ECS 132 - Homework #2

Bishal Thapa,
bthapa@ucdavis.edu

2018-04-27

Contents

Problem 1

On a multiple-choice exam with 3 possible answers for each of the 5 questions, what is the probabilitythat a student will get 4 or more correct answers just by guessing?

$$
\begin{aligned}
P(X \geq 4) &= P(X = 4) + P(X = 5) \\
&= \binom{5}{4}(\frac{1}{3})^4(1-\frac{1}{3})^1 + \binom{5}{5}(\frac{1}{3})^5(1-\frac{1}{3})^0 \\
&= 0.0453
\end{aligned}
$$

Problem 2

We consider a noisy communication link in which the message is encoded in to binary digits (0,1)(bits) before being transmitted. We will denote the length of the encoded message byn. Since thechannel is noisy, the bits can get flipped; a 0 to 1 or a 1 to 0. We will assume that each bit isflipped independently with probabilityp. In order to be able to detect that the received message isin error, a simple method is to add a parity bit at the transmitter. There can two types of parity- even parity and odd parity. If even (odd) parity is used , the parity bit is set such that totalnumber of 1s in the encoded message is even (odd). For the sake of this problem, we will onlyconsider even parity.

2.1

Suppose $n = 7$ and $p = 0.1$, what is the probability that the received message has errors which go undetected? Since we are considering even parity. Errors that go undetected still contain an even parity greater than 0 but the wrong bits.

$$
\begin{aligned}
P(X\ is\ even) &= P(X = 2) + P(X = 4) + P(X = 6) \\
&= \binom{7}{2}(0.1)^2(1-0.1)^5 + \binom{7}{4}(0.1)^4(1-0.1)^3 + \binom{7}{6}(0.1)^6(1-0.1)^1 \\
&= 0.127
\end{aligned}
$$

2.2

For general n and p, write down an expression (as a sum) for the probability that the receivedmessage has errors which go undetected.

$$
\begin{aligned}
P(X\ is\ even) &= \sum_{i \in even, x \geq 2}^{n} P(X = i) \\
&= \sum_{i \in even, x \geq 2}^{n} \binom{n}{i}(0.1)^i(1-0.1)^{n-i}
\end{aligned}
$$

Problem 3

Consider the following program statement consisting of a while loop \$ while $\neg$B do S\$ Assume that the Boolean expression B takes the value true with probability p and the value false with probability q. Assume that the successive test on B are independent.

3.1

Find the probability that the loop will be executed k times.

$$P(X = k) = pq^{k-1}$$

3.2

Find the expected number of times the loop will be executed.

$$\begin{aligned} E(X) &= \sum_{i=1}^{\infty} pq^i \\ &= \frac{pq}{1-q} \end{aligned}$$

3.3

Considering the same above assumptions, suppose the loop is now changed to “repeat S until B” What is the expected number of times that the repeat loop will be executed?
It will run the same number of times the previous loop will run except 1 more loop because the condition is checked after the loop has run once rather than checking before starting the loop.

Problem 4

Suppose that we want to generate the outcome of the flip of a fair coin, but that all we have at our disposal is a biased coin that lands on heads with unknown probability p that need not be equalto 1/2. Consider the following procedure
1. Flip the coin
2. Flip the coin again
3. If both flips land on heads or both lands on tails, return to step 1.4.
Let the result of the last flip be the result of the experiment.

Since we are only worried about two rolls at a time and if HH or TT appears, the procedure is reset, $sample\ space = HT, TH$.

$$\begin{aligned} P(ending\ H) &= \frac{P(TH)}{P(TH) \cup P(HT)} \\ &= \frac{p(1-p)}{p(1-p)+p(1-p)} \\ &= \frac{1}{2} \\ P(ending\ T) &= \frac{P(HT)}{P(TH) \cup P(HT)} \\ &= \frac{p(1-p)}{p(1-p)+p(1-p)} \\ &= \frac{1}{2} \end{aligned}$$

Problem 5

The following problem is called the coupon collector problem and has many applications in computer science. There are N different types of coupons. Each time one obtains a coupon it is independent of the previous selection and equally likely to be any of the N types. Let T denote the random variable that denotes the coupons that needs to be collected until one obtains a complete set of at least one of each type. Write a R simulation code to compute the E[T] for N= 20,30,50.

```
couponSim <- function(N){
  coupons <- numeric(N)
  max <- 10000
  for(i in 1:max){
    x <- sample.int(N,1,replace=TRUE)
    coupons[x] <- 1
    if(sum(coupons)==N){
      return(i)
    }
  }
}
sim20 <- couponSim(20)
sim20
```

```
## [1] 61
```

```
sim30 <- couponSim(30)
sim30
```

```
## [1] 75
```

```
sim50 <- couponSim(50)
sim50
```

```
## [1] 169
```

Problem 6

Suppose a bit stream is subject to errors, with each bit having probability p of error, and with the bits being independent. Consider a set of four particular bits. Let X denote the number of erroneous bits among those four.
1. Find $P(X = 2)$ and $E(X)$. 2. What famous parametric family of distributions does the distribution of X belong to? 3. Let Y denote the maximum number of consecutive erroneous bits. Find $P(Y = 2)$ and $Var(Y)$.

6.1

$$P(X = 2) = \binom{4}{2} p^2 (1 - p)^2$$
$$E(X) = n * p = 4p$$

6.2

Binomial distribution

6.3

$$
\begin{aligned}
P(Y=2) &= 1 \textit{ consecutive error bit} + 2 \textit{ consecutive error bits} \\
&= 2p^3(1-p) + 3p^2(1-p)^2 \\
Var(Y) &= 4p(1-p)
\end{aligned}
$$

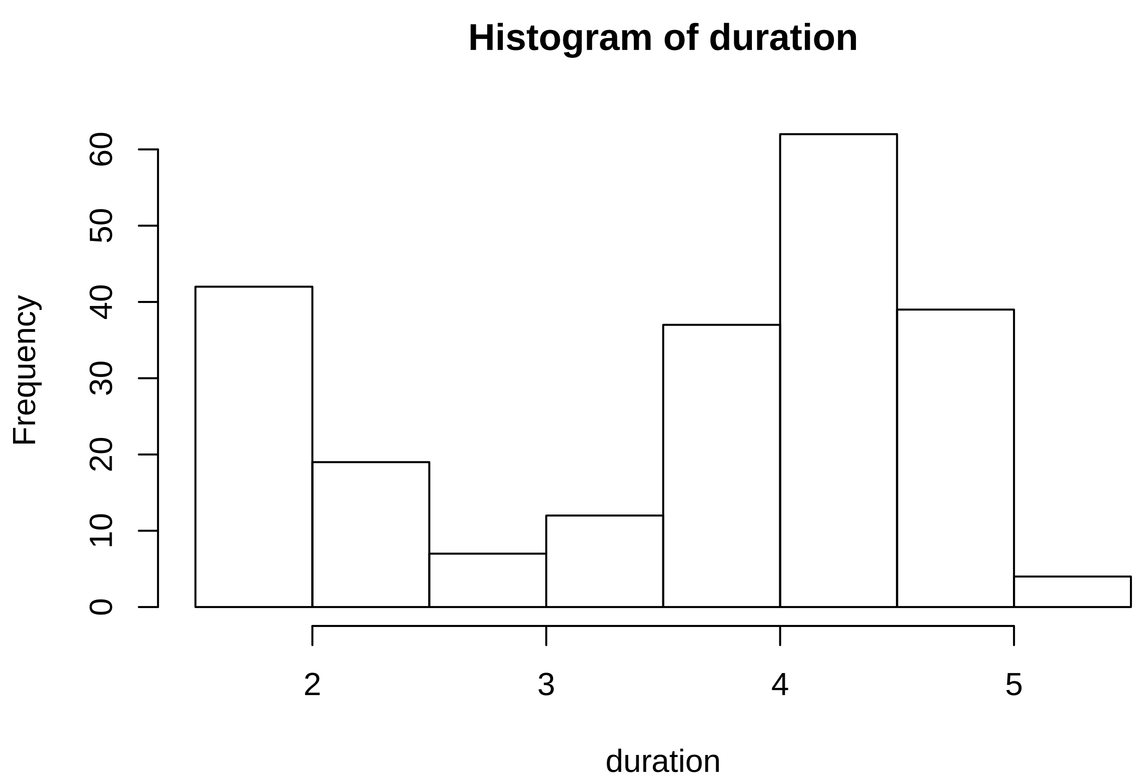
Problem 7

In lecture we discussed preliminary data analysis and used the inter-erruption time of the OldFithfuldata. Answer the following questions for the erruption duration (the last column of the data set)
1. Plot the histogram of the erruption duration?
2. Draw the boxplot of the erruption duration?
3. What are the values for the 95, 97, 99 quantiles of the erruption duration?
4. Suppose we classify the erruption duration using the following simple rule: if the durationis less than or equal to 3 mins then we classify it as a short erruption otherwise (i.e., of ifthe duration is greater than 3 mins) it is a long erruption. Find the probabilities that a long erruption is followed by a long erruption, a long by a short, a short by a long, and short by a short,

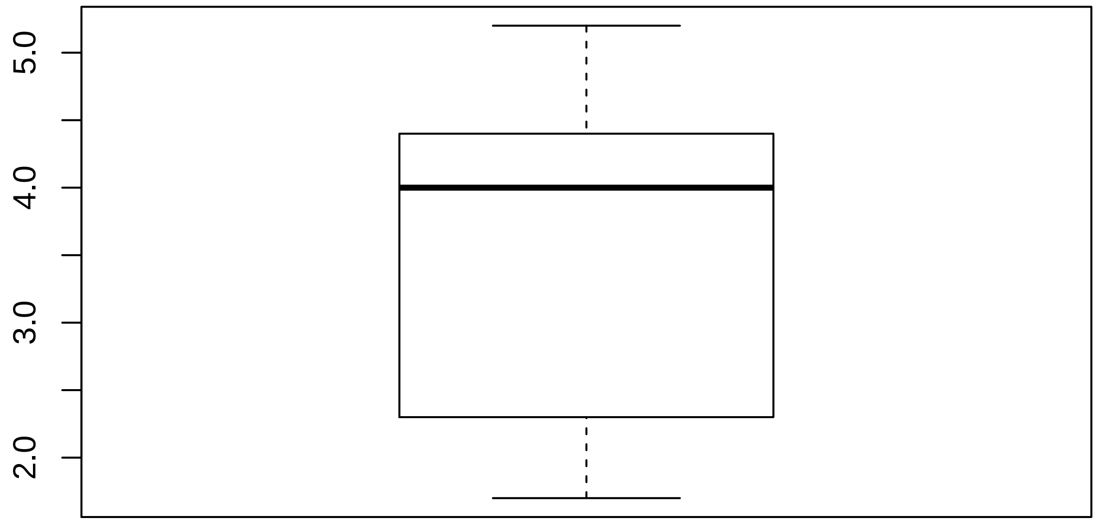
```r
faithful <- read.table("Old_faithful.txt", header=TRUE)

duration <- faithful$duration

hist(duration, xlab="duration")
```



```r
boxplot(duration)
```




```r
quantile(duration,probs=c(0.95,0.97,0.99))
```

```
## 95% 97% 99%
## 4.8 4.8 5.1
```

```r
longCounter <- 0
shortCounter <- 0
for(i in 1:length(duration)){
  a <- duration[i]
  b <- duration[i+1]

  if(!(is.null(b))){
    if(a > 3 & b>3){
      longCounter = longCounter + 1
    }
    if(a<= 3 | b<=3){  # doesn't want to work
      shortCounter = shortCounter + 1
    }
  }
}
shortProb <- shortCounter / length(duration)
longProb <- longCounter / length(duration)
shortProb
```

```
## [1] 0.6081081
```

```
longProb
```

```
## [1] 0.3918919
```

Classmate Collaborators

Inclulde the names and email IDs of everyone you collaborated with. You are free to discuss with your peers but everyone's work must be individual.

1. Brendan Gerrity, bgerrity@ucdavis.edu