Recall $P(B|A) = \frac{P(B\cap A)}{P(A)} = \frac{P(A|B)\cdot P(B)}{P(AB)+P(A\bar{B})} = \frac{P(A|B)\cdot P(B)}{P(A|B)P(B)+P(A|\bar{B})P(\bar{B})}$

for cts: $f(\theta|x) = \frac{f(x|\theta)\, f(\theta)}{\underbrace{\int f(x|\theta)\, f(\theta)\, d\theta}_{f(x)}} = \frac{\mathcal{L}(\theta;x)\, f(\theta)}{\int f(x|\theta)\, f(\theta)\, d\theta}$ where $\mathcal{L}(\theta;x) = f(x|\theta)$

posterior

likelihood

prior

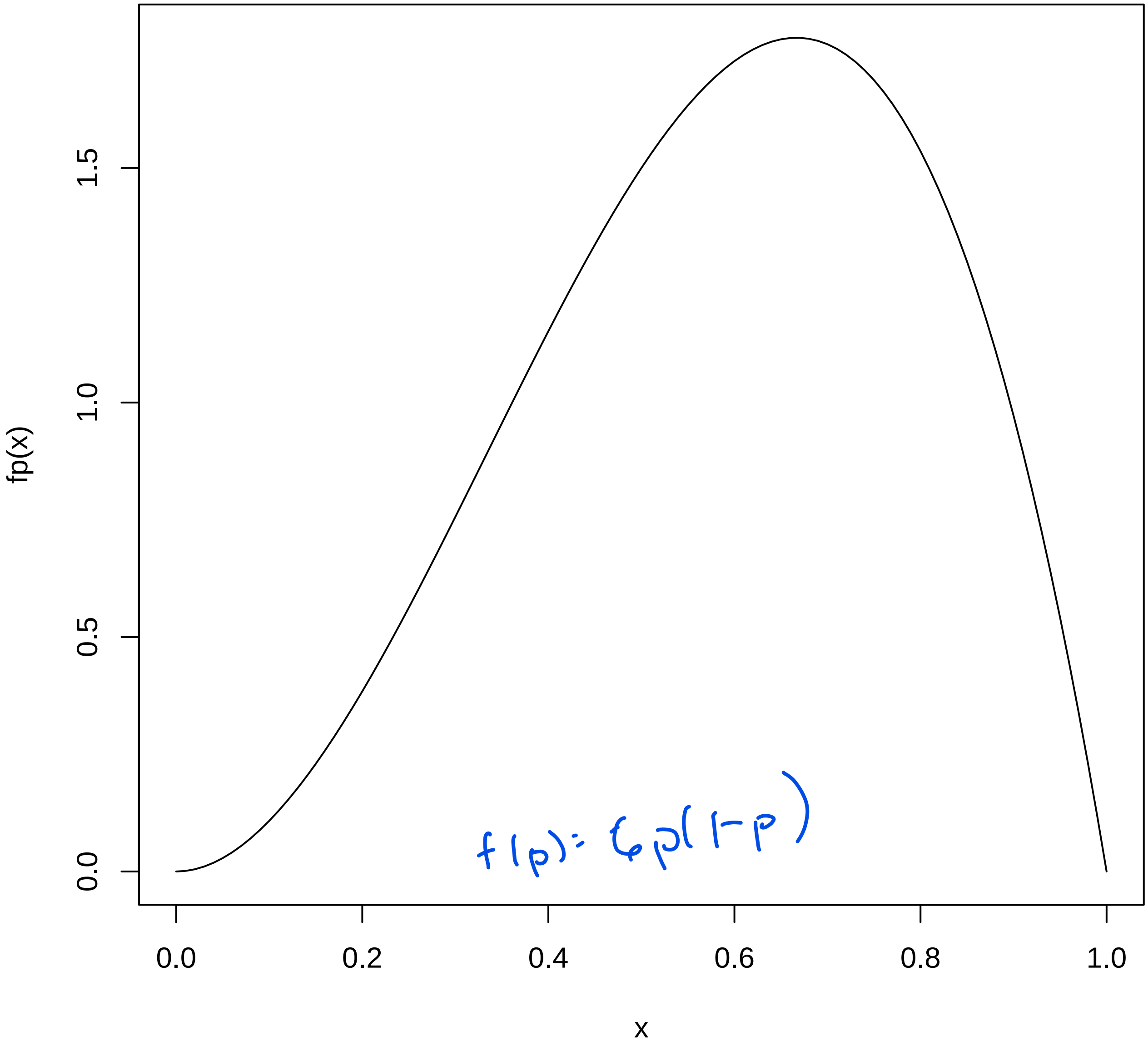
Bayesian

posterior weighted avg likelihood

9.1 Bayesian Analyses

We've previously discussed Bayesian approaches. The major philosophical difference is that unlike the analyses we've conducted until now about an unknown parameter of interest, the Bayesian approach is *not* to treat it like a fixed, unknown value. Instead, it's treated as a stochastic/random value. This is different than how an estimator is a random value. A simple analogy to think of is a very bad machine that produces very poorly manufactured coin-like objects. Suppose that the distribution of unfair coins follows a Beta(3,2) so the distribution of parameters for coins is given by $f(p) = 6p(1-p)$ (shown below) where p represents the true proportion of times the corresponding coins flips to heads.

```
fp = function(p)12*p^2*(1-p)
curve(fp,0,1)
```



Clearly, the machine mostly produces coins that give heads approximately 60% of the time. We can perform simple integration approaches to ask the probability a randomly selected coin generated by the machine is within 10% of being fair: $P(0.4 < X < 0.6)$, the mean, the variance, etc., all things you did in STAT2000 and Math3020.

Now, consider many people each receive one of these coins. Some people receive a coin corresponding to a parameter close to $p = 0.35$, lots of people have a coin with a parameter close to $p = 0.55$ and very few people have a coin with a parameter close to where $p = .001$. Everyone receives a different coin with (almost surely) different p following the distribution above. Now, suppose we ask one randomly selected person to flip it 10 times, and record how many heads as that X_1. Then, we randomly select another person and record how many heads after 10 flips as X_2. Not only do we have the randomness of a binomial of 10 coin flips, we have the randomness of what the value of p was from each of the coins used (and we never learned). The number of heads for a particular X is always conditional on the value of p for that coin.

To obtain the distribution of this new distribution, we average over (integrate over) the random p:

$$f(x) = \int \underbrace{f(x,p)} \, dp$$

$$6p(1-p)$$

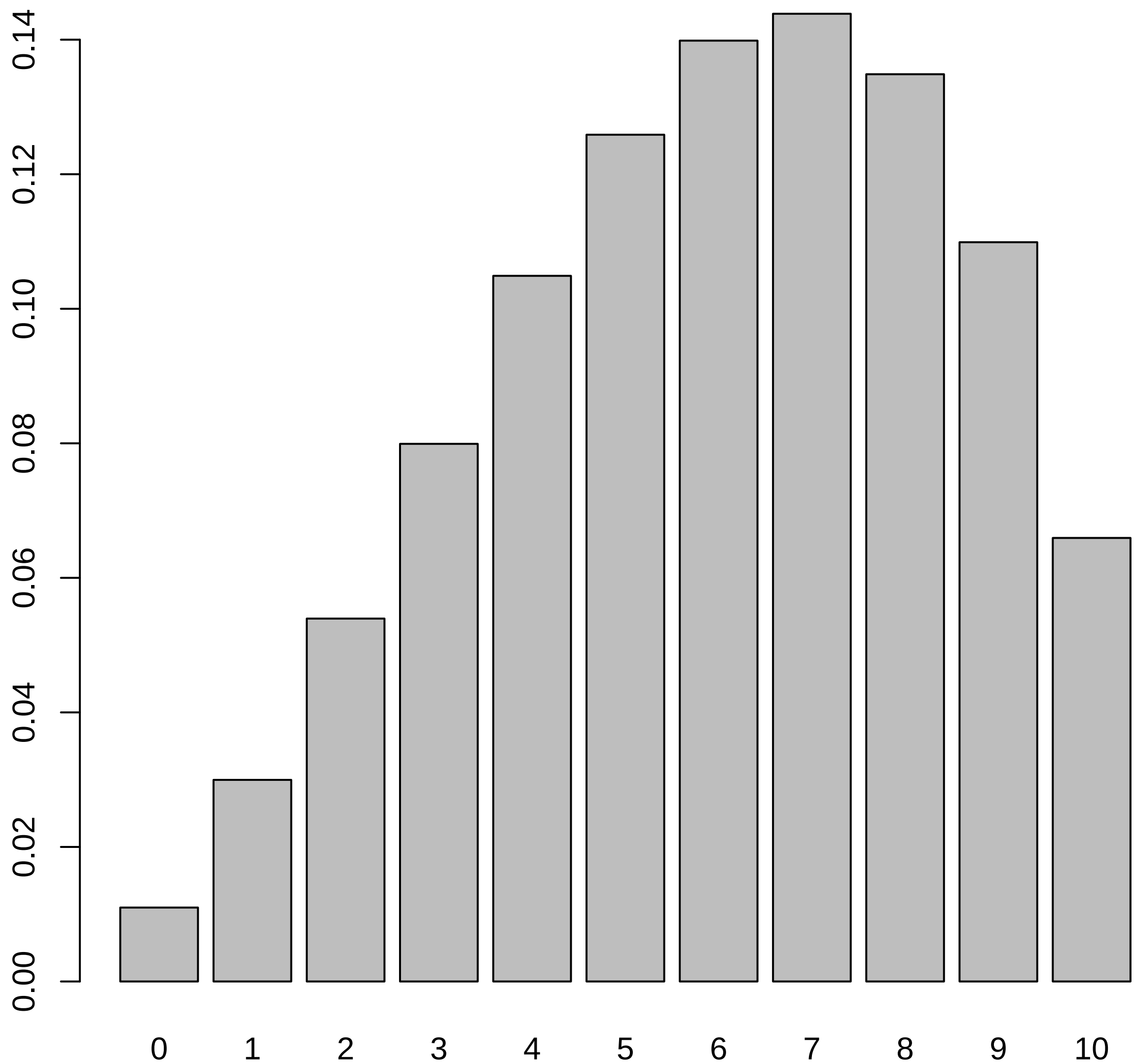
$$f(x) = \int_p f(x|p)dF(p) = \int_p f(x|p)f(p)dp$$

$$\binom{n}{x} p^x (1-p)^{n-x}$$

Earlier in the course we did this example and found a distribution for f in an example like this. However, the integral will not always exist in a closed form. On tests/exams we focus when this does have a closed form so you can actually finish questions, but in real life it's almost irrelevant because we can numerically integrate with computers. Computers can just *simulate* the sampling process to indirectly estimate this marginal density. Note: when we simulate we only take a finite approximation, which means we are estimating distributions with discrete distributions.

In our example, the data's distribution is $f(x) = \frac{1}{2002}(x+1)(x+2)$ for $x = 0, 1, 2, \ldots, 10$, so we should obtain the following mass function

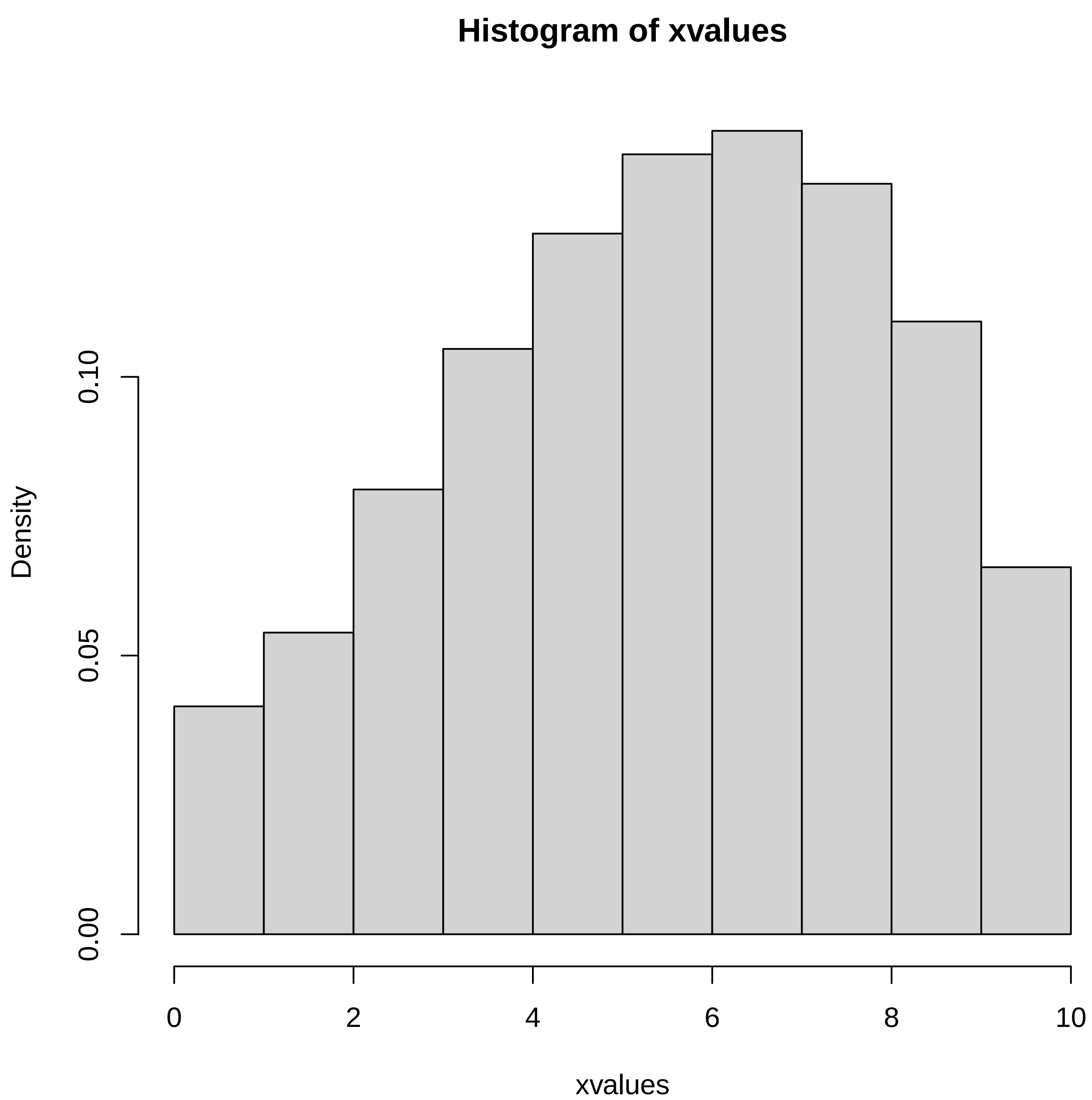
```
fx = function(x)(x+1)*(x+2)*(11-x)/2002
barplot(fx(0:10),names=0:10)
```



To simulate this we simulate lots of values for p, then simulate lots of values for x accordingly. In general, we simulate one x for every simulated p.

```
simulateOneX = function(){
            #simulate parameter that is integrated away
  p=rbeta(1,3,2 )
            #create the x according to that parameter
```

```
  x=rbinom(n=1,size=10,prob=p)
}

xvalues = replicate(1e6,simulateOneX())
hist(xvalues,probability=T,breaks=11)
```



```
#create 1 bar for every possible value and scale to be probabilities
```

We can also easily find mean, variance, etc., which should both be 6.:

```
mean(xvalues)

## [1] 5.999652

median(xvalues)

## [1] 6

var(xvalues)

## [1] 5.998074
```

So far, we haven't actually constructed a posterior distribution. we will now, and this is where Bayesian statistics really shines. It lets us perform inference on p as though it's random, which is *intuitive* and easy (if you can calculate probabilities with a computer).

Example 1

Suppose we wait until the machine creates one more coin, and flip it 10 times and see 2 heads. What do you guess the value of p is for that coin? We know values near 0.6 are likely, but we know a small number of heads are likely when p is smaller (around 0.2). Can you use the observed ~~information~~ data while keeping in mind you were more likely to get a coin where $p = .6$ was more likely (but not guaranteed?).

Rather than focus on a distribution for p alone, $f(p)$, we want a distribution in light of some data, or given some data, $f(p|x)$.

From Bayes's Theorem we know

fixed

$$f(p|x) = f(x|p)\frac{f(p)}{f(x)} = \binom{10}{x} p^x (1-p)^{10-x} \times \frac{12p^2(1-p)^3}{\frac{1}{2002}(x+1)(x+2)(11-x)}$$

fixed

.

Note that we usually only care about the part of the density that includes our variables. The denominator doesn't involve p and only acts as a normalizing constant when you only have one collection of data. It's not wrong to include it, but since we're approximating with a discrete distribution, we have to rescale to be a proper pmf, anyway, so it's kind of useless in approximations. So then, we really only care about

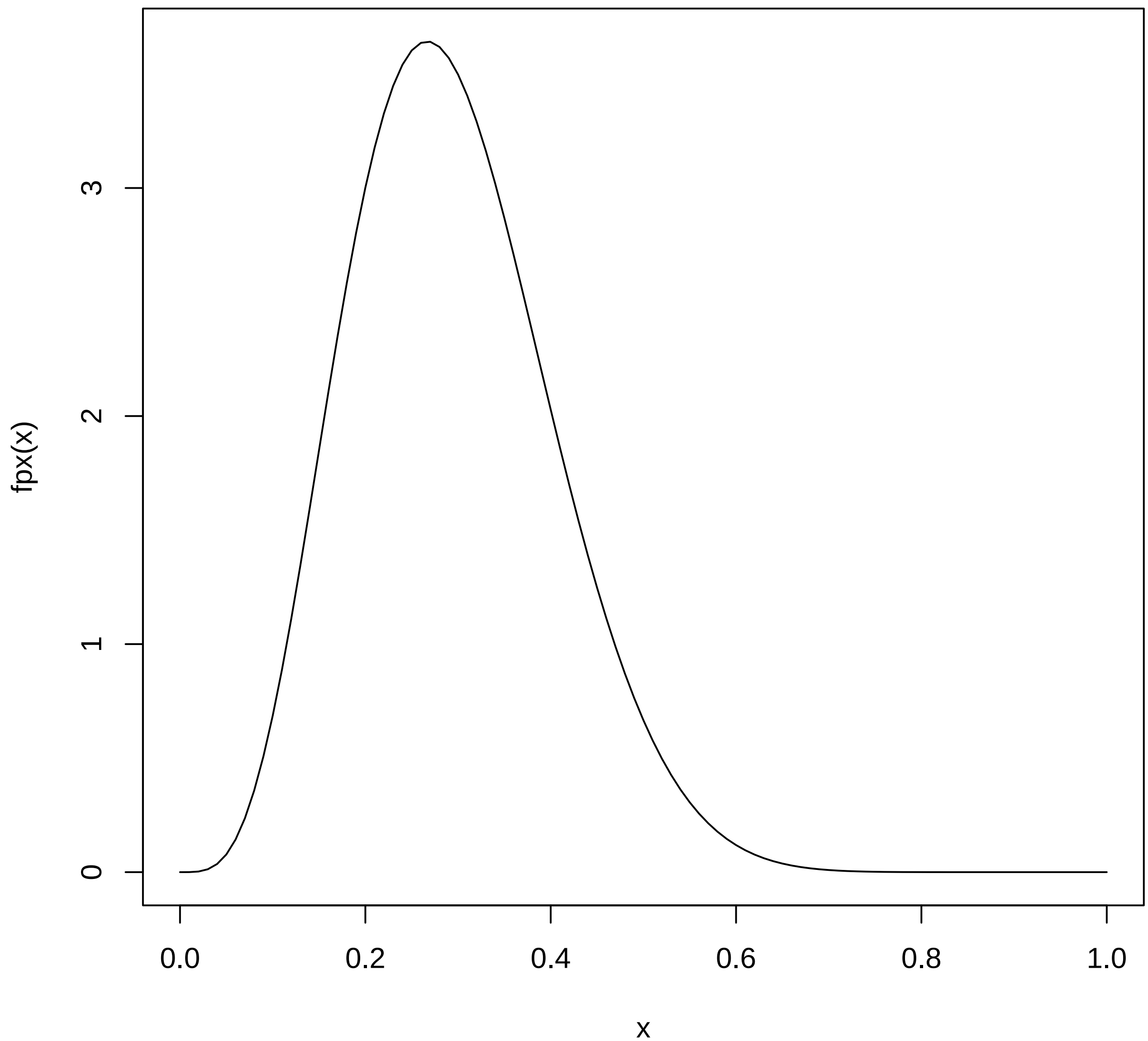
$$f(p|x) \propto f(x|p)f(p) \propto p^{x+2}(1-p)^{13-x}$$

In our case, we have $x = 2$ so

$$f(p|x=2) = f(2|p)\frac{f(p)}{f(2)} = 21840p^4(1-p)^{11} \propto p^4(1-p)^{11}$$

which is a simple polynomial:

```
fpx=function(p)21840*p^4*(1-p)^11
curve(fpx,0,1)
```



In our closed form, we can find a middle 95% interval, say $F^{-1}(.025)$ to $F^{-1}(0.975)$. Some guessing with an online integrator gives us $F(0.11) = 0.0248489$ and $F(0.52) = 0.9731766$. On tests and exams, I'll give you something you can actually integrate, or I'll even give you a table of values to approximate integrals.

We then obtain the interval of $p \in (0.11, 0.52)$. This is almost like a confidence interval, except, because p is random, it can't be confidence interval. Instead, it's a Bayesian credible interval.

This is not a confidence interval, however, it's a credible interval. The actual probability a value of p is between this and will then yields 2 heads is 0.11 and 0.52. Again, the interpretation is *extremely* simple and understandable, unlike that confidence interval.

$f(x=2, p=0) \cdot f(p=0) = a_0$
$f(x=2, p=0.1)\ f(p=0.1) = a_1$
$\vdots$
$f(x=2, p=1.0)\ f(p=1.0) = a_{10}$

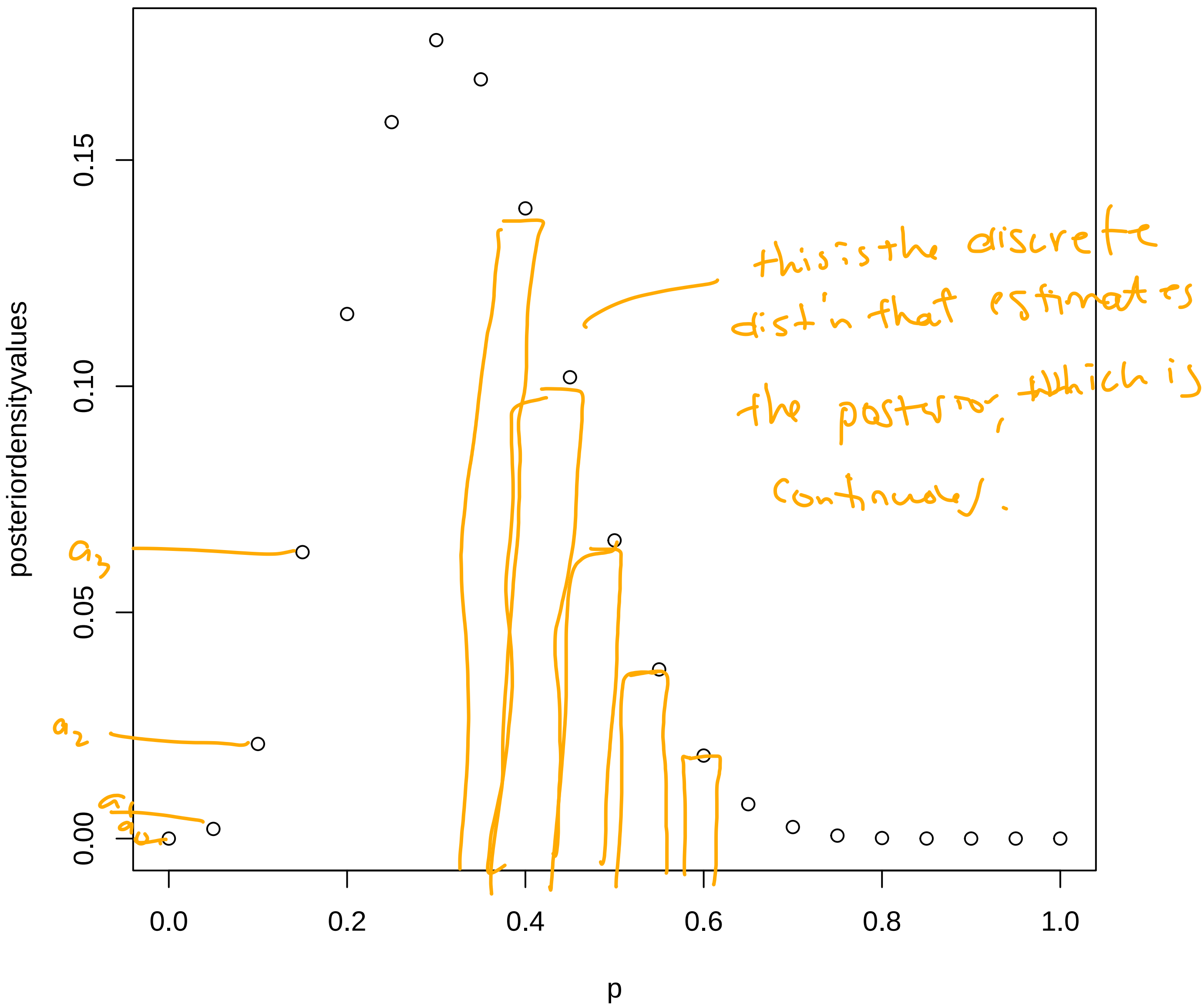
$p=0$	$p=.1$	$p=.2$	$\cdots$	$p=1.0$
$a=a_0$	$a=a_1$	$\cdots$		$a=a_{10}$

$$f(p|x) = \frac{f(x|p) \cdot f(p)}{f(x) \doteq a_0 + a_1 + \cdots a_{10}}$$

} estimate of $\int_p f(x|p) f(p)\, dp$

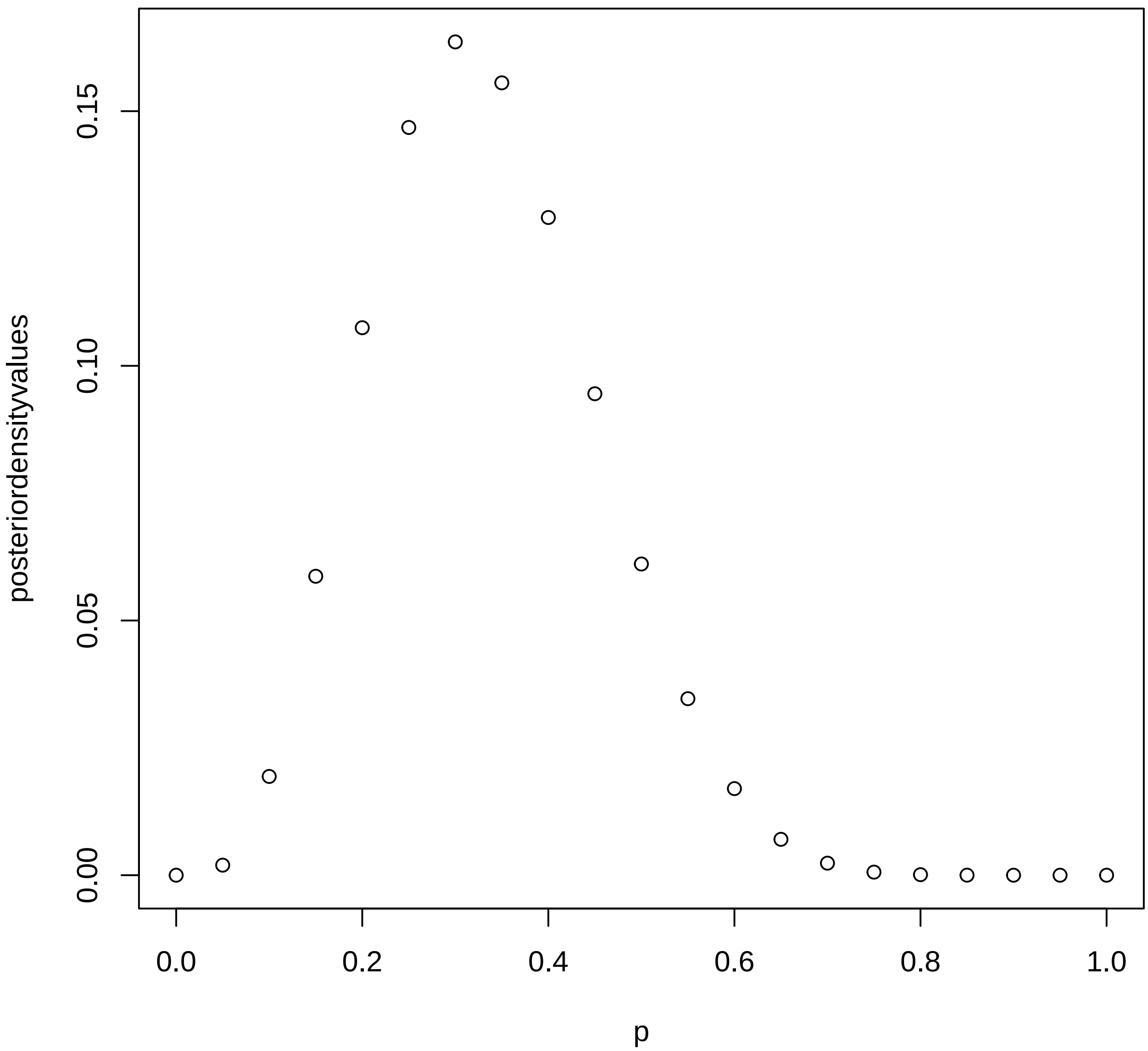
```r
    #generate some p
p = seq(0,1,.05)

    # define likelihood: f(x=2|p) and prior:f(p)
likelihood=function(p,x=2) dbinom(x=x,size=10,prob=p)
prior =function(p) dbeta(p,3,2)

    #calculate all the prior values (for all the p)
priorvalues = sapply(p,prior)
    #calculate all the likelihood values (for all the p)
likelihoodvalues= sapply(p,likelihood)


    #calculate the posterior density

posteriordensityvalues = likelihoodvalues * priorvalues
plot(p,posteriordensityvalues)
```

$f(x=2|p)$ – (likelihood); $f(p)$ – (prior); a_i (posteriordensityvalues)

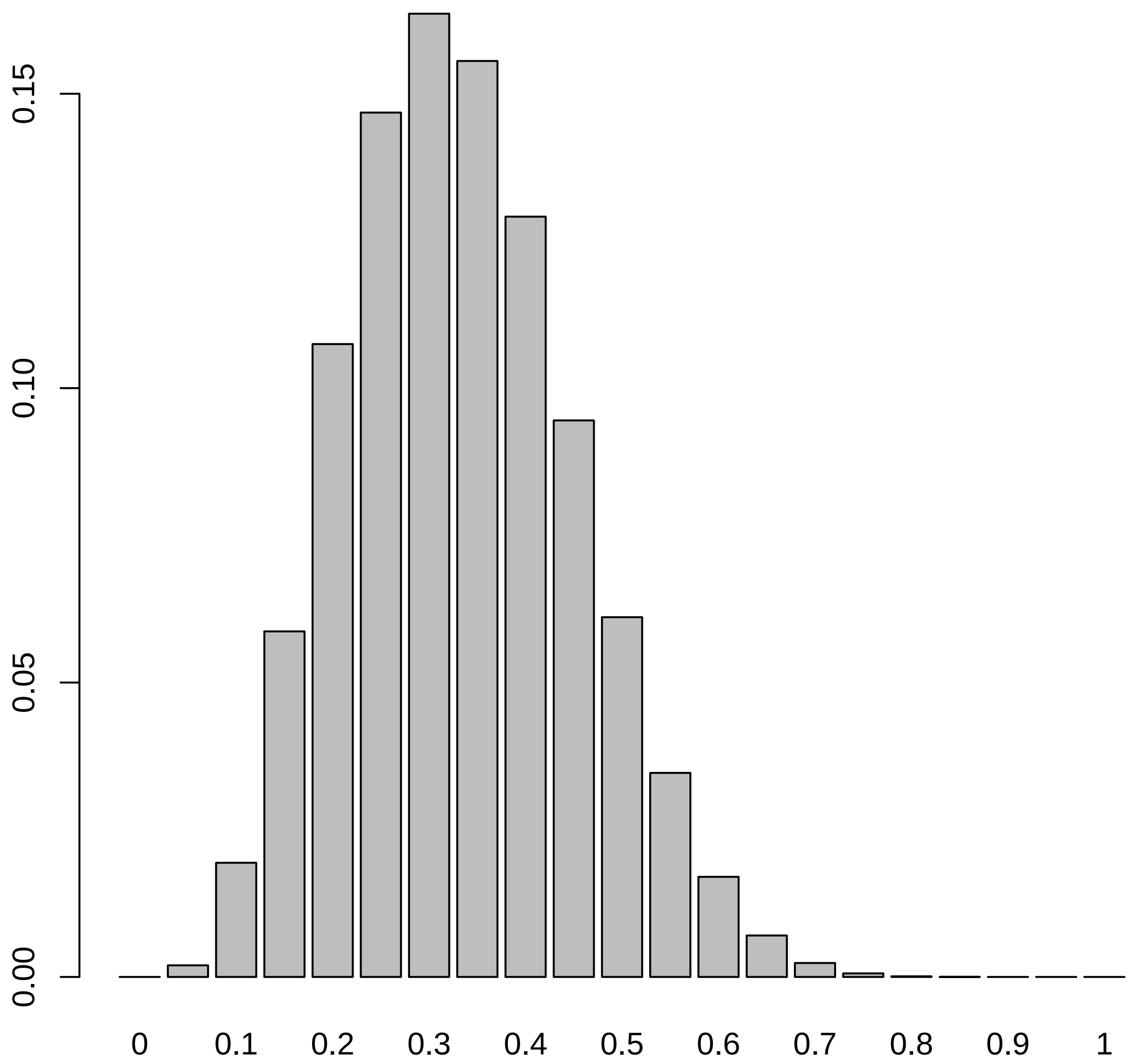


We actually treat this like a discrete mass function for these values of p and scale accordingly. We do that by dividing by the corresponding sum (we have to do this because we didn't properly integrate)

```
posteriordensityvalues=posteriordensityvalues/sum(posteriordensityvalues)
plot(p,posteriordensityvalues)
```

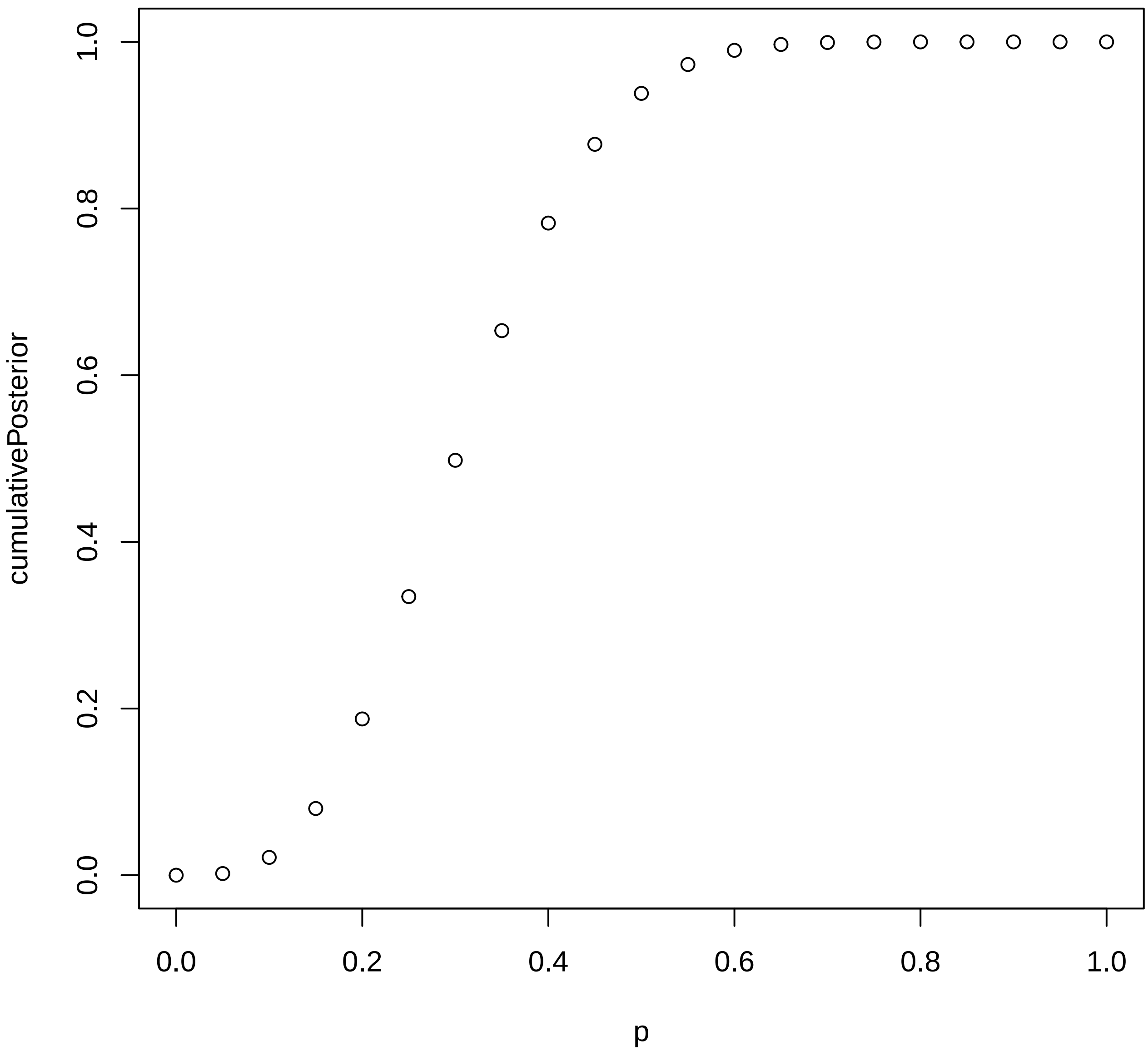



```
barplot(posteriordensityvalues,names=p)
```



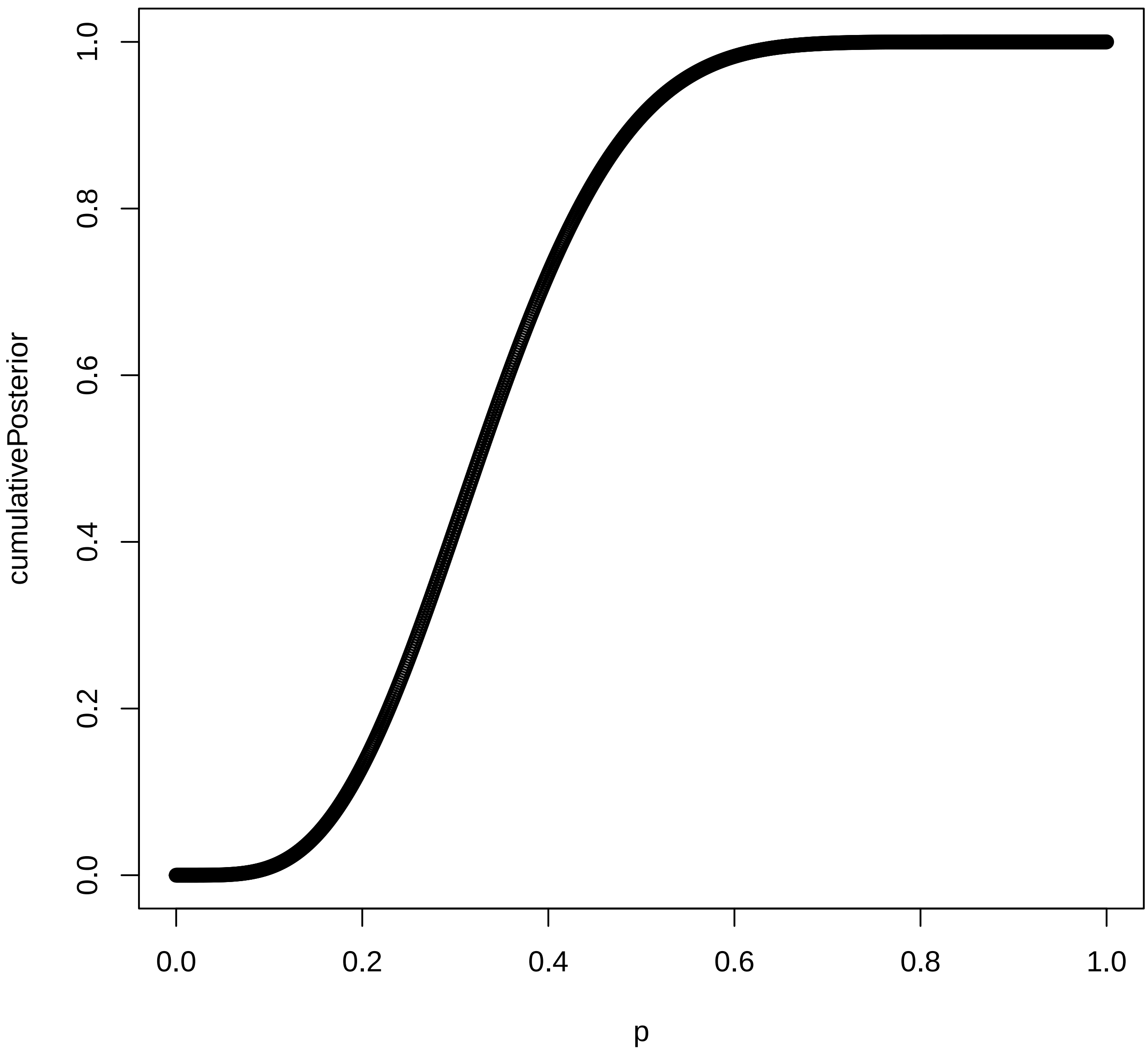
We have now created a probability mass function with 21 values of p to approximate $f(x|p)$. If we pick finer values of p (many more), repeat this but you get the idea. We also easily obtain $F(X|p)$

```
cumulativePosterior=cumsum(posteriordensityvalues)
plot(p,cumulativePosterior)
```



You can easily see here that we won't get good "resolution" of quantiles like $F^{-1}(.025)$ and $F^{-1}(.975)$ so we repeat the above with `p = seq(0,1,.001)` in the first actual line of code (or some small steps. When then obtain

```
plot(p,cumulativePosterior)
```



Now finding quantiles will be easy

```
p[which.max(cumulativePosterior>0.025)]
## [1] 0.128

#or
p[which.min(cumulativePosterior<0.025)]
```

```
## [1] 0.128

p[which.max(cumulativePosterior>0.975)]

## [1] 0.581

p[which.min(cumulativePosterior<0.975)]

## [1] 0.581
```

Which are very close to the previous estimates

Definition 1: Credible Interval

For $0 < \alpha < 1$ let $1-\alpha$ be the credible coefficient. Consider the posterior $f(\theta|\boldsymbol{X}$ for sample $\boldsymbol{X}$. Then, $L = L(\theta|\boldsymbol{X})$ and $U = U(\theta|\boldsymbol{X})$ be two (random) statistics of the sample. Then the interval (L, U) is the $(1-\alpha)\times 100\%$ credible interval for the parameter θ if $P(\theta \in (L, U)) = 1-\alpha$.

As before, we can also easily find the mean and variance from the definition for discrete variables $E(P|x=2) = \sum pf(p|x=2)$

```
#f(p|x) is the posterior density

#mean is E(p|X) =sum of p*f(p|x)
sum(p*posteriordensityvalues)

## [1] 0.3333333

#variance is E(p^2|x)-(E(p|x))^2
sum(p^2*posteriordensityvalues) - (sum(p*posteriordensityvalues))^2

## [1] 0.01388889
```