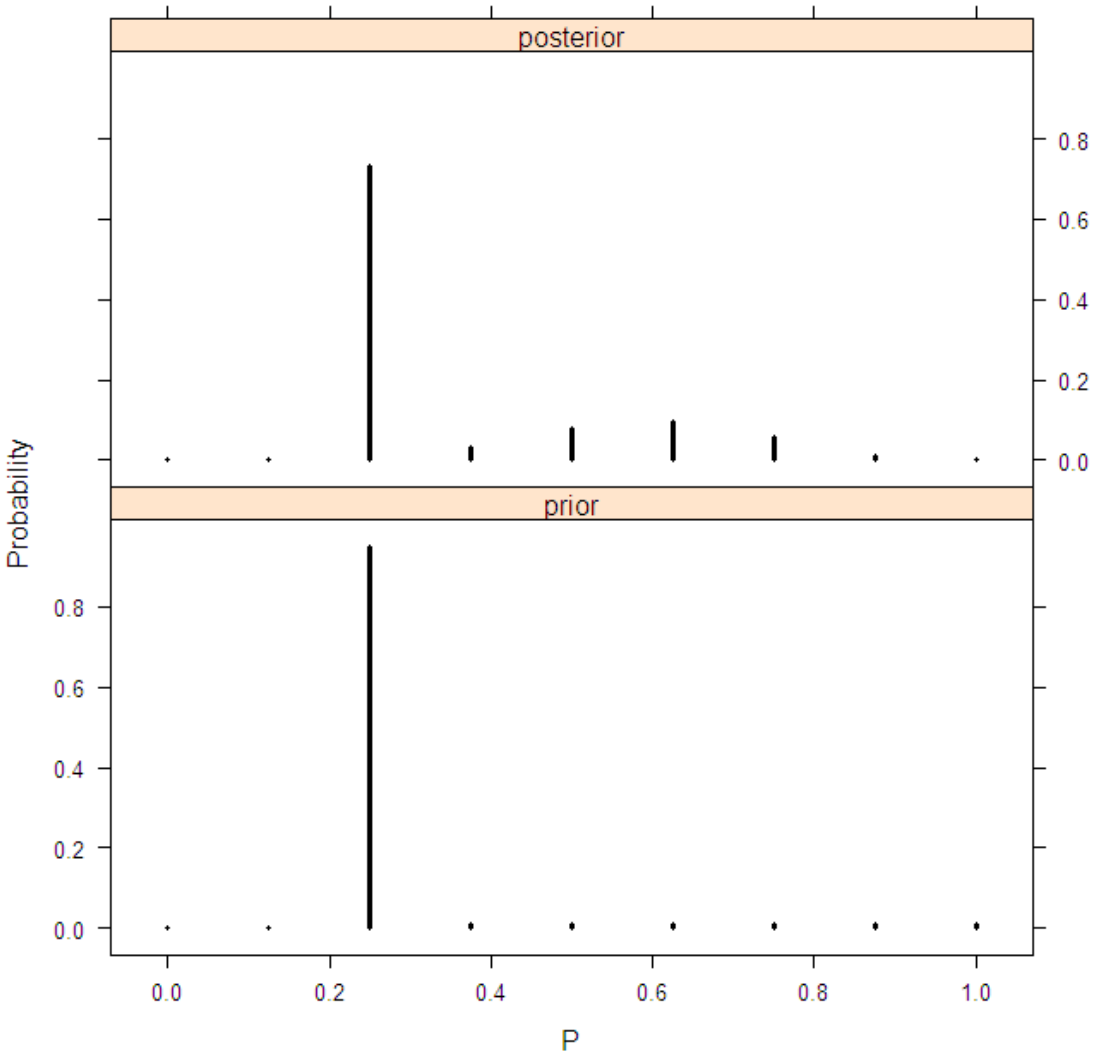
HOMEWORK 2 ANSWERS

2.1.

```
#
# Chapter 2 -- Bayesian Computation With R
#              Demonstrates Priors
#
#  Remove all objects just to be safe
#
rm(list=ls(all=TRUE))
#
library(LearnBayes)
library(lattice)
#
# Probability of Bob NOT Having ESP
#   He is correct 4 times and incorrect 6 times
#   Because given what you observe, he beats the odds .4 > .25
#   Hence, your Posterior probability of Bob NOT having ESP
#   MUST GO DOWN!
#
data = c(6, 4)
#
# Prior Beliefs --
#
pesp <- c(0, .125, .250, .375, .500, .625, .750, .875, 1.000)
priorpesp <- c(.001, .001, .950, .008, .008, .008, .008, .008, .008)
#
#  Posterior Distribution
#
postesp <- pdisc(pesp, priorpesp, data)
#
#  Round to 3 decimal places
#
xx <- round(cbind(pesp, priorpesp, postesp),3)
#
PRIOR=data.frame("prior",pesp,priorpesp)
POST=data.frame("posterior",pesp,postesp)
names(PRIOR)=c("Type","P","Probability")
names(POST)=c("Type","P","Probability")
data=rbind(PRIOR,POST)
yy <-
xyplot(Probability~P|Type,data=data,layout=c(1,2),type="h",lwd=3,col="b
lack")
plot(yy)
```

	pesp	priorpesp	postesp
[1,]	0.000	0.001	0.000
[2,]	0.125	0.001	0.000
[3,]	0.250	0.950	0.730
[4,]	0.375	0.008	0.034
[5,]	0.500	0.008	0.078
[6,]	0.625	0.008	0.094
[7,]	0.750	0.008	0.055
[8,]	0.875	0.008	0.009

[9,] 1.000 0.008 0.000



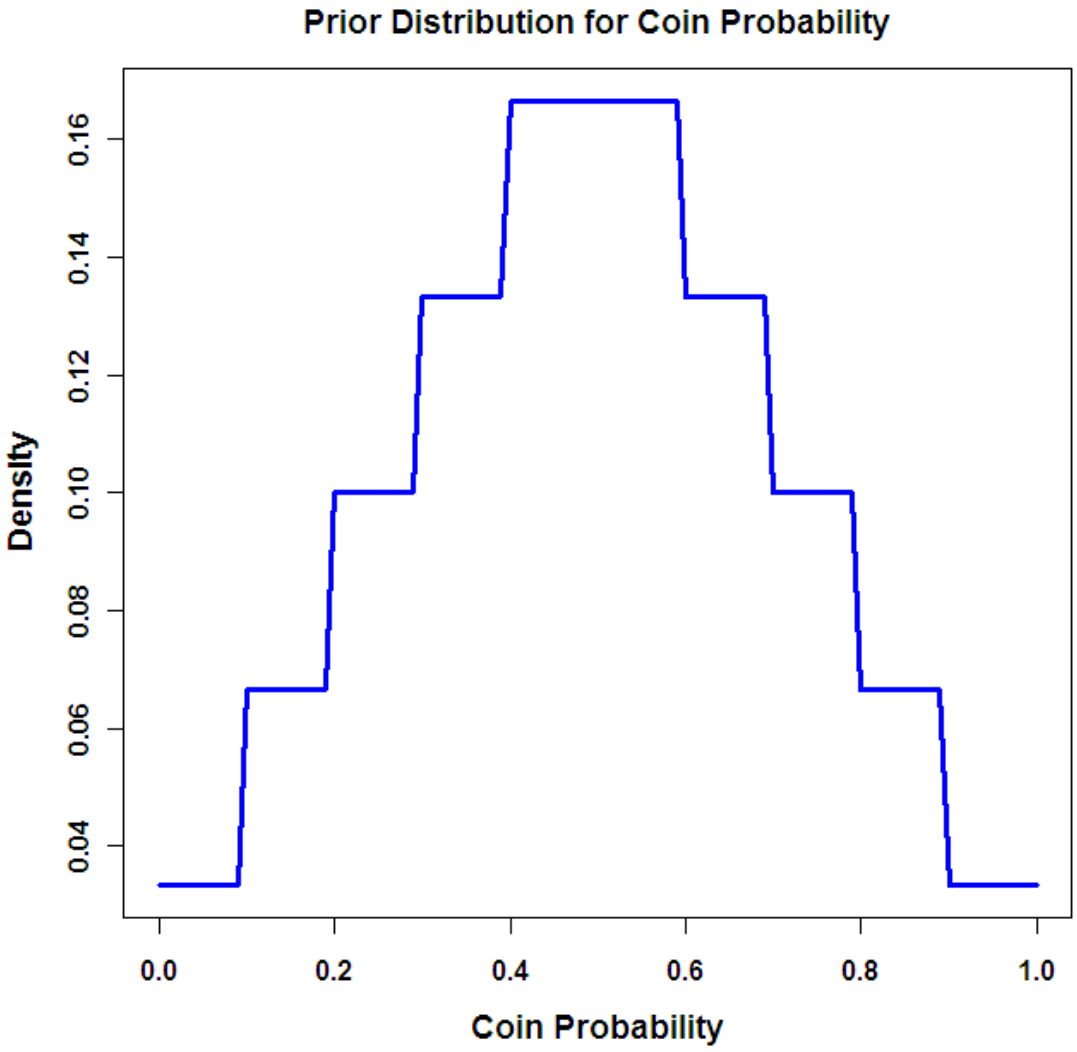
2.2.

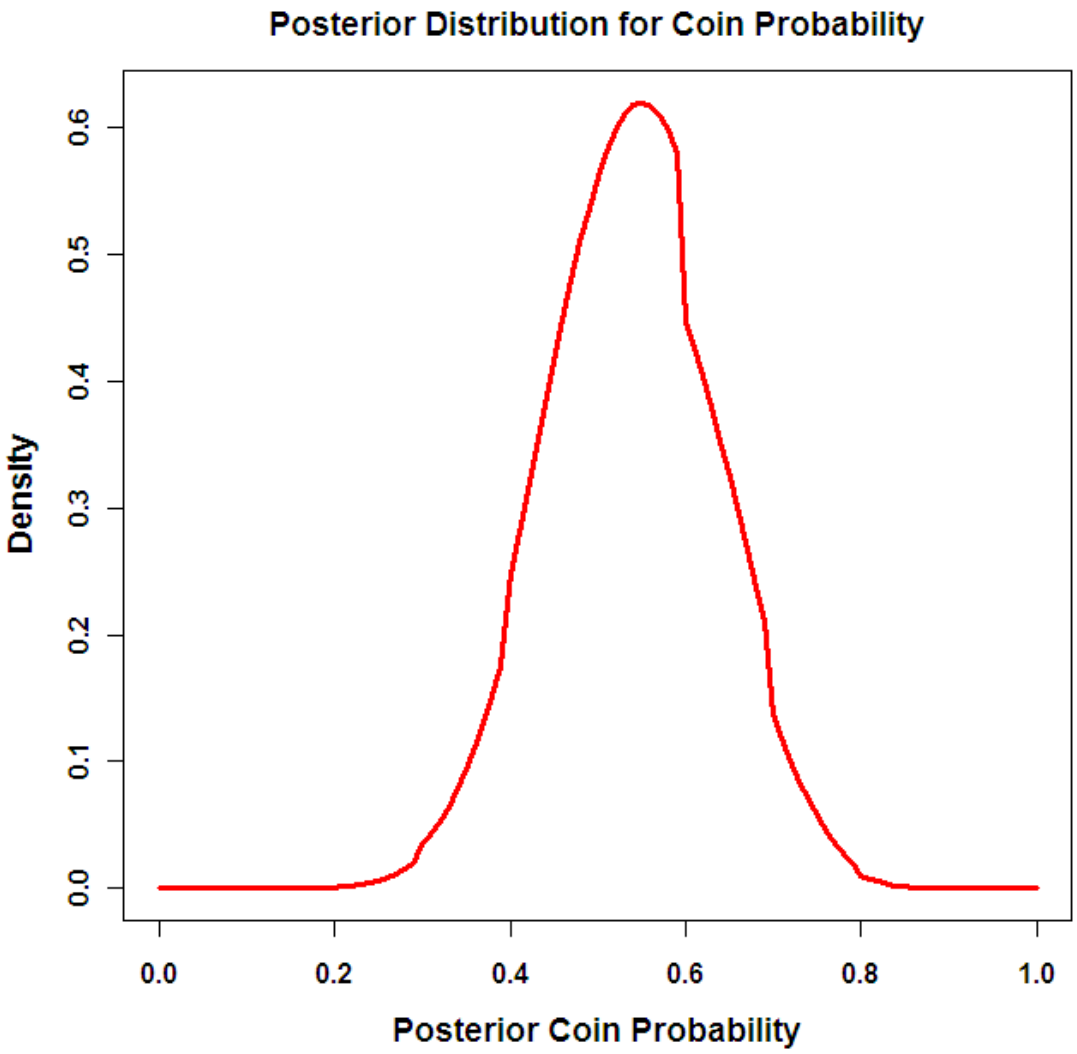
```
#
# Chapter 2 -- Bayesian Computation With R
#              Demonstrates Priors
#
#  Remove all objects just to be safe
#
rm(list=ls(all=TRUE))
#
library(LearnBayes)
library(lattice)
#
midpoints <- seq(0.05, 0.95, by=0.1)
#
#  Make up your own Prior -- I am putting the weight around
#    .4, .5, .6
```

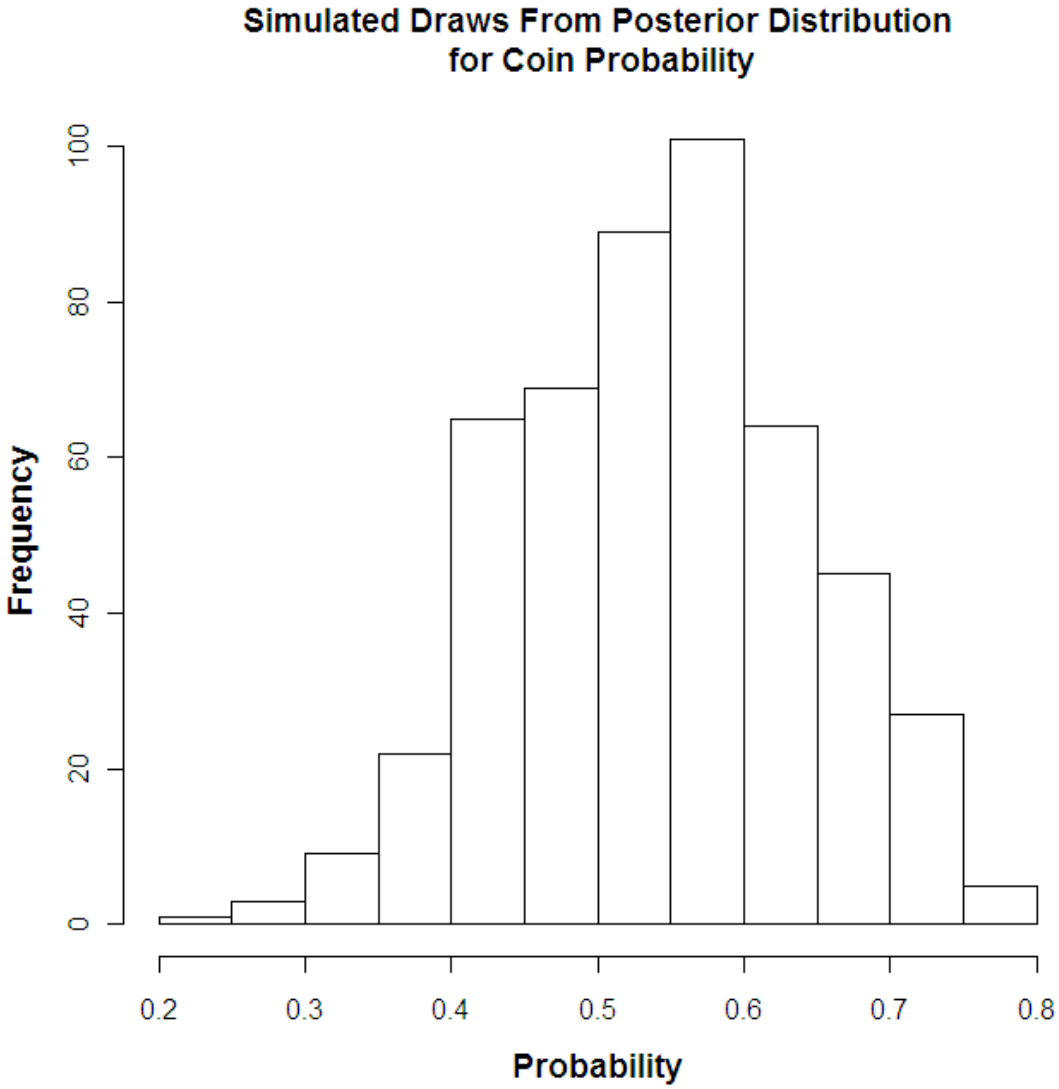
```
#
priorprob <- c(1,2,3,4,5,5,4,3,2,1)
priorprob <- priorprob/sum(priorprob)
#
#  Call histprior to get vector of probabilities for the intervals
#     (See page 27)
#
curve(histprior(x,midpoints,priorprob),from=0,
to=1,xlab="",ylab="",lty=1,lwd=3,font=2,col="blue")
# Main title
mtext("Prior Distribution for Coin
Probability",side=3,line=1.00,cex=1.2,font=2)
# x-axis title
mtext("Coin Probability",side=1,line=2.75,font=2,cex=1.2)
# y-axis title
mtext("Density",side=2,line=2.75,font=2,cex=1.2)
#
#  Experiment -- 11 Heads and 9 Tails
#
windows()
#
curve(histprior(x,midpoints,priorprob)*dbeta(x,12,10),from=0,
to=1,xlab="",ylab="",lty=1,lwd=3,font=2,col="red")
#
# Main title
mtext("Posterior Distribution for Coin
Probability",side=3,line=1.00,cex=1.2,font=2)
# x-axis title
mtext("Posterior Coin Probability",side=1,line=2.75,font=2,cex=1.2)
# y-axis title
mtext("Density",side=2,line=2.75,font=2,cex=1.2)
#
#  Sampling -- see page 28
#
p <- seq(0, 1, length=500)
postprob <- histprior(p,midpoints,priorprob)*dbeta(p,12,10)
postprob <- postprob/sum(postprob)
#
psample <- sample(p, replace = TRUE, prob = postprob)
windows()
hist(psample,xlab="",ylab="",main="")
#
# Main title
mtext("Simulated Draws From Posterior Distribution\n for Coin
Probability",side=3,line=1.00,cex=1.2,font=2)
# x-axis title
mtext("Probability",side=1,line=2.75,font=2,cex=1.2)
# y-axis title
mtext("Frequency",side=2,line=2.75,font=2,cex=1.2)
```

```
> midpoints
 [1] 0.05 0.15 0.25 0.35 0.45 0.55 0.65 0.75 0.85 0.95
```

```
> priorprob
 [1] 0.03333333 0.06666667 0.10000000 0.13333333 0.16666667 0.16666667
 [7] 0.13333333 0.10000000 0.06666667 0.03333333
```




Posterior Distribution for Coin Probability
Density
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.0
0.2
0.4
0.6
0.8
1.0
Posterior Coin Probability



2.3.

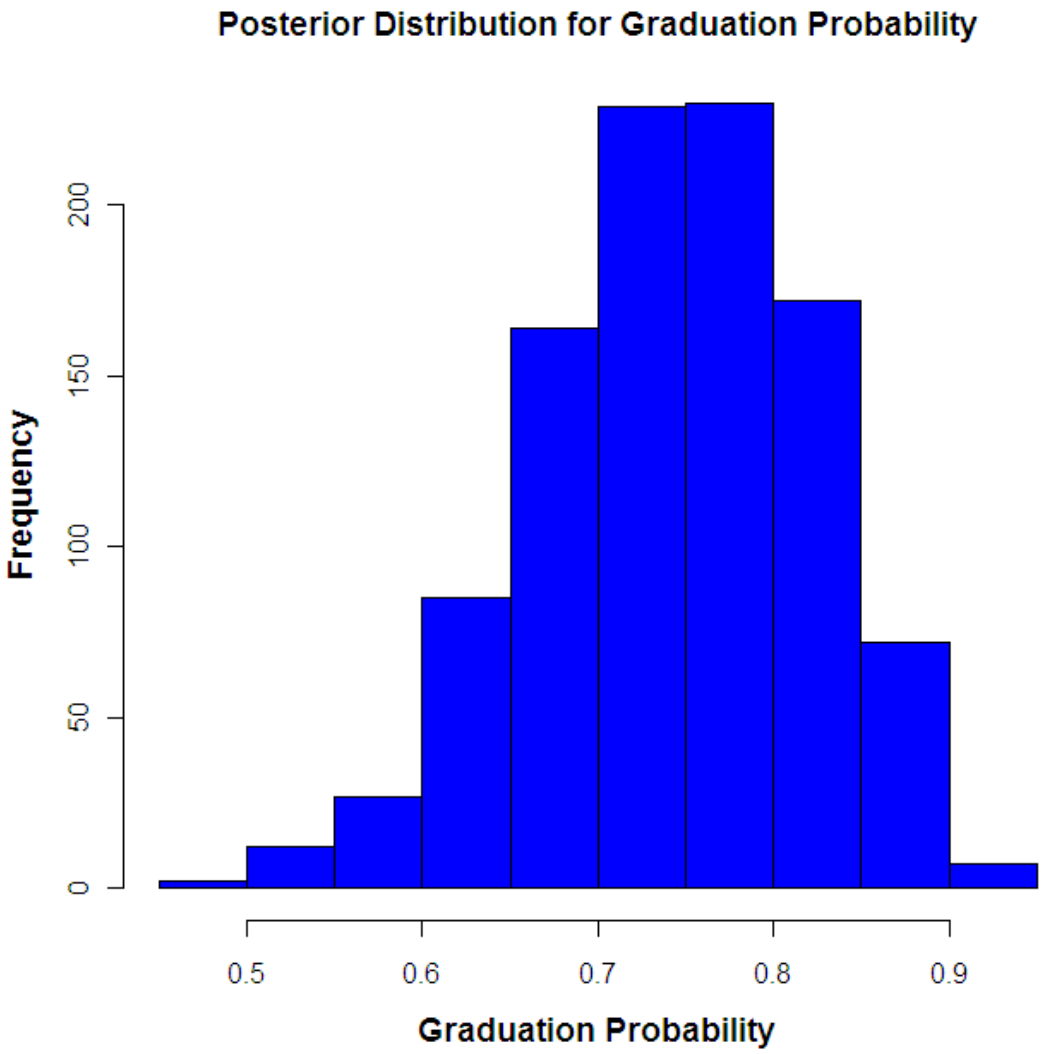
```
  #
# Chapter 2 -- Bayesian Computation With R
#              Demonstrates Priors
#
#  Remove all objects just to be safe
#
rm(list=ls(all=TRUE))
#
library(LearnBayes)
library(lattice)
#
# Lead Content -- 22 graduated, 7 did not
#
#   Prior = Beta(1, 1)
#   Posterior = Beta(22+1,7+1) = Beta(23,8)
#
#    See page 25
```

```
#
xx <- qbeta(c(0.05, 0.95), 23, 8)
#
#  Cumulative Probability P(X<=x) is pbeta(x, alpha, beta)
#    See page 24
#
yy <- 1.0 - pbeta(0.6,23,8)
#
#    See page 25-26
#
sampbeta <- rbeta(1000,23,8)
hist(sampbeta,col="blue",xlab="",ylab="",main="")
# Main title
mtext("Posterior Distribution for Graduation
Probability",side=3,line=1.00,cex=1.2,font=2)
# x-axis title
mtext("Graduation Probability",side=1,line=2.75,font=2,cex=1.2)
# y-axis title
mtext("Frequency",side=2,line=2.75,font=2,cex=1.2)
#
#  Prediction  10 more children with high lead content
#  See page 31-32
#
ynew <- rbinom(1000,10,sampbeta)
freq <- table(ynew)
ys <- as.integer(names(freq))
predprob <- freq/sum(freq)
windows()
plot(ys,predprob,type="h",xlab="",ylab="",main="")
# Main title
mtext("Predictive Distribution",side=3,line=1.00,cex=1.2,font=2)
# x-axis title
mtext("Predictive Probability",side=1,line=2.75,font=2,cex=1.2)
# y-axis title
mtext("YNEW",side=2,line=2.75,font=2,cex=1.2)
```

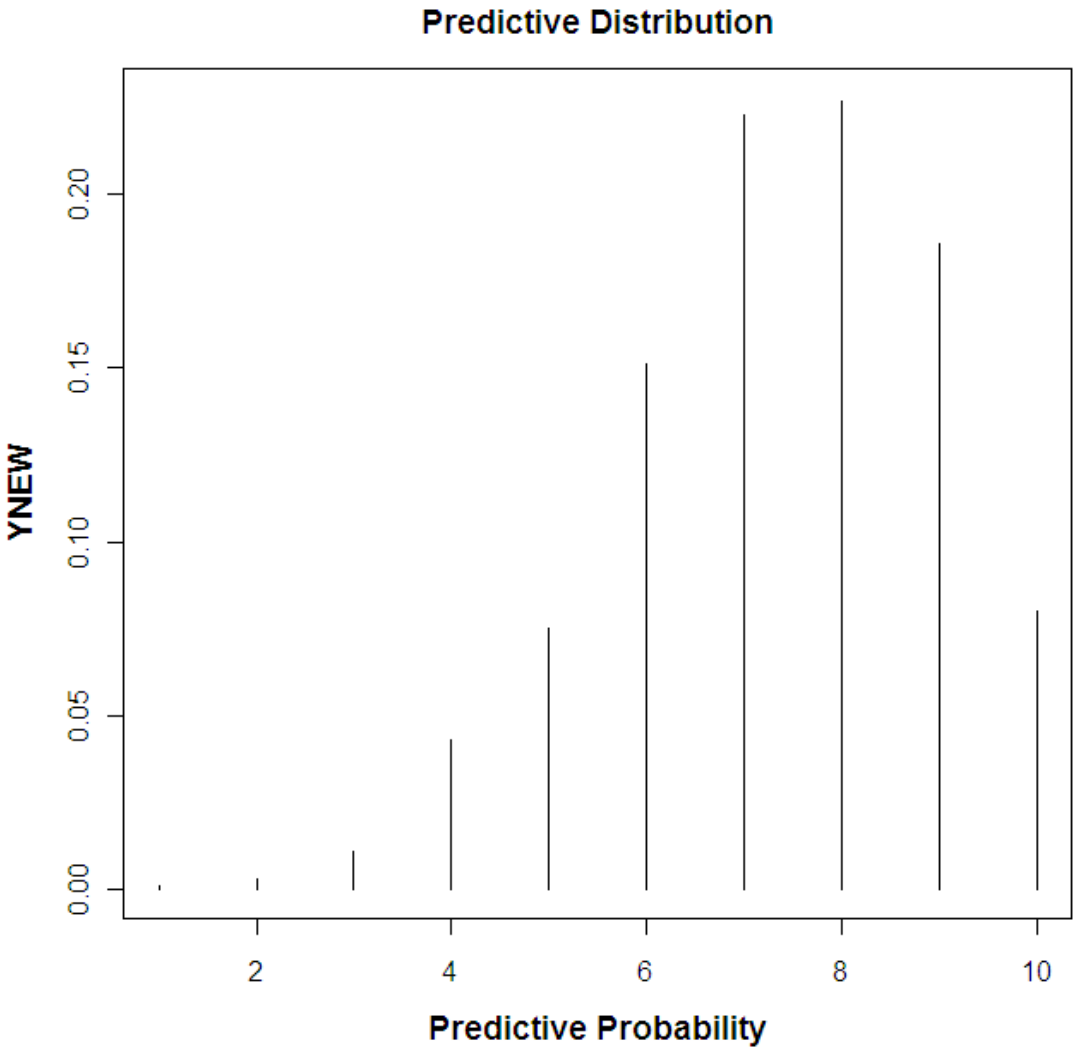
a)
[1] 0.6060526 0.8598149

b)
[1] 0.9564759

c)


Posterior Distribution for Graduation Probability
Frequency
0
50
100
150
200
0.5
0.6
0.7
0.8
0.9
Graduation Probability

d)
I get 0.266



2.4.

```
#
# Chapter 2 -- Bayesian Computation With R
#              Demonstrates Priors
#
#  Remove all objects just to be safe
#
rm(list=ls(all=TRUE))
#
library(LearnBayes)
library(lattice)
#
#  Joe's Prior
#
prob <- c(0.1, 0.2, 0.3, 0.4, 0.5)
joeprior <- c(0.5, 0.2, 0.2, 0.05, 0.05)
joexfx <- prob*joeprior
```

```
joemean <- sum(joexfx)
joex2fx <- prob*prob*joeprior
joex2sum <- sum(joex2fx)
varjoe <- joex2sum - joemean*joemean
sdjoe <- sqrt(varjoe)
#
#  Use Beta Mean and Variance Fromula -- page 577 Gelman
#
sammean <- 3/(3+12)
samvar <- (3*12)/((3+12)*(3+12)*(3+12+1))
sdsam <- sqrt(samvar)
#
#  Page 30 - 31 -- Joe's Prediction
#
msize <- 12
mysize <- 0:12
pred <- pdiscp(prob, joeprior, msize, mysize)
xx <- round(cbind(0:12,pred),3)
#
#  Page 31 -- Sam's Prediction
#
ab <- c(3, 12)
predsam <- pbetap(ab,msize,mysize)
yy <- round(cbind(0:12,predsam),3)
```

a) > joemean
[1] 0.195
> sdjoe
[1] 0.1160819
>
> sammean
[1] 0.2
> sdsam
[1] 0.1
>
b)
Joe's Prediction:
> xx

	pred
[1,]	0 0.158
[2,]	1 0.245
[3,]	2 0.209
[4,]	3 0.148
[5,]	4 0.100
[6,]	5 0.065
[7,]	6 0.039
[8,]	7 0.021
[9,]	8 0.010
[10,]	9 0.004
[11,]	10 0.001

[12,] 11 0.000
[13,] 12 0.000

Sam's Prediction:
> yy
predsam
[1,] 0 0.140
[2,] 1 0.219
[3,] 2 0.219
[4,] 3 0.174
[5,] 4 0.117
[6,] 5 0.069
[7,] 6 0.036
[8,] 7 0.016
[9,] 8 0.006
[10,] 9 0.002
[11,] 10 0.001
[12,] 11 0.000
[13,] 12 0.000

2.5.

```
#
# Chapter 2 -- Bayesian Computation With R
#              Demonstrates Priors
#
#  Remove all objects just to be safe
#
rm(list=ls(all=TRUE))
#
library(LearnBayes)
library(lattice)
#
# Mean Snowfall levels (in inches)
#
mu <- c(20, 30, 40, 50, 60, 70)
#
# Prior probabilities of the corresponding snowfall levels
#
gmu <- c(.1, .15, .25, .25, .15, .1)
#
#  Actual Observed Yearly Snowfall totals
#
y <- c(38.6, 42.4, 57.5, 40.5, 51.7, 67.1, 33.4, 60.9, 64.1, 40.1,
40.7, 6.4)
ybar <- mean(y)
vary <- var(y)
sdy <- sd(y)
n <- 12
#
```

```
#  Likelihood of the 12 Means using Actual Mean and Variance of
Observed
#   snowfall totals  (Computed two ways as a check)
#
like <- exp(-(n/(2*vary))*(mu-ybar)*(mu-ybar))
like2 <- exp(-(n/(2*sdy*sdy))*(mu-ybar)*(mu-ybar))
#
#  Posterior Probabilities for the 6 possible means
#
post <- (gmu*like)/sum(gmu*like)
#
#  Find 80% Probability Interval for the mean snowfall
#    using the function discint(dist, prob) -- see page 34
#    for description of arguments
#
dist <- cbind(mu,post)
probinterval <- discint(dist,0.8)
```

a)
mu <- c(20, 30, 40, 50, 60, 70)
gmu <- c(.1, .15, .25, .25, .15, .1)

b) > ybar
[1] 45.28333

c) Done two ways as a check:

> like
[1] 9.442011e-07 6.287729e-03 5.456483e-01 6.170515e-01 9.093262e-03
[6] 1.746255e-06
> like2
[1] 9.442011e-07 6.287729e-03 5.456483e-01 6.170515e-01 9.093262e-03
[6] 1.746255e-06
>

d)
> post
[1] 3.222723e-07 3.219168e-03 4.655982e-01 5.265262e-01 4.655534e-03
[6] 5.960273e-07

e)
> dist

	mu	post
[1,]	20	3.222723e-07
[2,]	30	3.219168e-03
[3,]	40	4.655982e-01
[4,]	50	5.265262e-01
[5,]	60	4.655534e-03
[6,]	70	5.960273e-07

Probability =
> probinterval

$prob
[1] 0.9921244

$set
[1] 40 50