Exam 1

Ian McGroarty

March 15, 2020

Problem 1

Let X and Y have the joint density:

$$f(x, y) = \frac{6}{7}(x + y)^2$$

(a) Calculate P(X>Y)

$$P(x > y) = 1 - P(x < y) = 1 - \int_0^1 \int_0^x \frac{6}{7}(x + y)^2 dydx = 1 - \frac{6}{7}\int_0^1 \int_0^x x^2 + 2xy + y^2 dydx$$

$$= 1 - \frac{6}{7}\int_0^1 [x^2y + xy^2 + \frac{1}{3}y^3 \,|_0^x] = 1 - \frac{6}{7}\int_0^1 \frac{7}{3}x^3 dy$$

$$= 1 - \frac{6}{7}[\frac{7}{12}x^4 \,|_0^1] = \frac{1}{2}$$

(b) Calculate $P(X < \frac{1}{2})$

$$f_x(x) = \int_0^1 f_{x,y}(x, y)dy = \int_0^1 \frac{6}{7}(x + y)^2 dy = \frac{6}{7}\int_0^1 x^2 + 2xy + y^2 dy$$

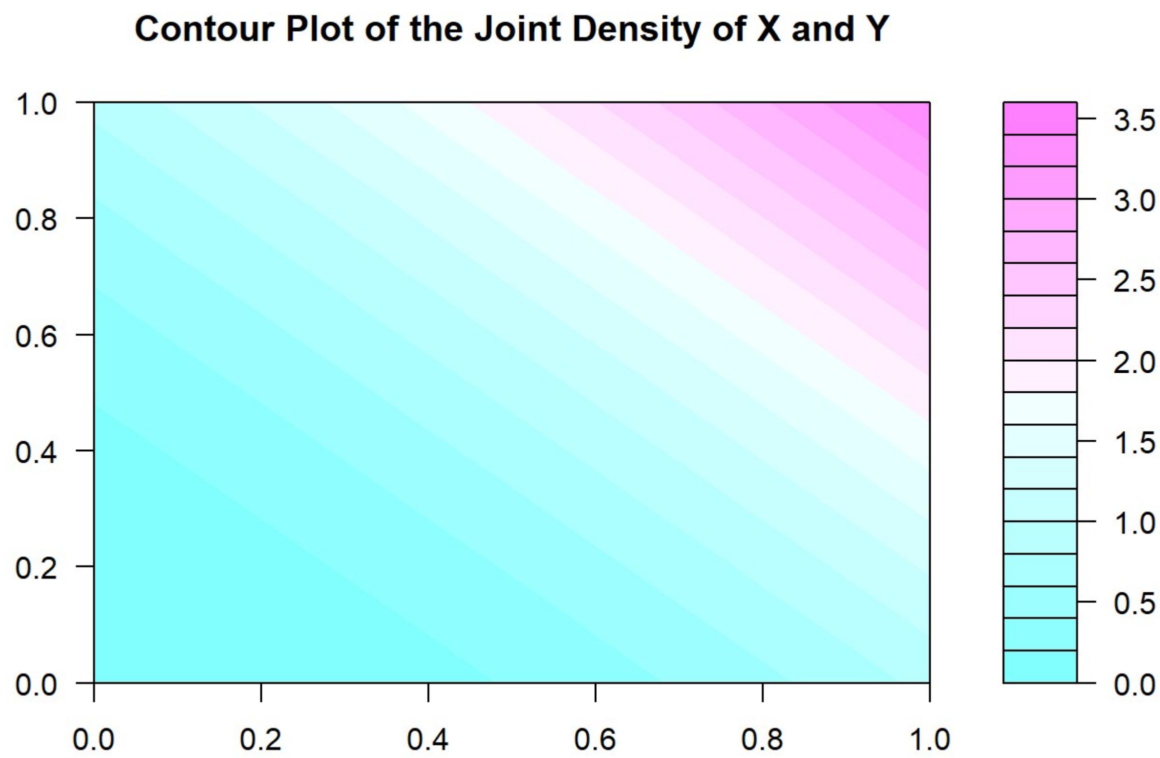
$$= \frac{6}{7}(x^2y + xy^2 + \frac{1}{3}y^3 \,|_0^1) = \frac{6}{7}(x^2 + x + \frac{1}{3})$$

$$P(x < \frac{1}{2}) = \int_0^{\frac{1}{2}} f_x(x)dx = \int_0^{\frac{1}{2}} \frac{6}{7}(x^2 + x + \frac{1}{3})$$

$$= \frac{6}{7}(\frac{1}{3}x^3 + \frac{1}{2}x^2 + \frac{1}{3}x \,|_0^{\frac{1}{2}}) = 0.2857$$

(c) Find the conditional density of X given Y

$$P(X|Y) = \frac{f_{x,y}(x, y)}{f_y(y)} = \frac{\frac{6}{7}(x + y)^2}{\frac{6}{7}(y^2 + y + \frac{1}{3})} = \frac{x^2 + 2xy + y^2}{y^2 + y + \frac{1}{3}}$$



Problem 2

Let X be a continuous random variable with density function below. Let $S = X_1 + X_2 + ... + X_{30}$ where X_i are random varables with density f. What is $P(\frac{S}{30} \geq 0.2)$?

$$f_X(x) = \frac{3}{2}x^2 \qquad -1 \leq x \leq 1$$

Solution

I'm going to rework the S to be $S = \frac{1}{30}(X_1 + \cdots + X_{30})$. This means that the probability can be $P(S \geq 0.2)$ and we can apply the central limit theorem. I imagine you can complete the problem without doing this but I was having trouble.

$$E(X) = \int_{-1}^{1} \frac{3}{2}x^2 \cdot x \cdot dx = \frac{3}{8}x^4 = 0$$

$$Var(x) = E(X^2) - E(X)^2 = \int_{-1}^{1} \frac{3}{2}(x^2) \cdot x^2 \cdot dx - 0$$

$$= \int_{-1}^{1} \frac{3}{2}x^4 \cdot dx = \frac{3}{10}x^5 |_{-1}^{1} = \frac{6}{10} = 0.6$$

$$P(S \geq 0.2) = 1 - P(S \leq 0.2) = 1 - P(S \leq 0.2)$$

$$S \sim N(0, \sqrt{0.6}/\sqrt{30}) \text{ By the Central Limit Theorem}$$

$$1 - P(S \leq 0.2) = 1 - P(\frac{s-0}{\sqrt{0.6}/\sqrt{30}} \leq \frac{0.2-0}{\sqrt{0.6}/\sqrt{30}})$$

$$= 1 - P(Z \leq 1.414) = 0.079$$

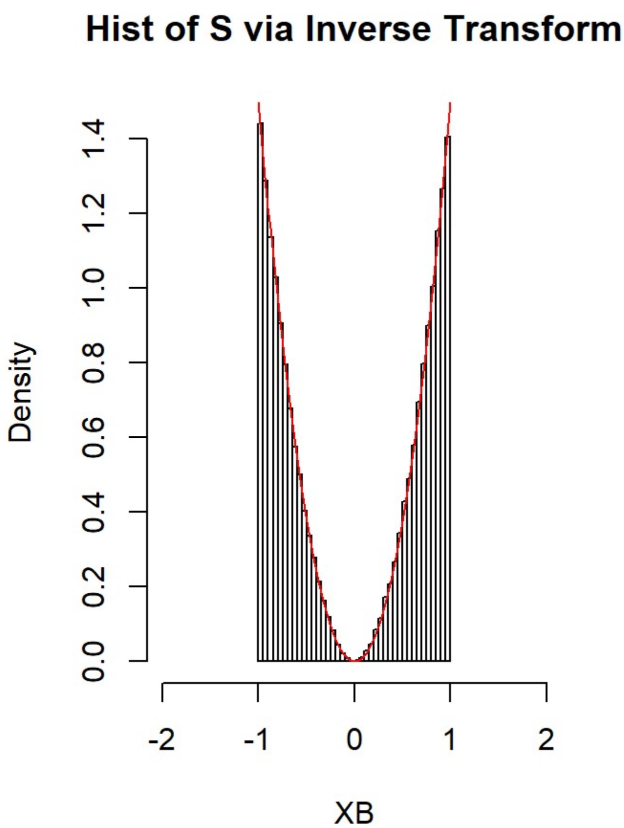
Solution part 2

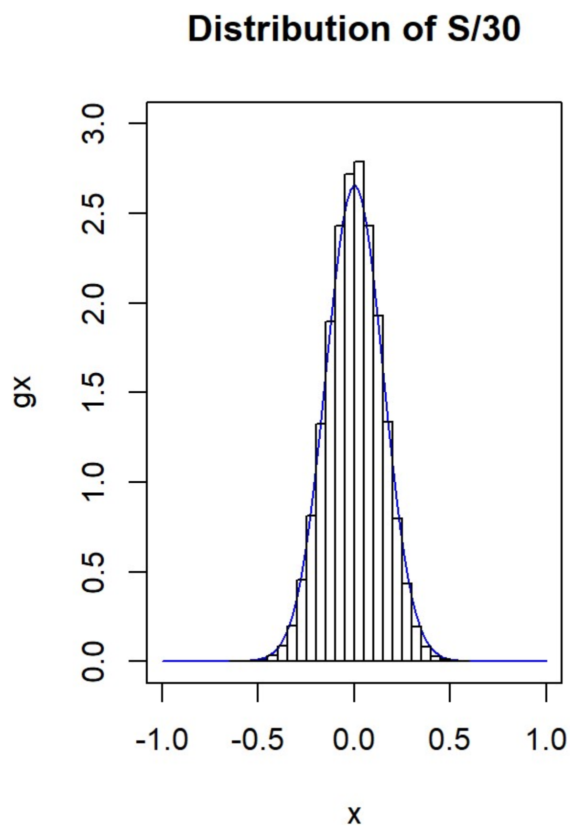
Okay so I struggled with this problem because it took me awhile to figure out to apply the central limit theorm. While I was working on it I decided to use R to try and approximate the probability. First I use the inverse transform method to generate random variables that match the distribution. Then I create Nsim interations of $S = X_1 + \cdots X_{30}$ calculate the percent that are greater than or equal to 0.2. And in doing so I realized that it is a normal distribution at which point I remembered the central limit theorem. But I just thought I'd share. You can see how close the probablility is to the theoretical probability and how well the normal distribution approximates S/30.

```r
Nsim <- 100000

## Using Inverse Transform Lets draw from fx
  # Define the function
    fx <- function(x){
      (3/2)*x^2
    }
  # Get Uniform
    U <- runif(Nsim,min=-1,max=1)
    a <- (U)
  # R can't do negative numbers to off fractional roots so I have to do it as an ifels
e
    XB <- (ifelse(a>=0,(a)^(1/3),(-1)*((-1*a)^(1/3))))

## Create Nsim iterations of S
  # Create S
    s <- c()
      for (i in  1:Nsim) {
        U <- runif(30,min=-1,max=1)
        a <- (U)
        X <- ifelse(a>=0,a^(1/3),(-1)*((-1*a)^(1/3)))
        s[[i]] <- sum(X)
      }
  # Probability
    (sum((s/30)>=0.2))/Nsim
```

```
## [1] 0.07727
```

```r
  # Plot the normal distribution
      gx <- function(x){
        (dnorm(x,0,0.15))
      }
## The Plots
  par(mfrow = c(1,2))

  hist(XB,freq=F,xlim=c(-2,2),breaks = 30,main="Hist of S via Inverse Transform")
  plot(fx,xlim=c(-1,1),ylim=c(0,2),col = "red",add=TRUE)

  plot(gx,xlim=c(-1,1),ylim=c(0,3),col="blue", main = "Distribution of S/30")
  hist((s/30),freq = FALSE,add=TRUE)
```





Problem 3

Consider a random experiment where we draw uniformly and independently n numbers $X_1 \cdots X_n$ from the interval [0,1]. Let $X_{(n)}$ be the largets of the n numbers. Determine the pdf of $X_{(}n)$

Solution

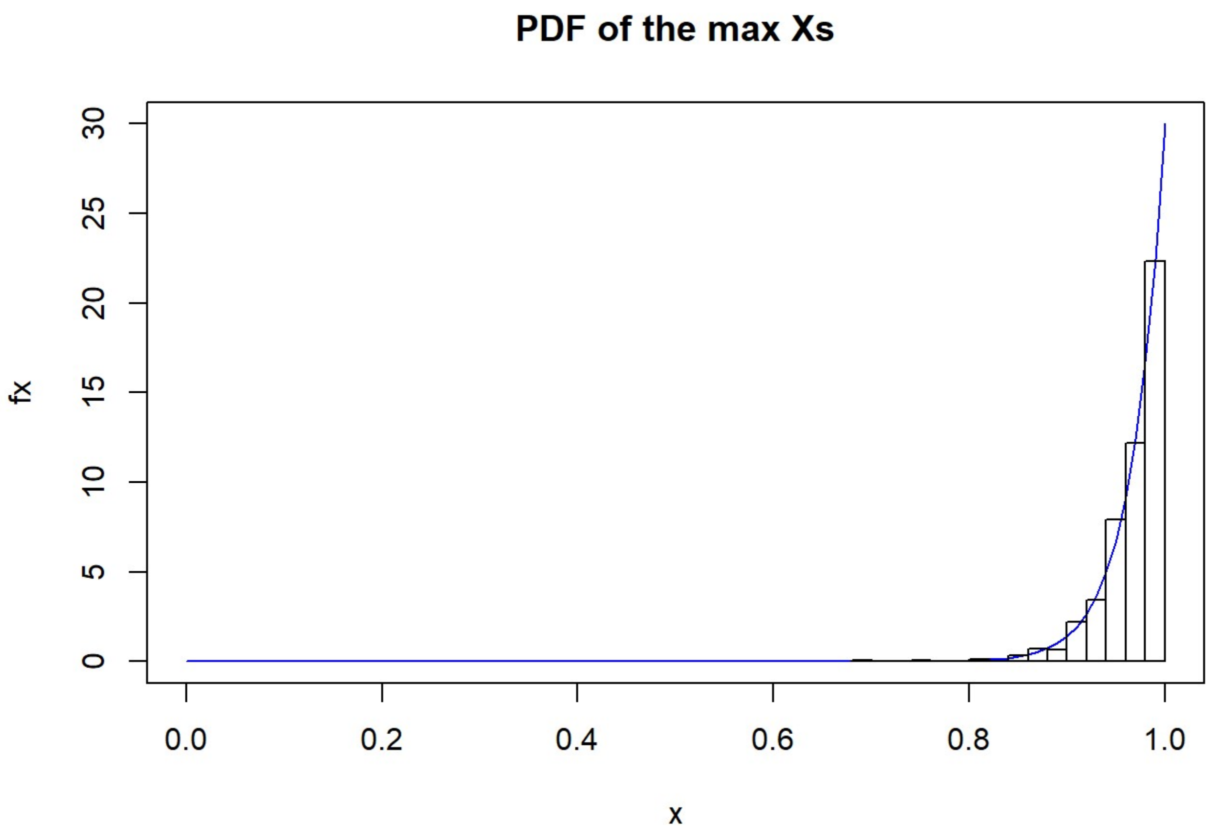
Consider the case that X_i is the max we are interested in. Then we need X_i to be greater than all X_n and since the trials are independent the individual probabilities are multiplicative. Since they are uniform each individual pdf is 1 and the individual CDFs are just x.

$$
\begin{aligned}
P(X_i = X_{max}) &= P(X_1 \leq X_i) \cap P(X_3 \leq X_i) \cap \ldots \cap P(X_n \leq X_i) \\
&= P(X_2 \leq X_1) \cdot P(X_3 \leq X_i) \cdot \ldots \cdot P(X_n \leq X_i) \\
&= P(X_n \leq X_i)^n \\
&= (F_X(x))^n \\
f_{Xi}(x) &= f_x \cdot nF_X^{n-1} \\
X &\sim Unif(0, 1) \\
f_{Xi}(x) &= n \cdot x^{n-1}
\end{aligned}
$$

Solution part 2

Thankfully enough, we can test this with R. I drew a list of n uniformly distributed xs and found the max. Iterating over Nsim trials I can get a pdf of $X_{(n)}$. The histogram matches up well with the calculated pdf represented by the blue line.

```
n <- 30
Nsim <- 1000

## Take the max of n Independent Uniform Xs Nsim times
  s <- c()
  for (i in 1:Nsim) {
  x <- runif(n,min=0,max=1)
  s[i] <- max(x)
  }
## Plot the theoretical pdf of X
  fx <- function(x){
    n*x^(n-1)
  }

## Plot
  plot(fx, col = "blue", main = "PDF of the max Xs")
  hist(s,freq=FALSE,add=TRUE)
```



Problem 4

To make a crude model of a forest we might introduce states 0= grass, 1 = bushes, 2 = small trees, 3 = large trees, and write down a transition matrix. The idea behind this matrix is that, if left undistured, a grassy area will see bushes grow, then small trees which grow into large trees. Then everything changed when the fire nation attacked and reset the system to state 0.Only the avatar, master of all 4 elements could find the fraction of land of each of these states.

Solution 4

All we need to do is find the limiting distribution by raising the transition probability matrix to a high power. We can see that this is stable since the probability of moving to any state is the same regarless of the current state. In the long run 10% of the land is grass, 40% is bushes, 30% is small trees, and 20% is large trees.

```
x <- matrix(c((1/2),(1/24),(1/36),(1/8),
              (1/2),(7/8),(0),(0),
              (0),(1/12),(8/9),(0),
              (0),0,(1/12),(7/8)),nrow=4)
x %^% 1000
```

```
##      [,1] [,2] [,3] [,4]
## [1,]  0.1  0.4  0.3  0.2
## [2,]  0.1  0.4  0.3  0.2
## [3,]  0.1  0.4  0.3  0.2
## [4,]  0.1  0.4  0.3  0.2
```

Problem 5

Brands A and B have consumer loyalties of 0.7 and 0.8 meaning that a customer who buys A one week will buy it again with probability 0.7.

(a) What is the limiting market share for A and B

The code below calculates this in two ways. First similarly to question 4, we can raise the transition probability matrix to a high power. The other way to do this is, since the matrix is pretty clearly ergodic we can use the eigen vector for eigen value 1 to find the limiting distribution. Both ways show that 40% of consumers use A and 60 % of consumers use B.

```
## The transition matrix is
x <- matrix(c(0.7,0.2,0.3,0.8),nrow=2)
as.matrix(x)
```

```
##      [,1] [,2]
## [1,]  0.7  0.3
## [2,]  0.2  0.8
```

```
## The long run "limiting market share" is
  x %^% 1000
```

```
##      [,1] [,2]
## [1,]  0.4  0.6
## [2,]  0.4  0.6
```

```
## You can calculate it more elegantly by taking the eigen vector for eigen values 1
  evecs <- as.matrix(eigen(x)$vectors)
  evecsb <- evecs * 1/evecs[1,1]
  evecsinv <- Inverse(evecsb)
  evecsinv
```

```
##          [,1]      [,2]
## [1,] 0.400000  0.600000
## [2,] 0.509902 -0.509902
```

(b) Third Brand

Suppose now there is a third brand with loyalty 0.9 and that a consumer who changes brands picks one of the other two at random. What is the new limiting market share for these three products?

The code below shows that the limiting market distribution is: 18% use A, 27% use B, and 55% use C.

```
## The transition matrix is
  x <- matrix(c(0.7,0.1,0.05,0.15,0.8,0.05,0.15,0.1,0.9),nrow=3)
  as.matrix(x)
```

```
##      [,1] [,2] [,3]
## [1,] 0.70 0.15 0.15
## [2,] 0.10 0.80 0.10
## [3,] 0.05 0.05 0.90
```

```
#The limiting distribution is:
  x %^% 1000
```

```
##           [,1]      [,2]      [,3]
## [1,] 0.1818182 0.2727273 0.5454545
## [2,] 0.1818182 0.2727273 0.5454545
## [3,] 0.1818182 0.2727273 0.5454545
```

Problem 6

Customers arrive at a bank according to a Poisson process with rate 10 per hour. Given that two customers arrived in the first 5 minutes, what is the probability that:

(a) Both arrived in the first two minutes?

$$\begin{aligned} P(A \,\&\, B \leq 2 \mid A, B \leq 5) &= P(X_{2/60} = 2 \mid X_{5/60} = 2) = P(X_{2/60} = 2) \cdot P(X_{3/60}) / P(X_{5/60} = 2) \\ &= \frac{e^{-10 \cdot \frac{2}{60}} \cdot (10 \cdot \frac{2}{60})^2}{2!} \cdot \frac{e^{10 \cdot \frac{3}{60}} \cdot (10 \cdot \frac{3}{60})^0}{0!} / \frac{e^{10 \cdot \frac{5}{60}} \cdot (10 \cdot \frac{5}{60})^2}{2!} \\ &= \frac{e^{-1/3} \cdot (1/3)^2}{2!} \cdot \frac{e^{-1/2} \cdot (1/2)^0}{0!} / \frac{e^{-5/6} \cdot (5/6)^2}{2!} \\ &= 0.16 \end{aligned}$$

(b) At least one arrives in the first two minutes.

$$\frac{P(X_{2/60}=2) \cup P(X_{2/60}=1)}{P(X_{5/60}=2)}$$
$$= \frac{P(X_{2/60}=2) \cdot P(X_{3/60}=0) + P(X_{2/60}=1) \cdot P(X_{3/60}=1)}{P(X_{5/60}=2)}$$
$$= (\frac{e^{-1/3} \cdot (1/3)^2}{2!} \cdot \frac{e^{-1/2} \cdot (1/2)^0}{0!}) + (\frac{e^{-1/3} \cdot (1/3)^1}{1!} \cdot \frac{e^{-1/2} \cdot (1/2)^1}{1!}) / \frac{e^{-5/6} \cdot (5/6)^2}{2!}$$
$$= 0.64$$

Solution part 2

As with the others I've developed an R program to show that this is indeed the probability. I also wrote a formula to do the calculations for me. As we can see the proabilities are fairly close (if not exactly) the calculated probabilities.

```
## Poisson process formula to aid with calculations:
  pois.process <- function(x,n){
    exp(-x)*(x^n)/factorial(n)
  }

## Create a data frame where each column represents one minute that has a poisson proc
ess of 10*(1/60)=1/6
  # Each row represents one possibility for some hour
    DF <- data.frame(matrix(NA,nrow=Nsim,ncol=60))
    for (i in 1:Nsim){
      DF[i,] <- (rpois(60,(1/6)))
      }
  # Identify the rows that 2 people show up in the first 5 minutes.
    DF$five <- rowSums(DF[,1:5])
      # sum(DF$five==2)/Nsim    # - how often do 2 people show up in the first five mi
nutes
      # pois.process((5/6),2)   # - Matches the theoretical number

    cond <-  dplyr::filter(DF,DF$five==2)
  # Identify how many people show up in the first two minutes in the case that 2 showe
d up in five minutes
    cond$two <- rowSums(cond[,1:2])
## For Part A
    # How often do two people show up in the first five minutes (if 2 showed up in the
first 5 minutes)
     (sum(cond$two == 2))/nrow(cond)
```

```
## [1] 0.1617647
```

```
     (pois.process((1/3),2)*(pois.process((1/2),0))/pois.process((5/6),2))
```

```
## [1] 0.16
```

```
## For Part B
    # How often does at least one person show up in the first five minutes (if 2 showe
d up in the first 5 minutes)
      (sum(cond$two %in% c(1,2)))/nrow(cond)
```

```
## [1] 0.6617647
```

```
      (pois.process((1/3),2)*pois.process(.5,0)+pois.process((1/3),1)*pois.process(.5,
1))/(pois.process((5/6),2))
```

```
## [1] 0.64
```

Problem 7

Give the most efficient method possible to generate a rowndom variable having density function:

$$f(x) = \begin{cases} \frac{x-2}{2} & 2 \le x \le 3 \\ \frac{2-x/3}{2} & 3 \le x \le 6 \end{cases}$$

Solution part 1

I believe this is a problem about adaptive accept sampling. But I never really got the hang out it. So then I tried the inverse transform sampling which would be the most efficient way of doing it. I had a lot of trouble inverting the function, in fact I'm nearly certain it isn't invertable. However, just in case, I demonstrate the calculation used to get as far as I did. In a feeble attempt at getting some points for this problem I've done a typical accept reject algorithm in R using the normal distribution. The acceptance rate is around 50% which isn't horrible.

$$F(X) = \begin{cases} 2 \le x \le 3 & \int_2^x \frac{t-2}{2} = \frac{1}{2} dt = \frac{1}{2}[\frac{1}{2}t^2 - 2t|_2^x] \\ & = \frac{1}{2}[\frac{1}{2}x^2 - 2x] - \frac{1}{2}[\frac{1}{2}(2)^2 - 2(2) \\ & = \frac{1}{4}x^2 - x - 1 \\ 3 \le x \le 6 & \int_2^3 \frac{t-2}{2} + \int_3^x \frac{2-t/3}{2} = \frac{1}{2}[\frac{1}{2}t^2 - 2t|_2^3] + \frac{1}{2}[-\frac{1}{6}t^2 + 2t|_3^x] \\ & = \frac{x^2}{12} + x - 2 \end{cases}$$

Try To invert $F(X)$ but failed.

$$F^{-}(x) = \begin{cases} 2 \le x \le 3 & \begin{aligned} U &= (1/4)x^2 - x - 1 \\ 4(U+1) &= x^2 - 4x - 1 \\ x &= \ ? \end{aligned} \\ 3 \le x \le 6 & \begin{aligned} U &= \frac{x^2}{12} + x - 2 \\ 12(U+2) &= x^2 + 12x \\ x &= \ ? \end{aligned} \end{cases}$$

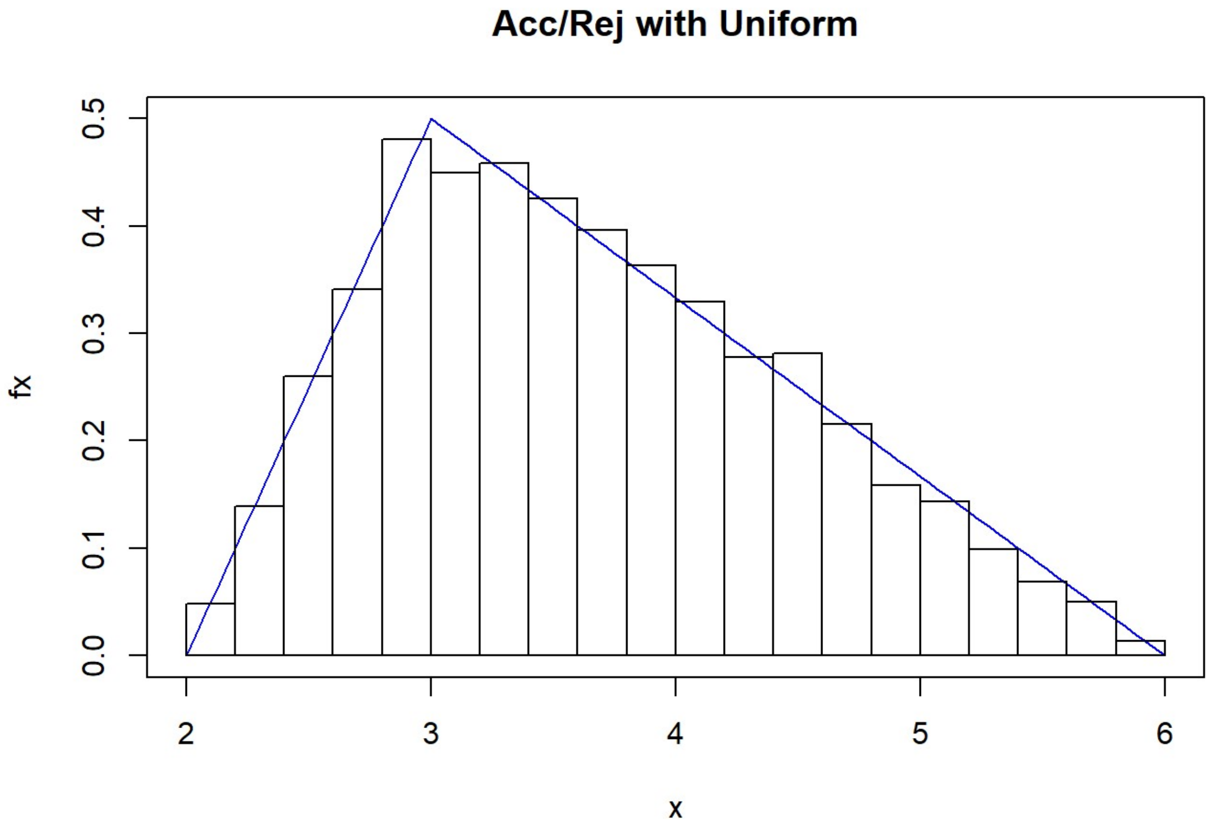
Solution part 2

Okay so back to basics. I know I can use the Accept reject method. I feel like it is best explained in the code so…

```
## Set the function
    fx <- function(x){
      ifelse(x>=2 & x<3,(x-2)/2,(2-x/3)/2)
    }

## Accept Reject with uniform
  Nsim = 10000
  # Generate x candidates over the range under condiseration
    Xcand <- runif(Nsim,min=2,max=6)
  # Generate y candidates up to M
    Ycand <- runif(Nsim, min=0, max=0.5)
  # Keep the x candidates for which the Y candidates are viable solutions
    XB <- Xcand[Ycand < fx(Xcand)]
  # Acceptance Rare
    length(XB)/Nsim
```

```
## [1] 0.5063
```

```
  # Plots
    plot(fx,xlim=c(2,6), col="blue",main="Acc/Rej with Uniform")
    hist(XB,freq=FALSE,add=TRUE)
```



Problem 8

Generate, in as efficient a way as possible, 10000 values of X where $X \sim f_X(x)$ and

$$f(X) = 30 \cdot (x^2 - 2x^3 + x^4)\ s.t.\ \ 0 \leq x \leq 1$$

Solution

Okay this is mostly coding but I tried a number of different ways to do this problem: I quickly realized that I could not hope to invert the CDF. Maybe some people can but I could not. I moved on to Accept reject Method. Using a proposal distribution $g(x) \sim N(0.5, 0.2)$ seemed to give me the best acceptance rate of about 85%. Moreover, you can see in the figure how well the normal distribution approximates f(x). I'll also note that Metropolis Hastings is literally always worse so I didn't bother with that one.

```
Nsim <- 10000

## Set Up
  # Set the function
    fx <- function(x){
      30*(x^2-2*x^3+x^4)
      }
  # Find the Max
    Xcand <- runif(Nsim,min=0,max=1)
    M <- max(fx(Xcand))

## Approx With Normal Distribution
    # Find the Approriate mean and variance for the cumulative normal distribution
      gx <- function(x){
        (dnorm(x,0.5,0.2))
      }

    # Find normalizing constant s.t. g(x)*M > f(x) for all x
      ## Set hx = fx/gx
        hx <- function(x){
          fx(x)/gx(x)
        }
      ## Find the Max of hx(x)
        hx.opt <- optimise(hx,interval =c(0,1),maximum=TRUE)$maximum
      ## Get the Constant M = hx(hx.opt) => fx < gx*M
        M <- hx(hx.opt)

      # Redefine gx do include the M
        gx2 <- function(x){
          M*(dnorm(x,0.5,0.2))
        }

    # X Candidates from Normal
      ## I can't figure out how to set a range with the rnorm command so this is what
I came up with
        counter <- 0
        Xcand <- c()
        while(counter<Nsim) {
          z <- 1*rnorm(1,0.5,0.2)
          if (z <= 1 & z>=0){
            Xcand <- append(Xcand,z)
            counter = counter + 1
          }
        }
    # Y Candidates from Uniform up to M*gx(Xcand)
      Ycand <- runif(Nsim,min=0,max=(gx2(Xcand)))
    # Accept or Reject
      XB <- Xcand[(Ycand < (fx(Xcand)))]
```

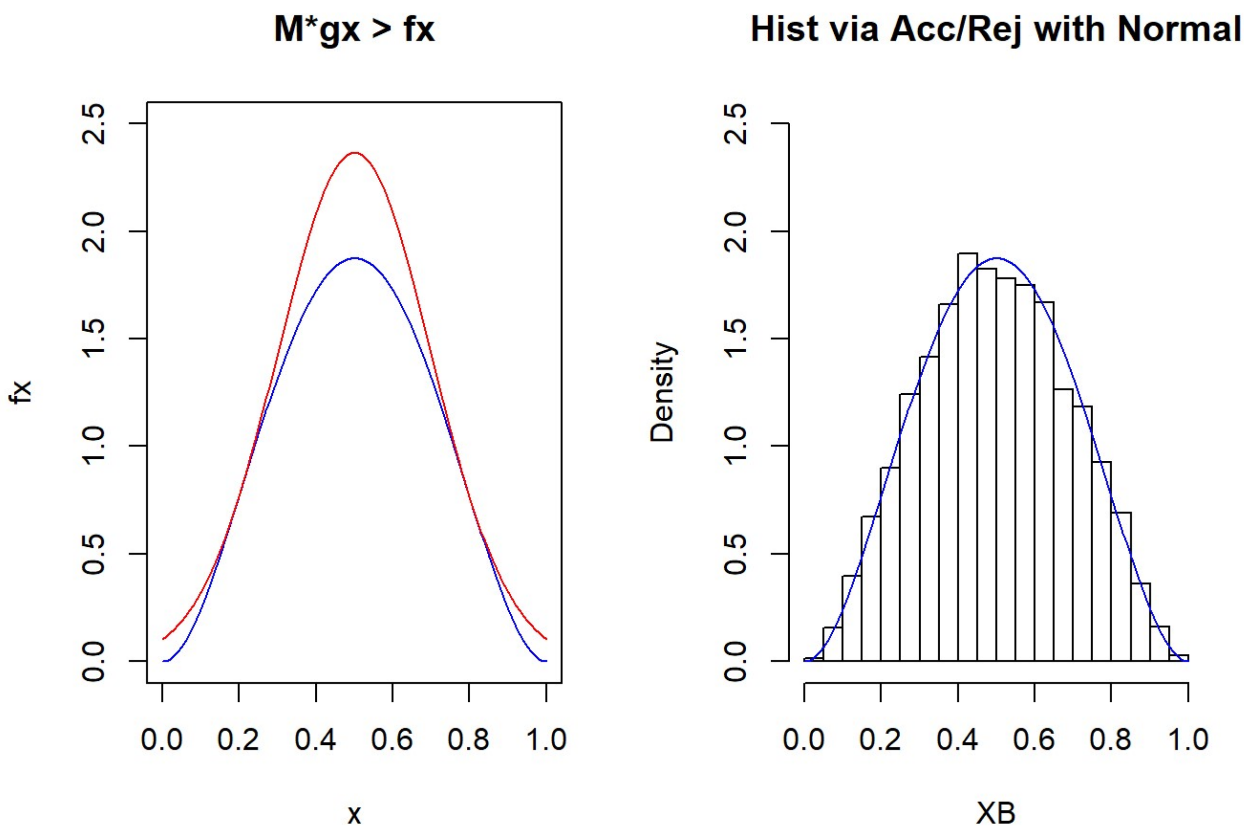
```
  # Accept Rate
    length(XB)/Nsim
```

```
## [1] 0.8495
```

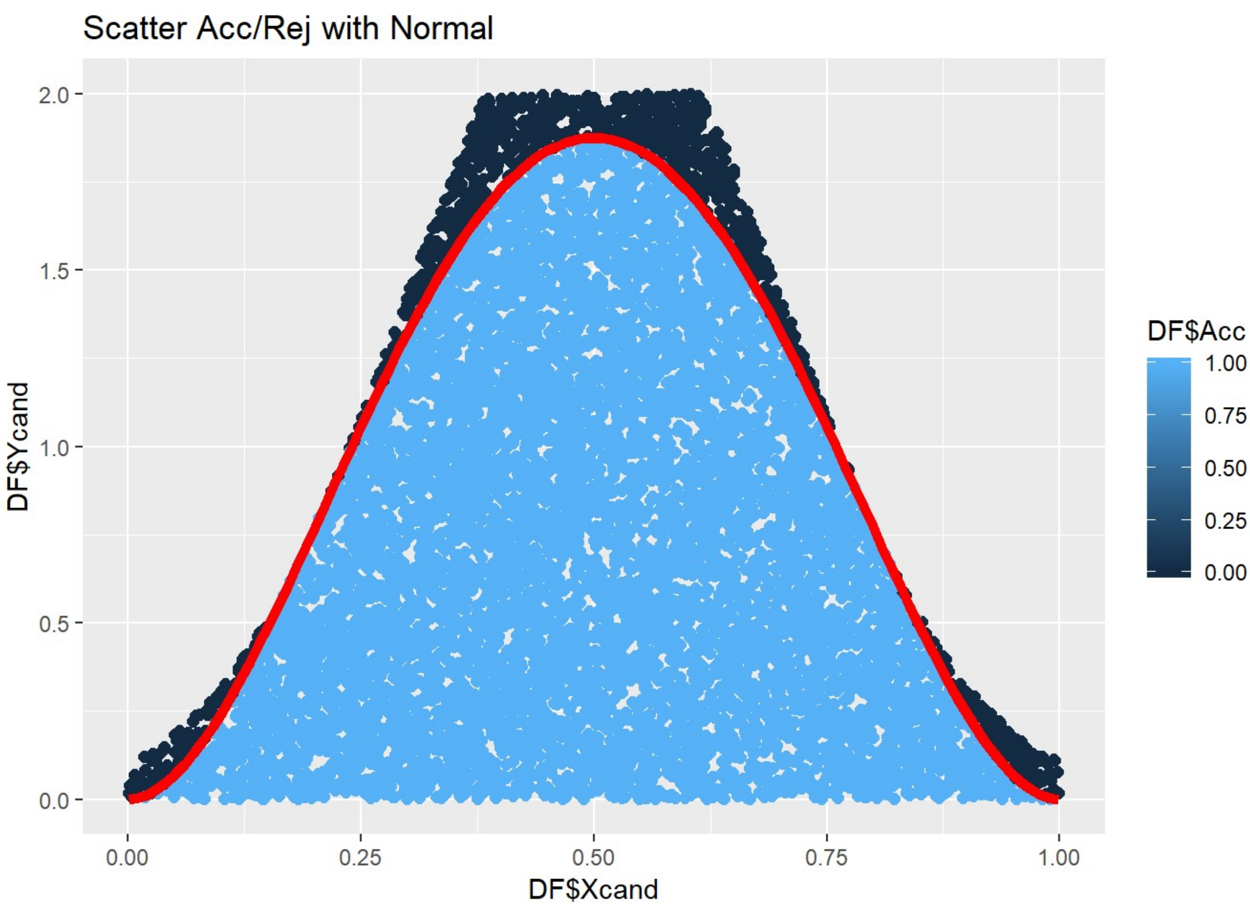
```
  # See how Close it is



  ## Get a scatter Plot
    DF <- as.data.frame(Xcand)
    DF$Ycand <- runif(Nsim,min=0,max=gx2(Xcand))
    DF$Acc <-ifelse(DF$Ycand < fx(Xcand),1,0)
    sum(DF$Acc)/Nsim
```

```
## [1] 0.8524
```

```
## Plots
    par(mfrow = c(1,2))
    # See that indeed gx*M > fx for all X
      plot(fx,xlim=c(0,1),ylim=c(0,2.5),col="blue", main = "M*gx > fx")
      plot(gx2,col="red",add = TRUE)
    # Histogram vs line
      hist(XB,freq=F,ylim=c(0,2.5),breaks = 30,main="Hist via Acc/Rej with Normal")
      plot(fx,xlim=c(0,1),ylim=c(0,2),col = "blue",add=TRUE)
```




```
    # Scatter plot
      ggplot(DF,aes(x=DF$Xcand,y=DF$Ycand,color=DF$Acc)) + geom_point(size=2) + ylim
(0,2) +
      ggtitle("Scatter Acc/Rej with Normal") +
      stat_function(fun=function(x)(30*(x^2-2*x^3+x^4)),geom="line",col="red",size=2
)
```


Scatter Acc/Rej with Normal
2.0
1.5
1.0
0.5
0.0
DF$Ycand
0.00
0.25
0.50
0.75
1.00
DF$Xcand
DF$Acc
1.00
0.75
0.50
0.25
0.00