider that two one-way subway lines, the A train line and the B train line, intersect at a transfer station. The A trains and B trains arrive at the station according to independently operating Poisson processes with rates $\lambda_A = 3\ trains/hr$ and $\lambda_B = 6\ trains/hr$. We assume that passenger boarding and un-boarding occurs essentially instantaneously. [Note that the superposition of two Poisson processes with rates λ_1 and λ_2 is also a Poisson process with rate $\lambda_1 + \lambda_2$.]

1. What is $P(X = 9)$, the probability that the station handles exactly 9 trains during any given hour?
2. An observer arrives at the station at 8:00am. At the top of each subsequent hour (i.e., 9:00am, 10am, 11am, noon, etc) they record the number of trains the station has handled during that last hour. What is the expected number of hours they will need to wait until they first count exactly 9 trains arriving in an hour? (Note each hour is assumed to be independent of the previous hour and success is defined as exactly 9 trains arrive in an hour.)

Answers

- The combined process $N(t)$ of the A trains and B trains is a Poisson process with rate $\lambda_A + \lambda_B = 9$. The probability that there are exactly 9 trains in 1 hour is given by

$$\begin{aligned}P\{X=9\} &= e^{-\lambda}\frac{\lambda^9}{9!} \\ &= e^{-9}\frac{9^9}{9!} \\ &= 0.132\end{aligned}$$

- Let success correspond to the event if there are exactly 9 trains in any given hour starting on the hour. The probability of success denoted by p is 0.132 as computed above. The probability of failure, which corresponds to the event that the number of trains in a hour is any number other than 9, is $1 - p = 0.868$. Now each hour can be considered as a Bernoulli random variable. Furthermore, if we consider a sequence of hours we have a sequence of Bernoulli random variables which are independent. The independence comes from the fact that the underlying arrival process is Poisson. In a Poisson process the number of events in non-overlapping intervals are independent.

  The number of hours, (which is the number of trials) denoted by N until first success (observe exactly 9 trains) is geometrically distributed, i.e.,

  $$P\{N = k\} \;\; = \;\; p^{k-1}p \quad k = 1, 2, \ldots, \infty$$

  and we can show that the $E[N] = 1/p$. Thus, the expected number of hours is 7.6 hours. Since the observer only measures at the top of the hour, they will have to wait 8 hours on average before seeing 9 trains in one hour.

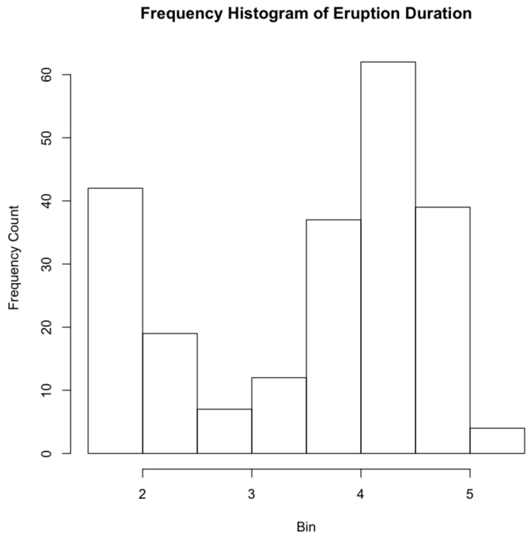
5 Problem

In lecture we discussed preliminary data analysis including the median, quantiles, quartiles, and boxplots using the inter-eruption time of the Old Faithful data. Use this data to answer the following questions concerning the eruption duration (the last column of the data set).

1. Plot the frequency histogram of the eruption duration with breaks of 1.
2. Draw the boxplot of the eruption duration, with the whiskers having the range=0.2 times the interquartile range.
3. What are the values for the 95, 97, and 99 quantiles of the eruption duration?
4. Suppose we classify the eruption duration using the following simple rule: if the duration is less than or equal to 3 mins then we classify it as a short eruption otherwise (i.e., if the duration is greater than 3 mins) it is a long eruption. Use the basic plot(x,y) function to draw a scatter plot that compares the current eruption duration (the x-axis) with the duration of the next eruption (the y-axis). In detail, suppose there are n data points $e[1] \ldots e[n]$. (You can find the number of data points using len(data[,3])). Then plot $(x = e[i], y = e[i+1])$ for $i = 1 \ldots n-1$. Note that both the x and y axes are in units time. On the scatter plot now draw a horizontal line at y=3 mins and a vertical line at x=3 mins. These two lines divide the area into 4 parts corresponding to long eruption followed by a long eruption, a long by a short, a short by a long, and short by a short.
5. Use R to analyze the data to estimate the probabilities that:
   - a long eruption is followed by a long eruption
   - a long eruption is followed by a short eruption
   - a short eruption is followed by a long eruption
   - a short eruption is followed by a short eruption

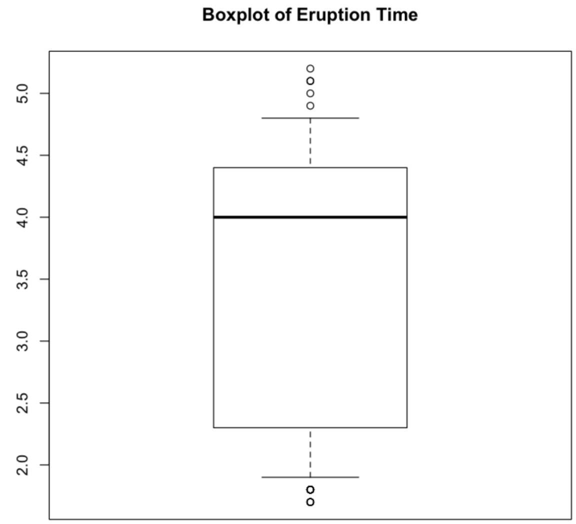
Answer

```
1) hist(data[,3], xlab ="Duration", ylab ="Frequency Count", main = "Eruption Durat
```



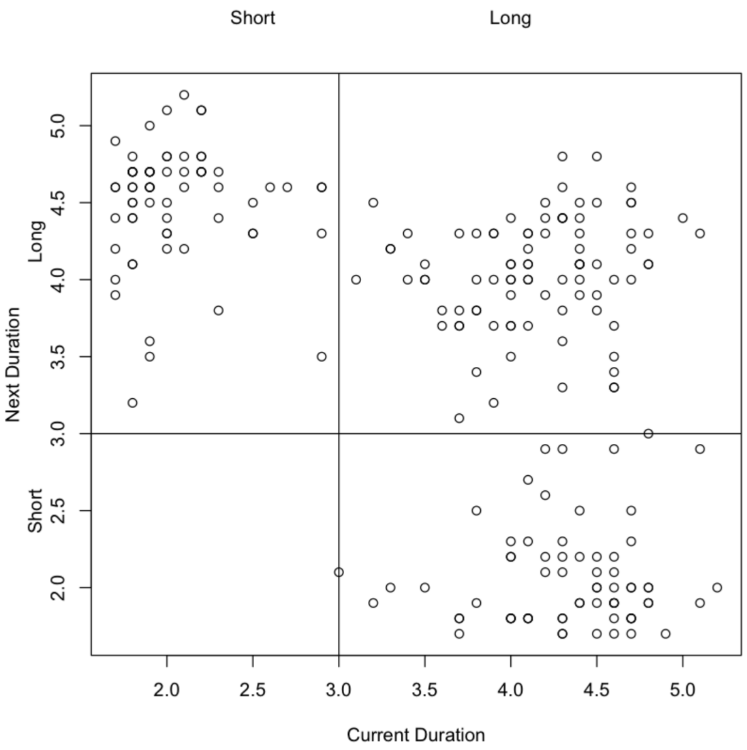
```
# Note the default hist function produces breaks of size 1 for this data set.
```

```
2) boxplot(data[,3], range=0.2, main = "Boxplot of Eruption Time")
```



```
3) q = c(.95, .97, .99)
quantile(data[,3], q)

Returns:
0.96      4.8
```

```
0.97      4.8
0.99      5.1

4 & 5)

data = read.table(file="./Data/Old_Faithful.txt", header=TRUE)

n = length(data[,1])
currentd = numeric(n-1)
nextd = numeric(n-1)
pss = 0
psl = 0
pll = 0
pls = 0

for (i in seq(1,n-1,1)) {
  currentd[i] = data[i,3]
  nextd[i] = data[i+1,3]
  if ((currentd[i] >= 3.0) & (nextd[i] >= 3.0))
      {pll = pll + 1}
    else if ((currentd[i] >= 3.0)  && (nextd[i] < 3.0))
        {pls = pls + 1}
    else if ((currentd[i] < 3.0)  && (nextd[i] < 3.0))
        {pss = pss + 1}
    else
        {psl = psl + 1}
}

plot(currentd,nextd, xlab = "Current Duration", ylab = "Next Duration")
mtext(c("Short","Long"),side=3,line=2,at=c(2.5,4))
mtext(c("Short","Long"),side=2,line=2,at=c(2.5,4))
abline(v=3)
abline(h=3)
```



```
5) see code above and then:

sprintf("Probability of Long Long is %f", pll/(n-1))
sprintf("Probability of Long Short is %f",pls/(n-1))
sprintf("Probability of Short Long is %f", psl/(n-1))
sprintf("Probability of Short Short is %f", pss/(n-1))

Returns:
Probability of Long Long is 0.398190
Probability of Long Short is 0.303167
Probability of Short Long is 0.298643
Probability of Short Short is 0.000000
```