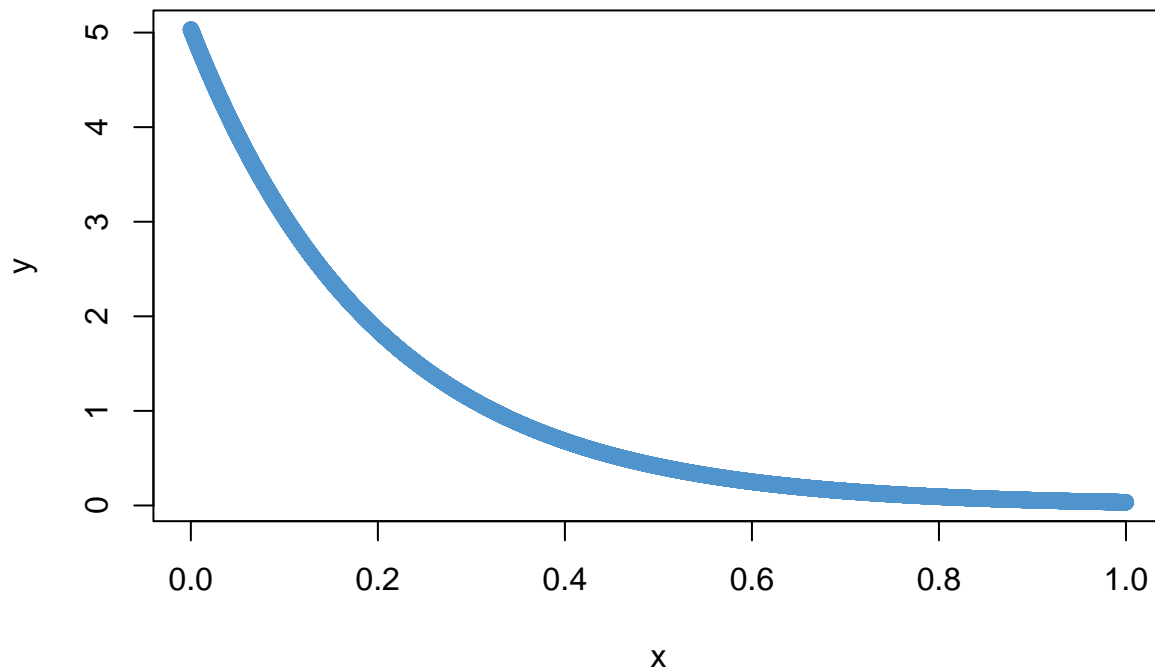


## Lab3\_Advanced\_Stats

(1A) Plot the prior graph for a situation for a coin where the prior belief for  $p(\text{head})$  is represented by the following R code : `dexp(x, rate = 5) / 0.9932621` for values of  $0 \leq x \leq 1$  and 0 otherwise. (We choose the denominator to make the Integral between 0 and 1 sum to 1).

```
x <- seq(0,1, by=0.0001)
y <- dexp(x, rate = 5)/0.9932621
plot(x,y, col="steelblue3")
```



(1B) Calculate the posterior graph with both the Metropolis algorithm and grid approximation for a case with 14 heads and 10 tails (where  $x = \text{prob}(\text{head})$ ). Show the two methods roughly agree. Compare these to a plot with a posterior for new data of 14 heads and 10 tails with a prior with  $\text{beta}(40,40)$ . (So for the observation of 14 heads and 10 tails you will end up with a graph with three plots superimposed: (i) the Metropolis algorithm with an exp prior, (ii) grid approximation with an exp prior and (iii) exact analytical solution from a  $\text{beta}(40,40)$  prior make the plots different colors so you can visualize them...)

Red Circles = Metropolis Approach

Magenta = Grid Approach

Blue = Binomial Function

```
#Metropolis Approach
piOld <- 0.5
iter <- 500000
posteriordist <- vector(length = iter)
for(i in 1:iter)
{
  pOld <- (dexp(piOld, rate = 5)/0.9932621)*dbinom(14, 24, piOld)
```

```

piNew <- piOld + rnorm(1, 0, sd=0.01)

if(piNew > 1)
  piNew = 1;
if(piNew < 0)
  piNew = 0

pNew <- (dexp(piNew, rate = 5)/0.9932621)*dbinom(14, 24, piNew)
ratio <- pNew/ pOld
if(ratio > 1||ratio >= runif(1))
  piOld = piNew;
posteriordist[i] = piOld;

#if(i %% 10000 == 0)
#{
  #histg <- hist(posteriordist, breaks=200, plot=FALSE)
  #plot(histg$mids, histg$counts/i, col="indianred3")
#}

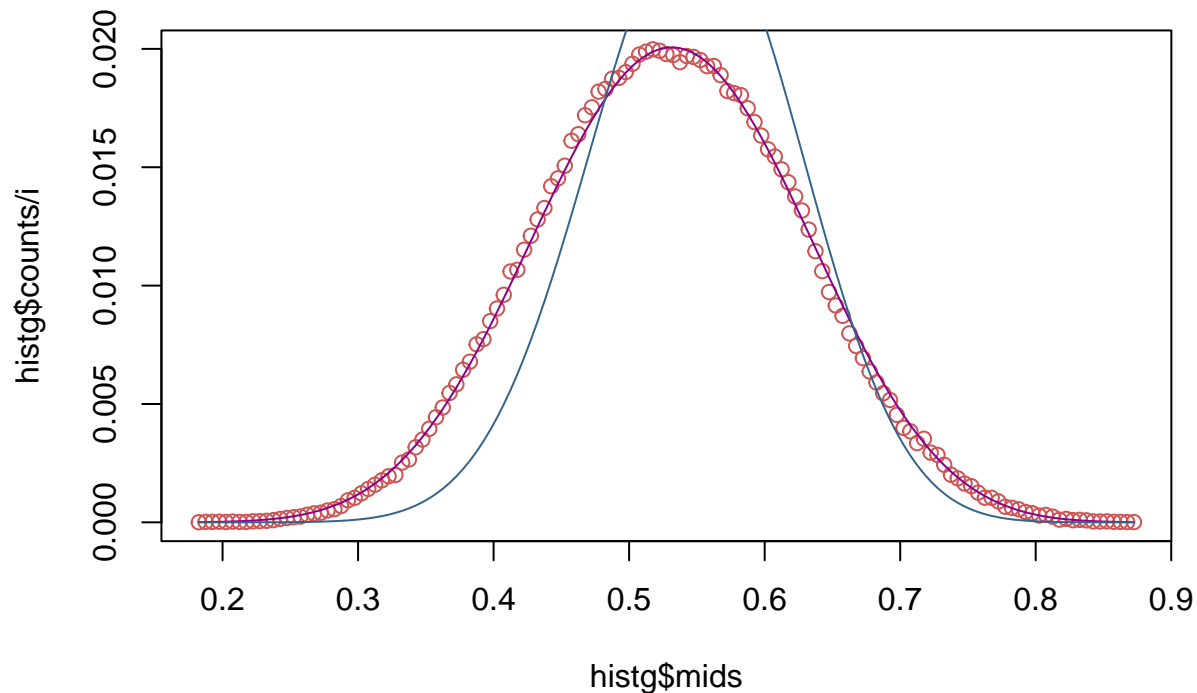
}
histg <- hist(posteriordist, breaks=200, plot=FALSE)
plot(histg$mids, histg$counts/i, col="indianred3")

#Grid Approach
posteriordist <- vector(length=length(histg$mids))
vals <- histg$mids
i <- 1;
sum <- 0;

for(x in vals)
{
  posteriodist[i] <- (dexp(x, rate = 5)/0.9932621)*dbinom(14, 24, x)
  sum = sum + posteriodist[i];
  i <- i + 1;
}

lines(histg$mids, posteriodist/sum , col= "darkmagenta")
lines(histg$mids, dbeta(histg$mids, 23, 19)/sum(dbeta(histg$mids, 23, 19)), col= "steelblue4")

```



(1C) Repeat the above calculation but for a case of 583 heads and 417 tails. (You may need to adjust your model step parameters to try and get the grid and Metropolis graphs to match up). How do the three posterior curves relate to each other now? Why does this plot look different than the plot in (1B)?

Red Circles = Metropolis Approach

Magenta = Grid Approach

Blue = Binomial Function

```
#Metropolis Approach
piOld <- 0.5
iter <- 500000
posteriordist <- vector(length = iter)
for(i in 1:iter)
{
  pOld <- (dexp(piOld, rate = 5)/0.9932621)*dbinom(583, 1000, piOld)

  piNew <- piOld + rnorm(1, 0, sd=0.01)

  if(piNew > 1)
    piNew = 1;
  if(piNew < 0)
    piNew = 0

  pNew <- (dexp(piNew, rate = 5)/0.9932621)*dbinom(583, 1000, piNew)
  ratio <- pNew/ pOld
  if(ratio > 1 || ratio >= runif(1))
    piOld = piNew;
  posteriordist[i] = piOld;

  #if(i %% 10000 == 0)
  #{
    #histg <- hist(posteriordist, breaks=200, plot=FALSE)
  }
}
```

```

    #plot(histg$mids, histg$counts/i, col="indianred3")
    #}

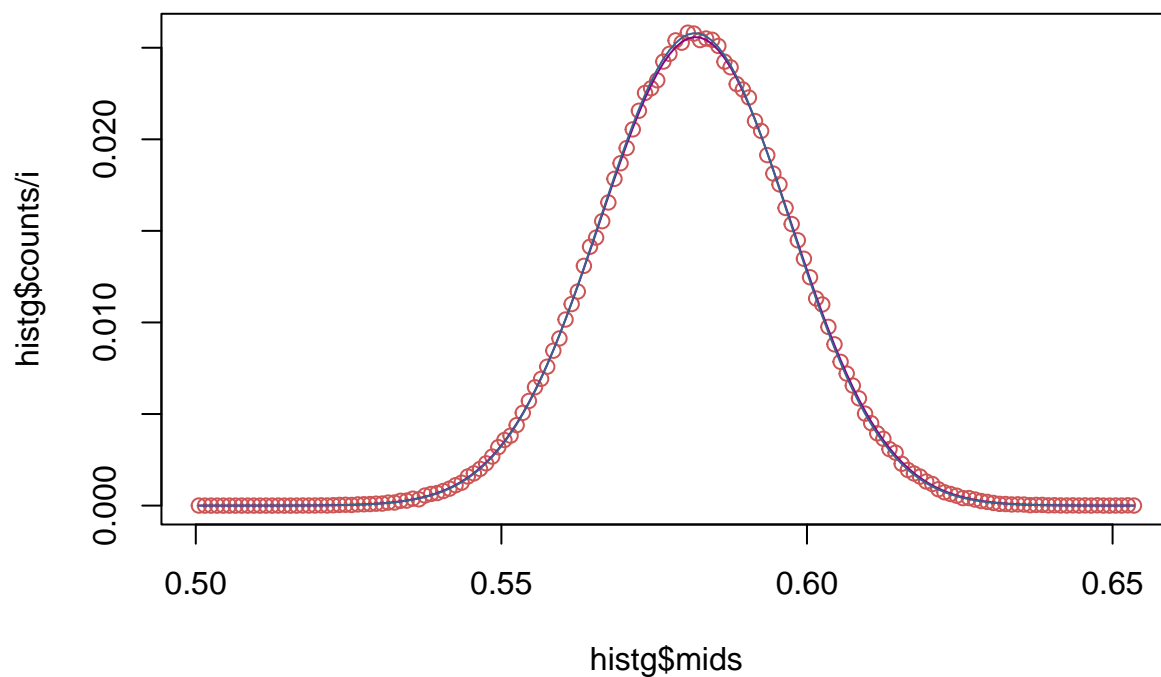
}
histg <- hist(posteriordist, breaks=200, plot=FALSE)
plot(histg$mids, histg$counts/i, col="indianred3")

#Grid Approach
posteriordist <- vector(length=length(histg$mids))
vals <- histg$mids
i <- 1;
sum <- 0;

for(x in vals)
{
  posteriordist[i] <- (dexp(x, rate = 5)/0.9932621)*dbinom(583, 1000, x)
  sum = sum + posteriordist[i];
  i <- i + 1;
}

lines(histg$mids, posteriordist/sum , col= "darkmagenta")
lines(histg$mids, dbeta(histg$mids, 592, 426)/sum(dbeta(histg$mids, 592, 426)), col= "steelblue4")

```



It looks different, because we are adding more and more trial data to the algorithm, which allows it to approach the true binomial function with less iterations.