HW 3

Zach White

9/13/2016

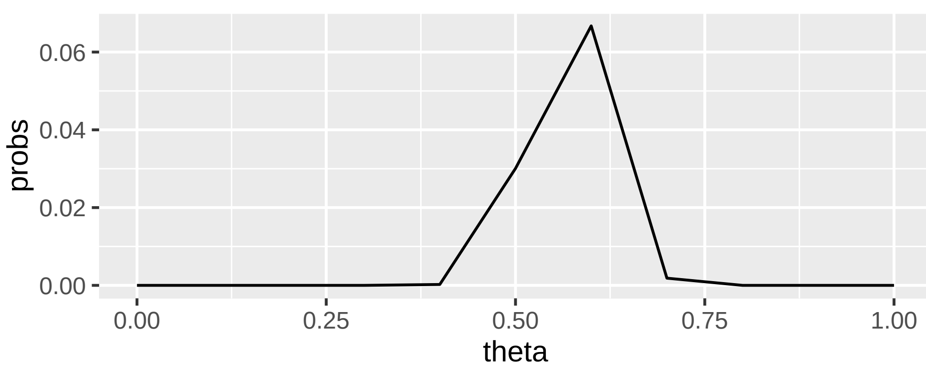
Problem 1

1.b

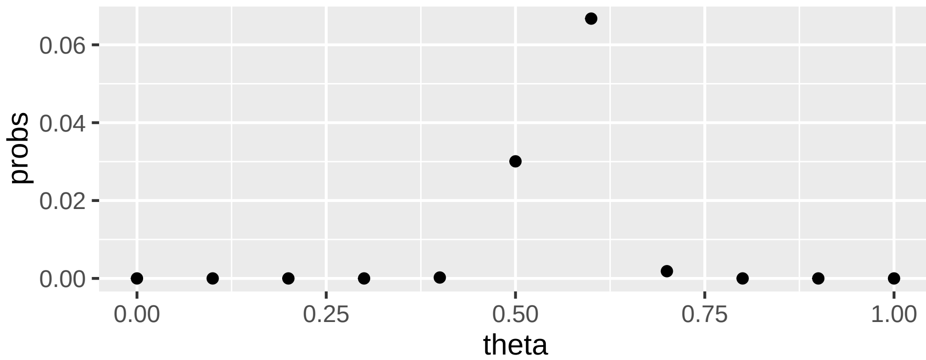
```
theta = seq(0,1,by= .1)
probs = dbinom(57, 100, theta)
data.bern = data.frame(cbind(theta,probs))
data.bern
```

```
##    theta        probs
## 1    0.0 0.000000e+00
## 2    0.1 4.107157e-31
## 3    0.2 3.738459e-16
## 4    0.3 1.306895e-08
## 5    0.4 2.285792e-04
## 6    0.5 3.006864e-02
## 7    0.6 6.672895e-02
## 8    0.7 1.853172e-03
## 9    0.8 1.003535e-07
## 10   0.9 9.395858e-18
## 11   1.0 0.000000e+00
```

```
plot = ggplot(data = data.bern, aes(x = theta, y = probs))
plot + geom_line()
```




```
plot + geom_point()
```

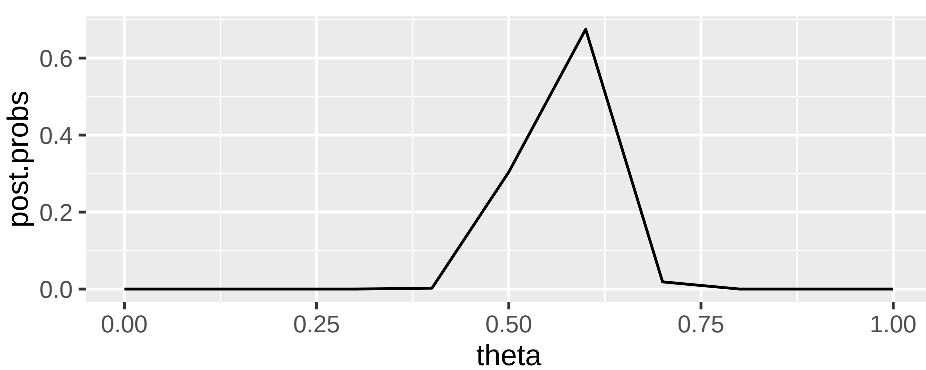


1.c

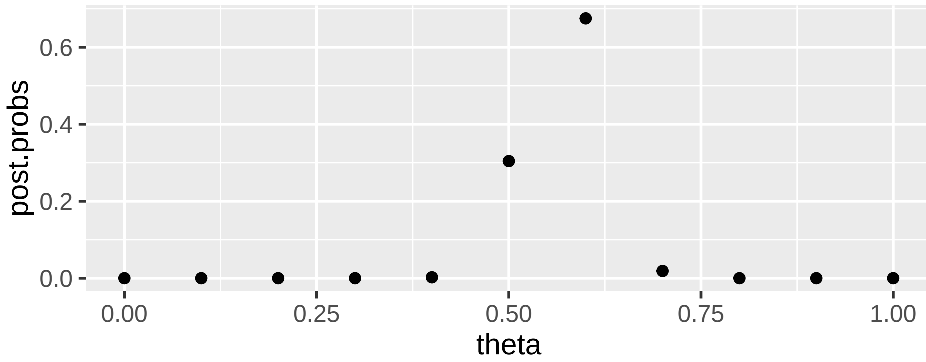
```
prior = 1/11
n = 100
sum.y = 57
probs = dbinom(57, 100, theta)
probs.prior = probs * prior
marg.sumy = sum(probs.prior)
post.probs = probs.prior /marg.sumy
data.post = data.frame(cbind(theta,post.probs))
data.post
```

```
##    theta   post.probs
## 1    0.0 0.000000e+00
## 2    0.1 4.153701e-30
## 3    0.2 3.780824e-15
## 4    0.3 1.321705e-07
## 5    0.4 2.311695e-03
## 6    0.5 3.040939e-01
## 7    0.6 6.748515e-01
## 8    0.7 1.874172e-02
## 9    0.8 1.014907e-06
## 10   0.9 9.502335e-17
## 11   1.0 0.000000e+00
```

```
plot = ggplot(data = data.post, aes(x = theta, y = post.probs))
plot + geom_line()
```



```r
plot + geom_point()
```

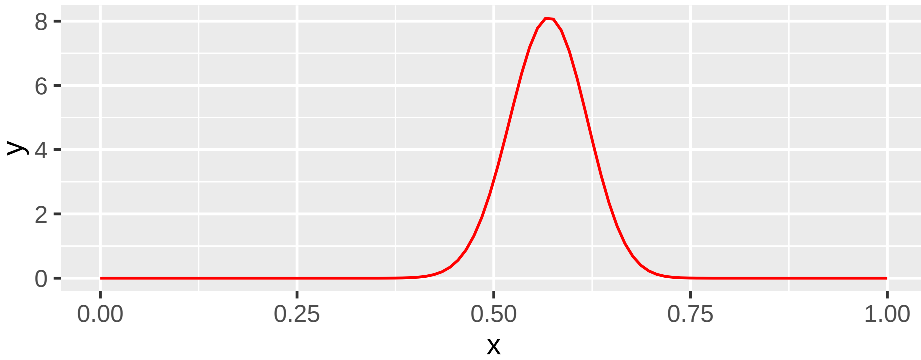


1.d

```r
alpha = 1
beta = 1
n = 100
sum.y = 57
post.alpha = sum.y + alpha
post.beta = n - sum.y + beta
post.probs = dbeta(theta,post.alpha,post.beta)
cbind(theta,post.probs)
```

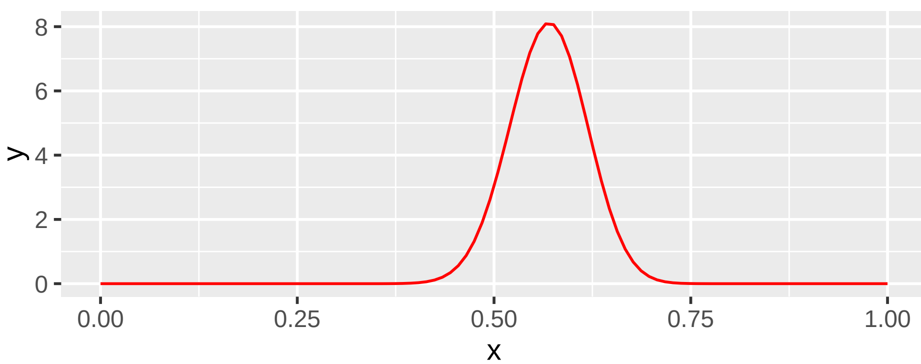
```
##       theta   post.probs
##  [1,]   0.0 0.000000e+00
##  [2,]   0.1 4.148228e-29
##  [3,]   0.2 3.775843e-14
##  [4,]   0.3 1.319964e-06
##  [5,]   0.4 2.308650e-02
##  [6,]   0.5 3.036933e+00
##  [7,]   0.6 6.739624e+00
##  [8,]   0.7 1.871703e-01
##  [9,]   0.8 1.013570e-05
## [10,]   0.9 9.489816e-16
## [11,]   1.0 0.000000e+00
```

```r
x <- seq(0, 1, len = 100)
p <- qplot(x, geom = "blank")
stat <- stat_function(aes(x = x, y = ..y..), fun = dbeta, colour="red"
                      , n = 100,args = list(shape1 = post.alpha, shape2 = post.beta))
p + stat
```



1.e

```
x <- seq(0, 1, len = 100)
p <- qplot(x, geom = "blank")
stat <- stat_function(aes(x = x, y = ..y..), fun = dbeta, colour="red"
                      , n = 100,args = list(shape1 = post.alpha, shape2 = post.beta))
p + stat
```



The first two are similiar. However, it is important to note that in the cse fothe first model. THe values aren't a complete density because we aren't sampling over all the values of θ. In the second example, since we use Baye's rule, it because a valid density, an the sum of the values sum to 1. The third and fourth plots are the same because a uniform prior is a beta distribution with $\alpha = \beta = 1$.

Problem 2

```
theta.vals = seq(.1,.9,by = .1)
n.vals = c(1,2,8,16,32)
theta.n =length(theta.vals)
n.n = length(n.vals)
a.data.frame = matrix(NA,n.n, theta.n)
b.data.frame = a.data.frame
for(i in 1:n.n){
  for(j in 1:theta.n){
    a.data.frame[i,j] = theta.vals[j] * n.vals[i]
    b.data.frame[i,j] = (1-theta.vals[j]) * n.vals[i]
  }
```

```
}

post.a = a.data.frame + sum.y
post.b = b.data.frame + n - sum.y
post.a
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9]
## [1,] 57.1 57.2 57.3 57.4 57.5 57.6 57.7 57.8 57.9
## [2,] 57.2 57.4 57.6 57.8 58.0 58.2 58.4 58.6 58.8
## [3,] 57.8 58.6 59.4 60.2 61.0 61.8 62.6 63.4 64.2
## [4,] 58.6 60.2 61.8 63.4 65.0 66.6 68.2 69.8 71.4
## [5,] 60.2 63.4 66.6 69.8 73.0 76.2 79.4 82.6 85.8
```

```
post.b
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9]
## [1,] 43.9 43.8 43.7 43.6 43.5 43.4 43.3 43.2 43.1
## [2,] 44.8 44.6 44.4 44.2 44.0 43.8 43.6 43.4 43.2
## [3,] 50.2 49.4 48.6 47.8 47.0 46.2 45.4 44.6 43.8
## [4,] 57.4 55.8 54.2 52.6 51.0 49.4 47.8 46.2 44.6
## [5,] 71.8 68.6 65.4 62.2 59.0 55.8 52.6 49.4 46.2
```

```
1 - pbeta(.5, post.a,post.b)
```

```
##           [,1]      [,2]      [,3]      [,4]      [,5]      [,6]      [,7]
## [1,] 0.9067174 0.9100204 0.9132361 0.9163656 0.9194100 0.9223703 0.9252477
## [2,] 0.8914671 0.8987201 0.9056150 0.9121591 0.9183604 0.9242272 0.9297689
## [3,] 0.7686623 0.8130640 0.8517895 0.8847656 0.9121799 0.9344278 0.9520516
## [4,] 0.5445167 0.6591158 0.7606645 0.8430584 0.9042520 0.9458344 0.9716716
## [5,] 0.1554088 0.3248808 0.5417239 0.7465826 0.8894420 0.9628561 0.9905703
##           [,8]      [,9]
## [1,] 0.9280434 0.9307587
## [2,] 0.9349949 0.9399155
## [3,] 0.9656774 0.9759580
## [4,] 0.9863403 0.9939428
## [5,] 0.9982198 0.9997538
```

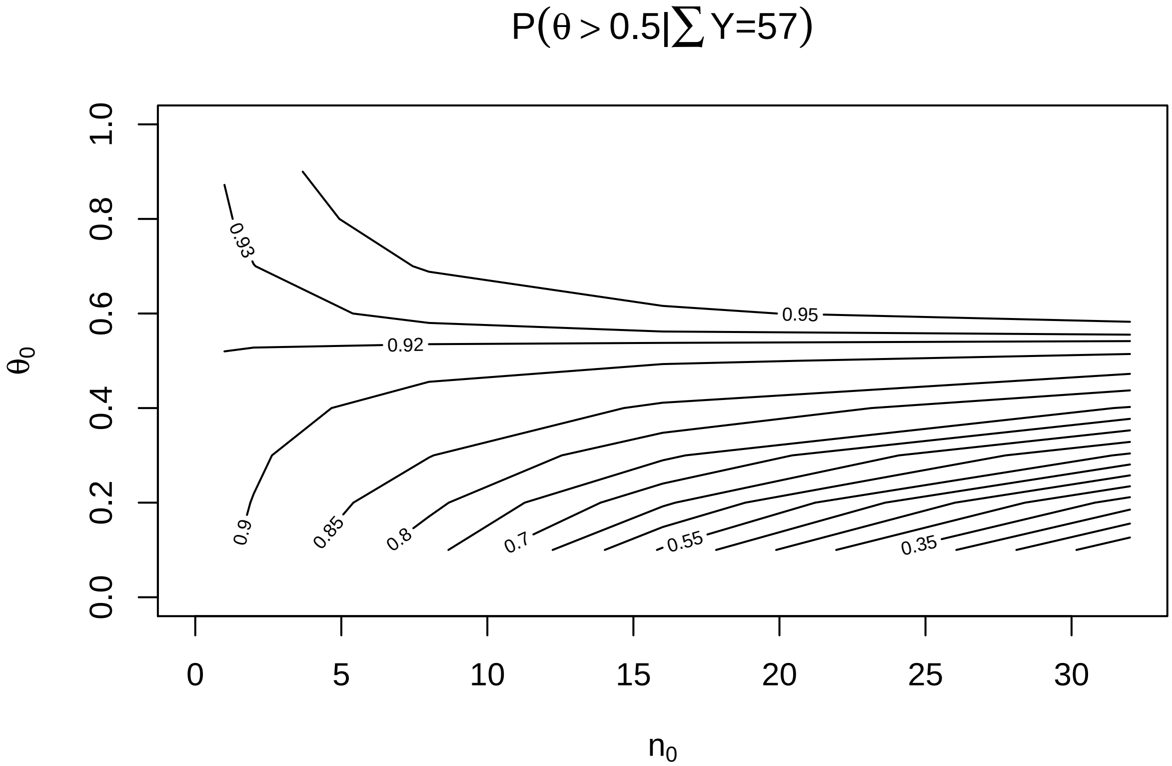
```
eta = seq(.1, .9, by=.1)
K = c(1, 2, 8, 16, 32)

post.probs.50 = function(n.input,theta.input){
  1 - pbeta(.5, theta.input*n.input + sum.y , n.input*(1-theta.input) + n - sum.y )
}

probs.post = outer(n.vals, theta.vals, post.probs.50)

levels1 = seq(.1,1,by = .05)

contour(n.vals, theta.vals, probs.post,
        levels=c(levels1,.92,.93),
        xlim = c(0,32), ylim = c(0,1),
        xlab=expression(n[0]), ylab=expression(theta[0] ), main=expression(P({theta > .5} *"|"* {sum(Y)
```



Each of the lines shows the conditions of n_0 and θ to produce the given probability. In this case, it is reasonable to assume that $\theta > .5$ since the plot is dominated by lines that of relatively high probability. The lines represent the probability that $\theta > .5$. It is important to note that under these conditions θ represents our prior guess of θ and n_0 represents our confidence in this guess. From this plot, we can see that when we don't have a high prior confidence, the plots with a lower guess of θ don't even show up hardly. Also, We observed $\sum Y = 57$, and so the probability .92 is actually relatively flat because it seems to start with the prior guess of $\theta \approx .55$, which is very close to our observed values. Under these conditions, it seems likely that $\theta > .5$