

Bayes: Homework 4

Leslie Gains-Germain

Problem 1

1. I chose a $Gam(20, 1)$ prior for λ , and the sum of the x 's was 409 in the fake data that I generated. The posterior distribution of $\lambda|x$ is then $Gam(429, 21)$. The code is included in the appendix, and the quantities of interest are shown in the following table for different sample sizes.

Sample Size	90% posterior interval	$Pr(10 < x < 20)$	$Pr(x < 5)$
10	(19.51, 20.88)	0.50	0
50	(19.02, 22.80)	0.26	0
100	(18.57, 22.01)	0.31	0
250	(18.90, 22.13)	0.30	0
1000	(18.78, 22.03)	0.36	0
10000	(18.81, 22.08)	0.34	0
Theoretical Distribution	(18.84, 22.08)	0.34	≈ 0

The posterior interval and posterior probabilities get closer to the theoretical quantities as the sample size increases. At a sample size of 10, the posterior interval and probabilities can change dramatically if you get one unusual draw. For sample sizes less than 1000, the quantities also can change depending on the draws obtained. For example, the posterior interval at a sample size of 50 is wider than 10 because I obtained draws in the tails of the distribution. At a sample size of 10, only the more probable values in the middle of the distribution are obtained. This could be reversed, however, if I happened to draw an unusual value in the sample of size 10. At a sample of size 10000, the quantiles and probabilities are very similar to those from the theoretical distribution. I think the main point is that if you are trying to characterize a posterior distribution by pulling random draws from it, you need a large sample size (10000 or more!) to appropriately characterize the distribution.

2. Below I obtain 1000 draws from a $Gam(429, 21)$ using the grid approximation method. I used grid sizes of 0.5, 0.1, 0.01, 0.001 and 0.00001. It looks like the grid size of 0.5 gave the worst

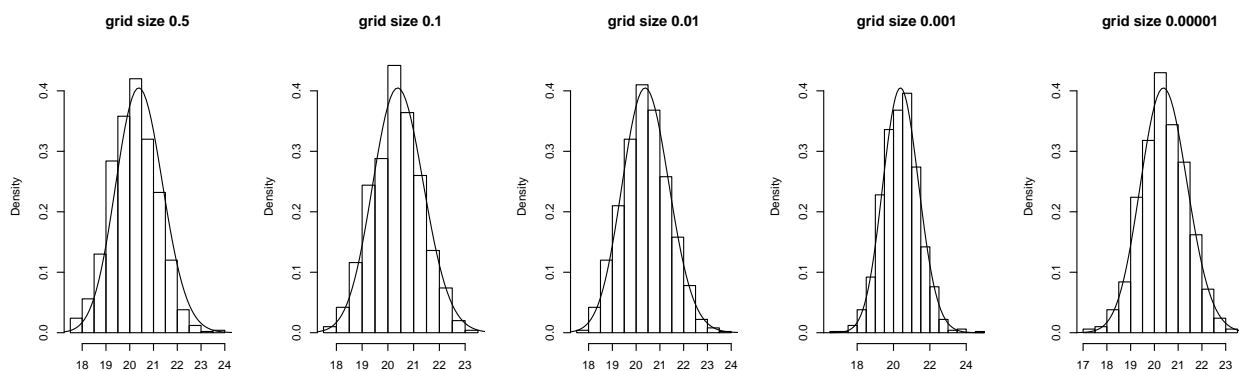
approximation because values less than 21 were overrepresented and values greater than 21 were underrepresented overall. At grid sizes of 0.1 and 0.01, the approximation still looks shifted left a little bit with values less than 21 overrepresented (although not as bad as the 0.5 case). It looks like the approximation was fairly equivalent at grid sizes of 0.001 and 0.00001. In practice, I would use a grid size of 0.001 because it looks like you get the best compromise between computing power and the quality of the approximation.

```
#function to calculate posterior probabilities
posterior <- function(x){
  dgamma(x, 429, 21)
}

#possible x values
#use small grid size
xsmall <- seq(15, 25, .00001)

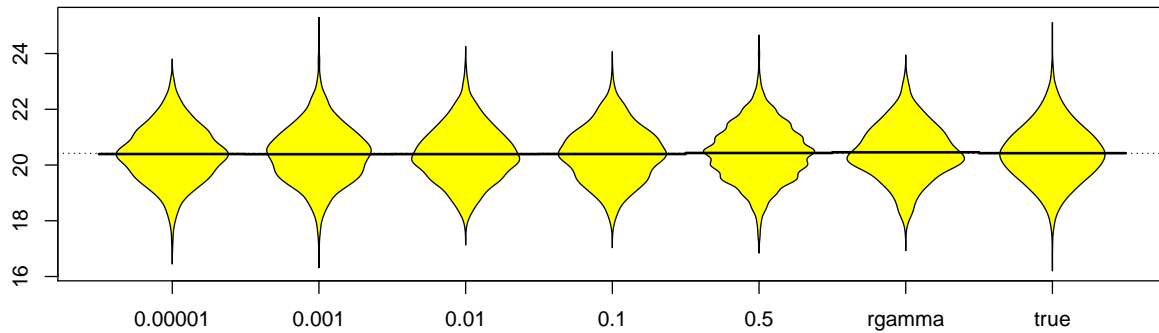
##sample x with probability proportional to height of function within that grid
set.seed(17)
grid.draws <- sample(xsmall, 1000, prob=posterior(xsmall))

#plot draws with posterior overlaid
#hist(grid.draws, freq=F, main="grid size 0.00001")
#lines(xsmall, posterior(xsmall))
```



3. The beanplots below show the distribution of 1000 draws for different grid sizes, as well as the draws from `rgamma` and the theoretical distribution. I think it is hard to tell how good the approximation is by just looking at the beanplots. The table, however, shows that the

posterior interval and posterior probability is closest to the truth at grid sizes of 0.001 and 0.00001. `rgamma` performs similarly to these grid sizes as well.



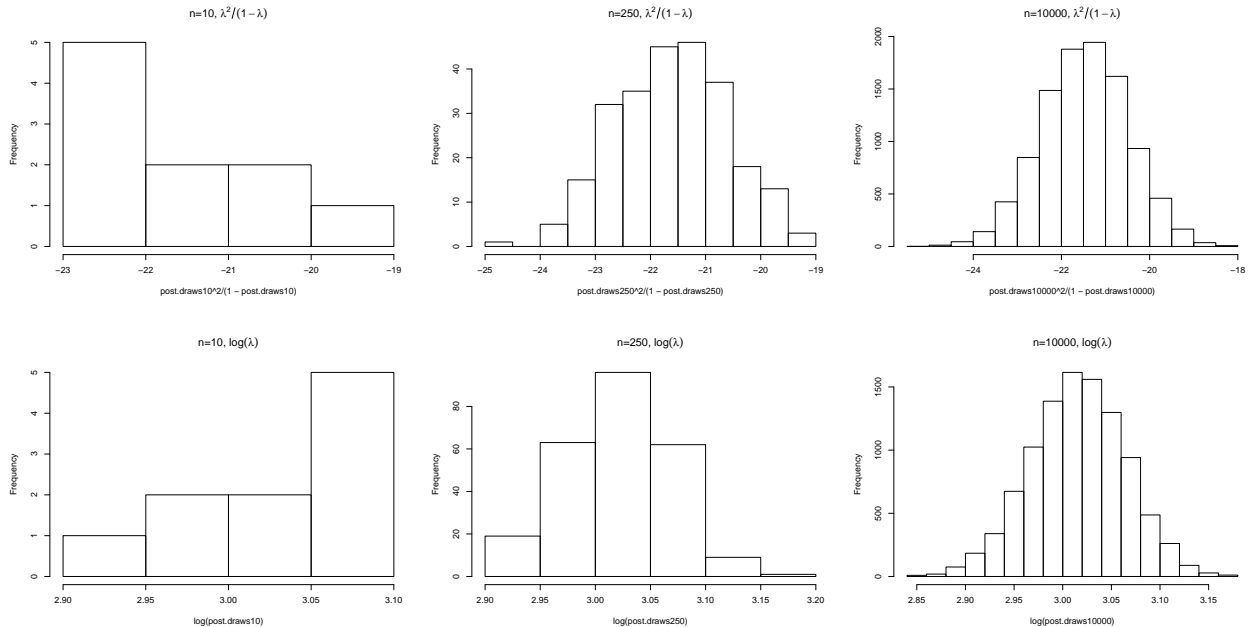
Grid Size	90% posterior interval	$Pr(10 < x < 20)$	$Pr(x < 5)$
0.5	(19.0, 22.0)	0.44	0
0.1	(18.80, 22.10)	0.36	0
0.01	(18.75, 21.97)	0.35	0
0.001	(18.81, 22.02)	0.33	0
0.00001	(18.83, 22.16)	0.32	0
rgamma	(18.86, 22.07)	0.36	0
Theoretical Distribution	(18.84, 22.08)	0.34	≈ 0

- I obtained 100 draws using five different `set.seeds`. For each, I characterized the distribution from which the draws came (a $Gam(429, 21)$). The mean varied between 20.4 and 20.6, and the variance was between 0.92 and 1.12 (the true mean is 20.43 and the true variance is 0.97). I think in many situations, it is adequate to be within 0.2 of the mean and the variance. So, yes, I think it could be *adequate* to characterize a distribution with only 100 observations. But, it is important to keep in mind the variability we might see in the characteristics of a distribution among different draws of 100.

set.seed	90% posterior interval	$Pr(10 < x < 20)$	$Pr(x < 5)$	mean	SD
set.seed(15)	(18.57, 22.01)	0.31	0	20.52	0.97
set.seed(16)	(19.06, 21.95)	0.33	0	20.51	0.97
set.seed(17)	(19.04, 22.05)	0.36	0	20.41	0.92
set.seed(18)	(19.12, 21.96)	0.35	0	20.52	0.94
set.seed(19)	(19.24, 21.93)	0.30	0	20.59	0.93
set.seed(20)	(18.55, 22.07)	0.31	0	20.45	1.12
Theoretical Distribution	(18.84, 22.08)	0.34	≈ 0	20.43	0.97

5. Using the code from (1), I display the histogram of transformed draws for sample sizes of 10, 250, and 10000. I used the sample size of 10000 to approximate the mean and variance of these functions of λ . I am really excited about this, because I don't need to do a mathematical transformation to find information about the distribution for a function of λ !!! And I don't need to use the Delta Method to approximate the variance of the distribution for a function of λ ! I can simply draw from the distribution of λ , calculate the desired function of λ , and use this to approximate the distribution for the function of λ (along with mean, variance, and other summary measures).

Transformation	Estimated mean	Estimated Variance
$\frac{\lambda^2}{1-\lambda}$	-21.47	0.97
$\log(\lambda)$	3.02	0.0024



6. The posterior predictive distribution, $p(\tilde{y}|y)$ is found by integrating the joint distribution of \tilde{y}, λ , and y over λ . Since y and \tilde{y} are independent given λ , it turns out that:

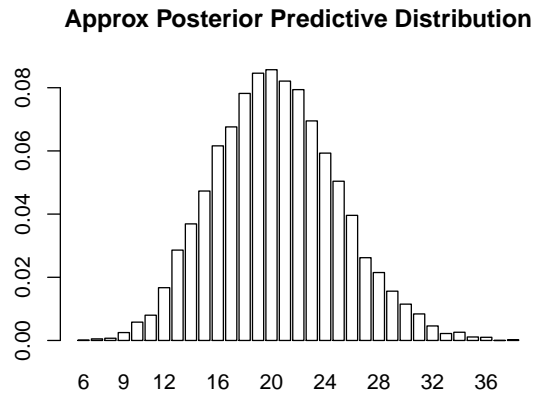
$$\begin{aligned} p(\tilde{y}|y) &= \int_{\lambda} p(\tilde{y}, \lambda, y) d\lambda = \int_{\lambda} p(\tilde{y}|\lambda, y) p(\lambda|y) d\lambda \\ &= \int_{\lambda} p(\tilde{y}|\lambda) p(\lambda|y) d\lambda \end{aligned}$$

Recall that in HW3 that the sum of the 20 observations drawn from a $Poi(20.66)$ distribution was 409. We then showed that $\lambda|y \sim Gam(429, 21)$.

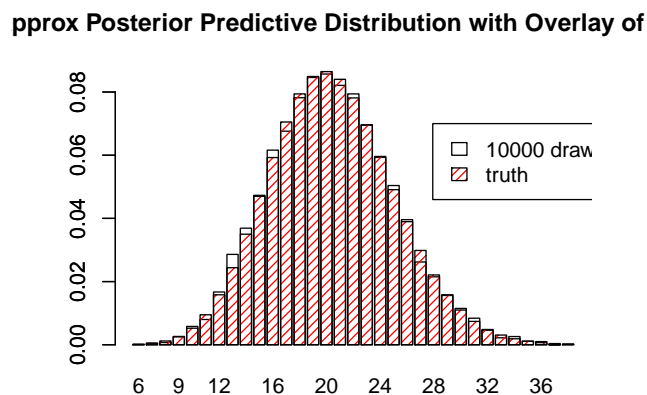
$$\begin{aligned} p(\tilde{y}|y) &= \int_{\lambda} \frac{\lambda^{\tilde{y}} e^{-\lambda}}{\tilde{y}!} \frac{\lambda^{428} e^{-21\lambda} 21^{429}}{\Gamma(429)} d\lambda \\ &= \int_{\lambda} \frac{\lambda^{\tilde{y}+428} e^{-22\lambda} 21^{429}}{\Gamma(\tilde{y}+1)\Gamma(429)} d\lambda \\ &= \frac{21^{429}}{\Gamma(\tilde{y}+1)\Gamma(429)} \int_{\lambda} \lambda^{\tilde{y}+428} e^{-22\lambda} d\lambda \\ &= \frac{\Gamma(\tilde{y}+429) 21^{429}}{\Gamma(\tilde{y}+1)\Gamma(429) 22^{\tilde{y}+429}} \\ &= \binom{\tilde{y}+429-1}{\tilde{y}} \frac{21^{429}}{22} \frac{1}{22} \frac{\tilde{y}}{22} \end{aligned}$$

The posterior predictive distribution is a Negative Binomial distribution with parameters 429 and 21, $Neg - bin(429, 21)$.

7. We must assume that future observations are exchangeable with the original data collected. This means that future observations come from the same design and the same process (class notes 9/23), and will be within the same range as the data collected.
8. The following shows a barplot of 10000 draws from the $Neg - bin(429, 21)$ distribution.



9. The following shows the true posterior predictive distribution overlaid on the barplot of 10000 draws from the previous part.



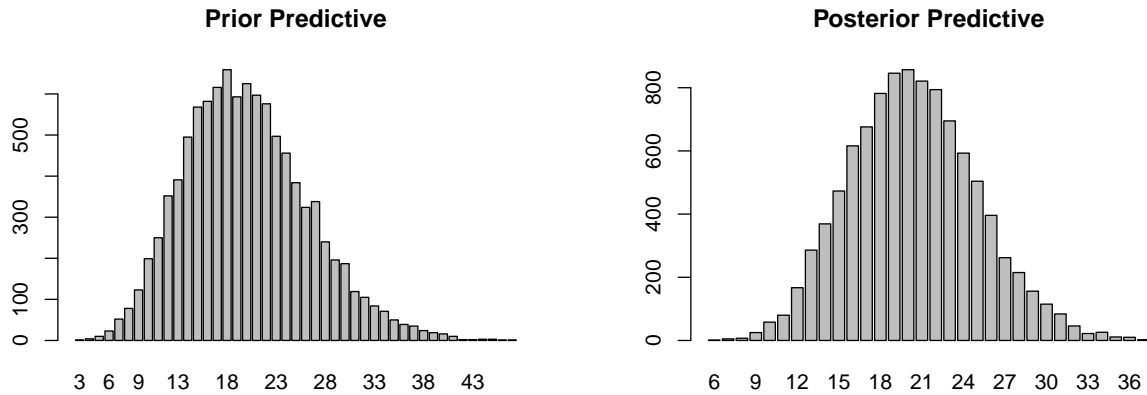
10. The mean of the posterior predictive distribution is $429/21 = 20.43$ and the variance of the posterior predictive distribution is $429/21^2(22) = 21.40$. The mean of the posterior distribution of λ ($Gam(429, 21)$) is $429/21 = 20.43$ and the variance of the posterior distribution is $429/21^2 = 0.9728$.

I am going to explain this to Chris, mine and Bobby's consulting client from last semester. He has taken 512, so he has discussed a similar concept in the context of prediction intervals vs.

confidence intervals. The posterior distribution for the parameter λ describes our knowledge about the possible values of λ after incorporating knowledge from the observed data. The mean of the posterior distribution is sometimes used as a point estimate for the parameter λ and the spread of the distribution gives us an idea of what values of λ we consider to be plausible. So, the larger the spread of the distribution, the more uncertainty there is about the true value of λ . The posterior predictive distribution, however, reflects two sources of uncertainty. The posterior predictive distribution reflects uncertainty in the true value of λ in addition to uncertainty in the value of a future observation. Even if λ was known, there is variability that comes from taking random draws from a known distribution. Combining the uncertainty in λ and the uncertainty in where the observations will fall around λ , the posterior predictive distribution has a larger variance. Our best guess for a future observation, however, still lies at our best guess for the mean of the posterior distribution, which is 20.43.

11. Recall that the prior I set for λ was $Gam(20, 1)$. Ignoring observed data, the prior predictive distribution for a single observation is $Neg - Bin(20, 1)$. I solve for the prior predictive distribution analytically below.

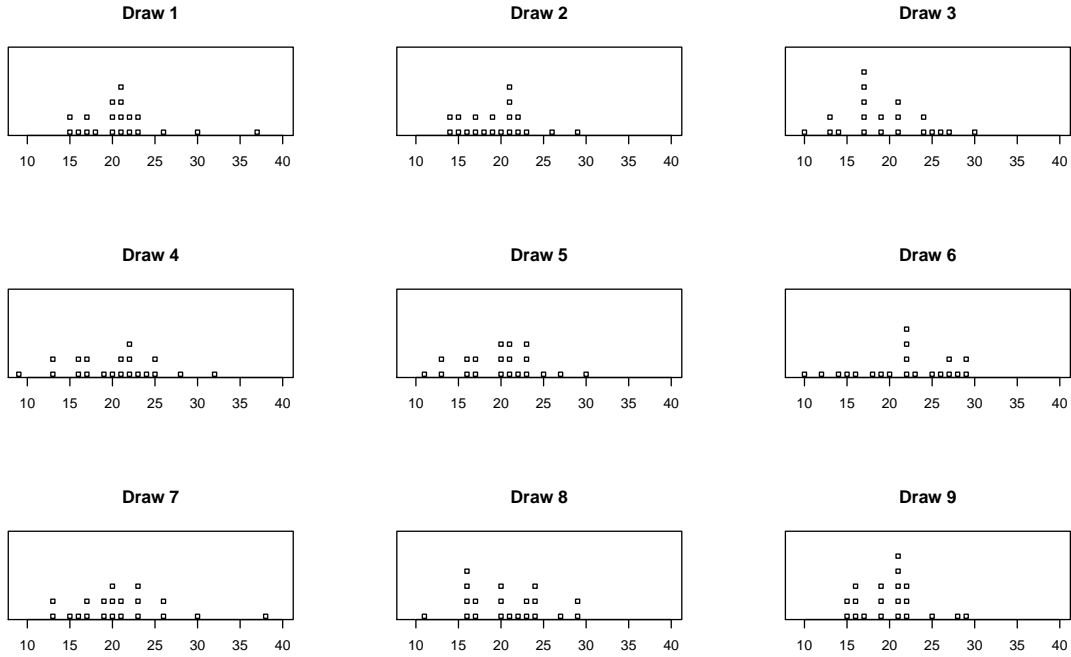
$$\begin{aligned}
p(y) &= \frac{p(y|\lambda)p(\lambda)}{p(\lambda|y)} \\
p(y) &= \frac{\frac{\lambda^y e^{-\lambda}}{y!} e^{-\beta\lambda} \beta^\alpha \lambda^{\alpha-1}}{\frac{\Gamma(\alpha)}{\lambda^{\alpha+y-1} e^{-\lambda(\beta+1)} (\beta+1)^{\alpha+y}}} \\
&= \frac{\lambda^{y+\alpha-1}}{\lambda^{\alpha+y-1}} \frac{\beta^\alpha}{(\beta+1)^{\alpha+y}} \frac{e^{-(\beta+1)\lambda}}{e^{-(\beta+1)\lambda}} \frac{\Gamma(\alpha+y)}{y! \Gamma(\alpha)} \\
&= \frac{\beta^\alpha}{(\beta+1)^{\alpha+y}} \frac{\Gamma(\alpha+y)}{\Gamma(\alpha) y!} \\
&= \binom{\alpha+y-1}{y} \left(\frac{\beta}{\beta+1}\right)^\alpha \left(\frac{1}{\beta+1}\right)^y \\
&= \binom{20+y-1}{y} \left(\frac{1}{1+1}\right)^{20} \left(\frac{1}{1+1}\right)^y
\end{aligned}$$



	90% interval	Estimated mean	Estimated Variance
Prior predicted	(11, 31)	19.99	6.32
Posterior predicted	(13, 28)	20.38	4.63

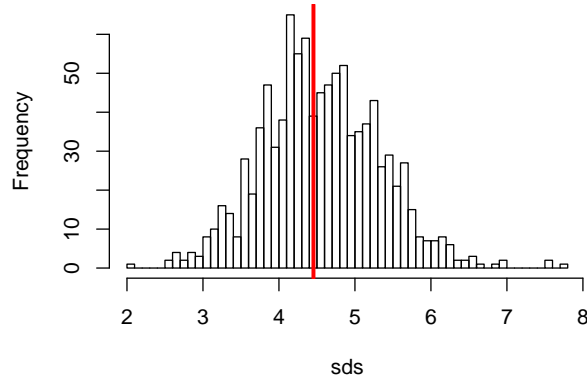
The main difference between the prior and posterior predictive distributions is in the spread. The prior predictive distribution has a larger variance and a wider range of plausible values than the posterior predictive distribution. The prior predictive distribution is slightly more skewed than the posterior predictive distribution.

12. I drew 1000 samples of size 20 from the posterior predictive distribution. 9 of them are shown below. I played around with using dotplots rather than barplots. The only thing I don't like about them is that I couldn't figure out how to get numbers on the y-axis.

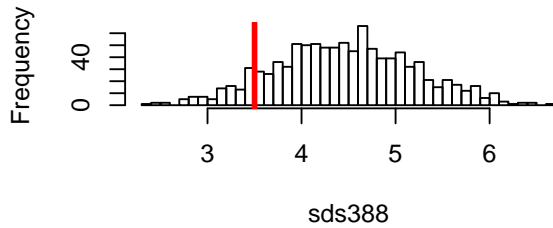


13. The approximate posterior predictive distribution for the sample standard deviation is shown below. The standard deviation observed in the original sample is in the center of the standard deviations found for the posterior predictive samples. I think the fact that the standard deviation from the original sample is right in the middle of the standard deviations from the posterior predictions is somewhat of a coincidence. We assume that the posterior predictions come from the same design and process as the original observations. This means that the summary measure of the original observed draws should be within the same range as what could be observed in the posterior predictions. I investigated further to see where the original sample standard deviation could lie in the posterior predictive distribution for the sample standard deviation. In the following plots, I generated a new sample of size 20 from a $Poi(20.66)$ distribution. I then found the corresponding posterior predictive distribution ($Neg - Bin(\sum x_i, 21)$), and then I constructed an approximate posterior predictive distribution for the sample standard deviation. The following plots show that the sample standard deviation from the original data can lie anywhere in the posterior predictive distribution.

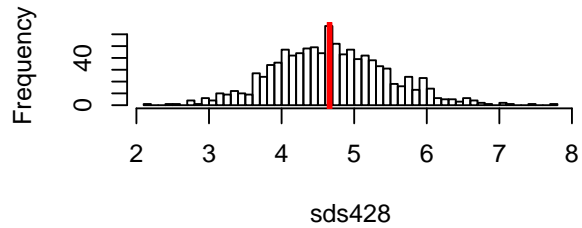
Approx Posterior Pred Dist for the Sample SD



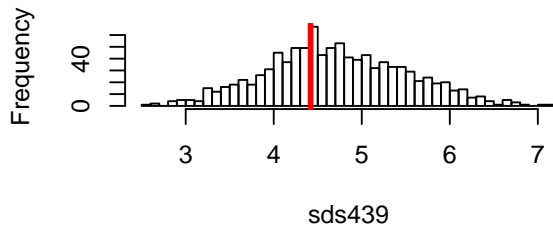
Approx Post Pred Dist for Sample SD



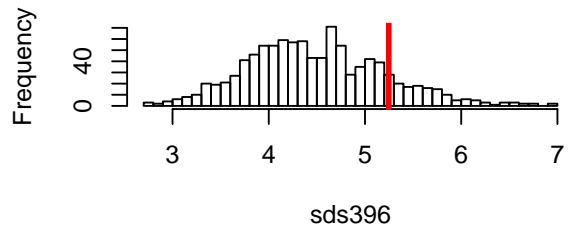
Approx Post Pred Dist for Sample SD



Approx Post Pred Dist for Sample SD

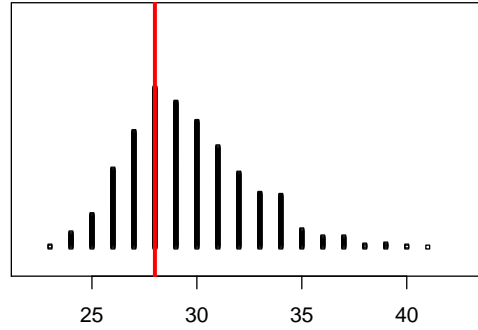


Approx Post Pred Dist for Sample SD

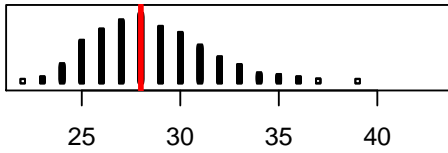


14. The approximate posterior predictive distribution for the sample maximum is shown below. The maximum observed in the original sample is in the center of the maximums found for the posterior predictive samples, but again I think this is just coincidence. I did a similar exploration as the previous problem. In the four samples I investigated, the location of the original sample maximum on the histogram of posterior predicted maximums varied; but the original maximum was always within the range of posterior predicted maximums.

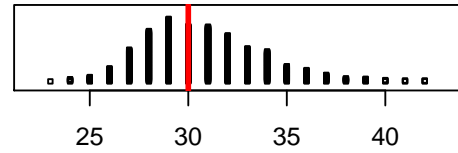
Approx Post Pred Dist for Sample Max



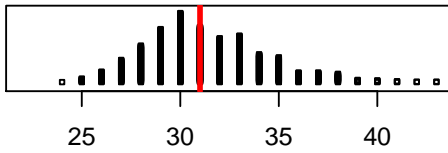
Approx Post Pred Dist for Sample Max



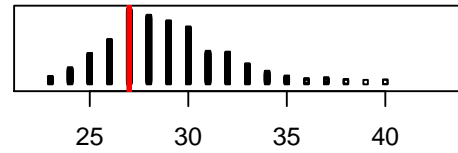
Approx Post Pred Dist for Sample Max



Approx Post Pred Dist for Sample Max



Approx Post Pred Dist for Sample Max



Problem 2:

Consider the regression model,

$$\log(\text{salary}_i) = \beta_0 + \beta_1 I_{\text{male}} + \epsilon_i$$

where $I_{\text{male}} = 1$ if male and 0 otherwise. In this regression model, β_1 represents the difference in salaries between males and females on the log scale. After backtransforming, e^{β_1} represents the

multiplicative difference in salaries between males and females. Gelman is thinking about what an appropriate prior for β_1 would be. If he puts a $Unif(-10, 10)$ prior on β_1 , then this implies that the multiplicative difference in salaries between males and females could range between $e^{-10} \approx \frac{1}{20000}$ and $e^{10} \approx 20000$, allowing for the possibility that either females have salaries 20000 times that of males, or males have salaries 20000 times that of females. These are natural constraints because it would be totally absurd for the gender effect on salary to be more extreme than that. Although an effect size of 10 doesn't seem very big on the log scale, it actually is big on the original scale.

We can think of a similar example on the logit scale. Consider the regression model,

$$\text{logit}(p_i) = \beta_0 + \beta_1 I_{male}$$

where $I_{male} = 1$ if male and 0 otherwise. For this example, suppose p_i represents the probability of going bankrupt. In this case, β_1 represents that odds ratio of going bankrupt between males and females. Suppose we put a $(-5, 5)$ prior on β_1 . This would imply that the odds ratio is somewhere between $e^{-5} \approx \frac{1}{150}$ and $e^5 \approx 150$. This would mean that the odds of females going bankrupt could be 150 times the odds of males going bankrupt, or the odds of males going bankrupt could be 150 times the odds of females going bankrupt. To explain what this means on the probability scale, I'll start by assuming the probability of females going bankrupt is 0.2. That means that the log odds of males going bankrupt is $\beta_1 + \log(0.2/0.8)$. If $\beta_1 = 5$, then $5 + \log(0.2/0.8) = 3.61$ and the probability of males going bankrupt is $e^{3.61}/(1 + e^{3.61}) = 0.97$. If $\beta_1 = -5$, then $-5 + \log(0.2/0.8) = -6.39$ and the probability of males going bankrupt is $e^{-6.39}/(1 + e^{-6.39}) = 0.0017$. I think Gelman is actually assuming that probability of females going bankrupt is 0.5. That means that the log odds of males going bankrupt is simply β_1 , and the probability of males going bankrupt is somewhere between $e^{-5}/(1 + e^{-5}) = 0.01$ and $e^5/(1 + e^5) = 0.99$. This would mean that the probability of males going bankrupt is somewhere between 0.49 lower or 0.49 higher than the probability of females going bankrupt.

Problem 3

1. If $y \sim \text{Expon}(\theta)$, then $P(y \geq 100|\theta) = e^{-100\theta}$ from the cdf of the exponential distribution.

Below I show that the posterior distribution of θ given $y \geq 100$ is $\text{Gam}(\alpha, \beta + 100)$. The mean is $\frac{\alpha}{\beta+100}$ and the variance is $\frac{\alpha}{(\beta+100)^2}$.

$$\begin{aligned} p(\theta|y \geq 100) &\propto e^{-100\theta} \theta^{\alpha-1} e^{-\beta\theta} \\ &\propto e^{-\theta(100+\beta)} \theta^{\alpha-1} \end{aligned}$$

2. Below I show that the posterior distribution is $\text{Gam}(\alpha + 1, \beta + 100)$. The posterior mean is now $\frac{\alpha+1}{\beta+100}$ and the posterior variance is now $\frac{\alpha+1}{(\beta+100)^2}$. So, if $y = 100$, the posterior mean and the posterior variance are larger than what they would have been if $y \geq 100$.

$$\begin{aligned} p(\theta|y = 100) &\propto \theta e^{-100\theta} \theta^{\alpha-1} e^{-\beta\theta} \\ &\propto e^{-\theta(100+\beta)} \theta^{\alpha+1-1} \end{aligned}$$

3. Yes, these results do surprise me. First, it surprises me that the posterior mean is higher when $y = 100$ compared to $y \geq 100$. If we observe lower values, shouldn't the estimate for θ be lower? Second, it surprises me that the posterior variance is higher when $y = 100$ compared to $y \geq 100$. If $y = 100$, we have more information about y , and more information generally leads to less uncertainty about θ . As a result, I would have expected the posterior distribution for θ to have *smaller* variance when $y = 100$ and *larger* variance when $y \geq 100$.

Problem 4

“Like Laplace, Fisher insisted that every probability statement depends on both the knowledge and ignorance that we bring to bear.” When I was at the ASA Alaska Chapter meeting, the room was full of biometricians from around the state. I had the opportunity to talk to many of them throughout the conference, and one of them told me that just *fifteen* years ago, nobody at Fish at Game ever reported confidence intervals (or other measures of uncertainty). They simply reported

averages! It's amazing me to that Fisher and Laplace came up with this concept of uncertainty over 100 years ago and that it took that long to infiltrate research in other disciplines (and it still hasn't in some disciplines).

The author of *Willful Ignorance* also talks about statements of probability never being absolutely true or false. But, he says probability statements generally line up with reality and can help inform decisions. This reminds me of the quote that is always thrown around when discussing models, "All models are wrong, some are useful." In statistics, we never actually believe that a model describing some effect is completely correct; we always know that there is process variability that cannot be accounted for by a model. Yet, we assume that models do generally line up with reality and that they can help us to describe relationships and effects. I don't have a lot of experience with how researchers in other disciplines use models. I suspect that they don't fully acknowledge how much uncertainty there is around a fitted model, and they probably put too much stock and belief into one fitted model.

He ends with a philosophical statement about probability being a technique for thinking and drawing conclusions about the world around us. Statistical inference is not a black and white method for making decisive conclusions, it's simply a guide to help researchers understand processes in the world. My consulting client this year told me that he doesn't want to use 'evidence' statements to describe conclusions, he wants to use 'significant' or 'not significant' because that is what his advisor prefers. Making decisive statements about whether a null hypothesis is true or false seems to contradict most of what the author said in this paragraph. Again, I'll say that it seems odd that the ideas in this paragraph were developed so long ago, yet inference in many disciplines developed in a way that contradicts these ideas. I think we are in a really interesting age where people are stepping back and revisiting how probability and inference should be used in practice. It is clearly important for statisticians (and maybe philosophers too!) to continue having discussions with researchers in other disciplines about how results of statistical analyses should be interpreted and

what conclusions, if any, can be drawn.

R code appendix

```
set.seed(15)
lambda.draw <- rgamma(1, 20, 1)
x.vec <- rpois(20, lambda.draw)
#sum(x.vec)=409

post.draws10 <- rgamma(10, 429, 21)
#90\% posterior interval
c(quantile(post.draws10, 0.05), quantile(post.draws10, 0.95))
#posterior probability of being between 10 and 20
require(mosaic)
pdata(20, post.draws10)-pdata(10, post.draws10)
#posterior probability of being less than 5
pdata(5, post.draws10)

post.draws50 <- rgamma(50, 429, 21)
#90\% posterior interval
c(quantile(post.draws50, 0.05), quantile(post.draws50, 0.95))
#posterior probability of being between 10 and 20
require(mosaic)
pdata(20, post.draws50)-pdata(10, post.draws50)
#posterior probability of being less than 5
pdata(5, post.draws50)

post.draws100 <- rgamma(100, 429, 21)
#90\% posterior interval
c(quantile(post.draws100, 0.05), quantile(post.draws100, 0.95))
#posterior probability of being between 10 and 20
require(mosaic)
pdata(20, post.draws100)-pdata(10, post.draws100)
#posterior probability of being less than 5
pdata(5, post.draws100)

post.draws250 <- rgamma(250, 429, 21)
#90\% posterior interval
c(quantile(post.draws250, 0.05), quantile(post.draws250, 0.95))
#posterior probability of being between 10 and 20
require(mosaic)
pdata(20, post.draws250)-pdata(10, post.draws250)
#posterior probability of being less than 5
pdata(5, post.draws250)

post.draws1000 <- rgamma(1000, 429, 21)
#90\% posterior interval
c(quantile(post.draws1000, 0.05), quantile(post.draws1000, 0.95))
#posterior probability of being between 10 and 20
require(mosaic)
```

```

pdata(20, post.draws1000)-pdata(10, post.draws1000)
#posterior probability of being less than 5
pdata(5, post.draws1000)

post.draws10000 <- rgamma(10000, 429, 21)
#90% posterior interval
c(quantile(post.draws10000, 0.05), quantile(post.draws10000, 0.95))
#posterior probability of being between 10 and 20
require(mosaic)
pdata(20, post.draws10000)-pdata(10, post.draws10000)
#posterior probability of being less than 5
pdata(5, post.draws10000)

```

```

#function to calculate posterior probabilities
posterior <- function(x){
  dgamma(x, 429, 21)
}

#possible x values
#use small grid size
xsmall <- seq(15, 25, .00001)

##sample x with probability proportional to height of function within that grid
set.seed(17)
grid.draws <- sample(xsmall, 1000, prob=posterior(xsmall))

#plot draws with posterior overlaid
#hist(grid.draws, freq=F, main="grid size 0.00001")
#lines(xsmall, posterior(xsmall))

```

```

set.seed(18)
xmiddle <- seq(15, 25, 0.1)

par(mfrow=c(1,5))

#try a big grid=0.5
xbig <- seq(15, 25, 0.5)
grid.drawsbig <- sample(xbig, size=1000, replace = TRUE,
  prob = posterior(xbig))
hist(grid.drawsbig, freq=F, main="grid size 0.5", xlab="", ylim=c(0, 0.45))
lines(xmiddle, posterior(xmiddle))

#try a middle grid=0.1
xmiddle <- seq(15, 25, 0.1)
grid.drawsmiddle <- sample(xmiddle, 1000, replace=TRUE, prob=posterior(xmiddle))
hist(grid.drawsmiddle, freq=F, main="grid size 0.1", xlab="", ylim=c(0, 0.45))
lines(xmiddle, posterior(xmiddle))

#try another middle grid=0.01
xmids <- seq(15, 25, 0.01)
grid.drawsmid <- sample(xmids, 1000, replace=TRUE, prob=posterior(xmids))

```



```

hist(grid.drawsmid, freq=F, main="grid size 0.01", xlab="", ylim=c(0, 0.45))
lines(xmiddle, posterior(xmiddle))

#try another middle grid=0.001
xmid2 <- seq(15, 25, 0.001)
grid.drawsmid2 <- sample(xmid2, 1000, replace=TRUE, prob=posterior(xmid2))
hist(grid.drawsmid2, freq=F, main="grid size 0.001", xlab="", ylim=c(0, 0.45))
lines(xmiddle, posterior(xmiddle))

#grid size=0.00001
hist(grid.draws, freq=F, main="grid size 0.00001", xlab="", ylim=c(0, 0.45))
lines(xmiddle, posterior(xmiddle))

#compare to rgamma
#hist(rgamma(1000, 429, 21), freq=F, xlab="", ylim=c(0, 0.45))
#lines(xmiddle, posterior(xmiddle))

```

```

post.draws1000 <- rgamma(1000, 429, 21)
#90% posterior interval
c(quantile(post.draws1000, 0.05), quantile(post.draws1000, 0.95))
#posterior probability of being between 10 and 20
require(mosaic)
pdata(20, post.draws1000)-pdata(10, post.draws1000)
#posterior probability of being less than 5
pdata(5, post.draws1000)

#grid.draws=0.00001
#90% posterior interval
c(quantile(grid.draws, 0.05), quantile(grid.draws, 0.95))
#posterior probability of being between 10 and 20
pdata(20, grid.draws)-pdata(10, grid.draws)
#posterior probability of being less than 5
pdata(5, grid.draws)

#mid=0.001
#90% posterior interval
c(quantile(grid.drawsmid2, 0.05), quantile(grid.drawsmid2, 0.95))
#posterior probability of being between 10 and 20
pdata(20, grid.drawsmid2)-pdata(10, grid.drawsmid2)
#posterior probability of being less than 5
pdata(5, grid.drawsmid2)

#middle=0.1
#90% posterior interval
c(quantile(grid.drawsmiddle, 0.05), quantile(grid.drawsmiddle, 0.95))
#posterior probability of being between 10 and 20
pdata(20, grid.drawsmiddle)-pdata(10, grid.drawsmiddle)
#posterior probability of being less than 5
pdata(5, grid.drawsmiddle)

#mid=0.01
#90% posterior interval

```

```

c(quantile(grid.drawsmid, 0.05), quantile(grid.drawsmid, 0.95))
#posterior probability of being between 10 and 20
pdata(20, grid.drawsmid)-pdata(10, grid.drawsmid)
#posterior probability of being less than 5
pdata(5, grid.drawsmid)

#big=0.5
#90\% posterior interval
c(quantile(grid.drawsbig, 0.05), quantile(grid.drawsbig, 0.95))
#posterior probability of being between 10 and 20
pdata(20, grid.drawsbig)-pdata(10, grid.drawsbig)
#posterior probability of being less than 5
pdata(5, grid.drawsbig)

```

```

a <- c(rep("0.5",1000), rep("0.1",1000), rep("0.01", 1000), rep("0.001", 1000), rep("0.00001", 1000), rep("rga

test <- c(as.numeric(grid.drawsbig), grid.drawsmiddle, grid.drawsmid, grid.drawsmid2, grid.draws, post.draws100)
beansetup <- cbind.data.frame(test, a)
require(beanplot, quietly=T)
beanplot(test~a, data=beansetup, log="", col=7, what = c(1,1,1,0))

```

```

post.draws100 <- rgamma(100, 429, 21)
#90\% posterior interval
c(quantile(post.draws100, 0.05), quantile(post.draws100, 0.95))
#posterior probability of being between 10 and 20
pdata(20, post.draws100)-pdata(10, post.draws100)
#posterior probability of being less than 5
pdata(5, post.draws100)

set.seed(15)
set.seed(16)
set.seed(17)
set.seed(18)
set.seed(19)
set.seed(20)

mean(post.draws100)
sd(post.draws100)

```

```

par(mfrow=c(1,3))

hist(post.draws10^2/(1-post.draws10), main=expression(paste("n=10, ", lambda^2/(1-lambda))))
#hist(post.draws50^2/(1-post.draws50), main=expression(paste("n=50, ", lambda^2/(1-lambda))))
#hist(post.draws100^2/(1-post.draws100), main=expression(paste("n=100, ", lambda^2/(1-lambda))))
hist(post.draws250^2/(1-post.draws250), main=expression(paste("n=250, ", lambda^2/(1-lambda))))
#hist(post.draws1000^2/(1-post.draws1000), main=expression(paste("n=1000, ", lambda^2/(1-lambda))))
hist(post.draws10000^2/(1-post.draws10000), main=expression(paste("n=10000, ", lambda^2/(1-lambda))))

hist(log(post.draws10), main=expression(paste("n=10, ", log(lambda))))

```

```
#hist(log(post.draws50), main=expression(paste("n=50, ", log(lambda))))
#hist(log(post.draws100), main=expression(paste("n=100, ", log(lambda))))
hist(log(post.draws250), main=expression(paste("n=250, ", log(lambda))))
#hist(log(post.draws1000), main=expression(paste("n=1000, ", log(lambda))))
hist(log(post.draws10000), main=expression(paste("n=10000, ", log(lambda))))
```

```
set.seed(21)
postpred <- rnbino(10000, 429, 21/22)
require(ggplot2)
barplot(table(factor(postpred, levels = 6:38))/10000, main="Approx Posterior Predictive Distribution", densit
```

```
x <- seq(6, 38, by=1)
y <- dnbinom(x, 429, 21/22)
barplot(table(factor(postpred, levels = 6:38))/10000, main="Approx Posterior Predictive Distribution with Over
barplot(y, col="red", density=20,add=TRUE)
legend(27, 0.07, c("10000 draws", "truth"), fill=c("lightblue", "red"), density=c(0, 20))
```

```
#remember  $p = \text{beta}/(\text{beta}+1)$ 
set.seed(21)
priorpred <- rnbino(10000, 20, 0.5)
par(mfrow=c(1,2))
barplot(table(priorpred), xlab="", main="Prior Predictive")
barplot(table(postpred), xlab="", main="Posterior Predictive")
```

```
postmatrix <- matrix(NA, nrow=1000, ncol=20)

for(i in 1:1000){
  postmatrix[i,] <- rnbino(20, 429, 21/22)
}

par(mfrow=c(3,3))
par(oma=c(1,1,1,5))
for(i in 1:9){
  stripchart(postmatrix[i,], method="stack", at=0.15, offset=0.9, xlab="", main=paste("Draw", i), xlim=c(9, 40)
}
```

```
sds <- apply(postmatrix, 1, sd)
hist(sds, main="Approx Posterior Pred Dist for the Sample SD", nclass=50)
set.seed(15)
lambda.draw <- rgamma(1, 20, 1)
x.vec <- rpois(20, lambda.draw)
abline(v=sd(x.vec), lwd=3, col="red")
```

```

set.seed(39)
prematrix <- matrix(NA, nrow=10, ncol=20)

for(i in 1:10){
  prematrix[i,] <- rpois(20, 20.66)
}
sums <- apply(prematrix, 1, sum)
obssds <- apply(prematrix, 1, sd)
obsmax <- apply(prematrix, 1, max)

postmatrix388 <- matrix(NA, nrow=1000, ncol=20)
postmatrix428 <- matrix(NA, nrow=1000, ncol=20)
postmatrix439 <- matrix(NA, nrow=1000, ncol=20)
postmatrix396 <- matrix(NA, nrow=1000, ncol=20)

for(i in 1:1000){
  postmatrix388[i,] <- rbinom(20, 408, 21/22)
}

for(i in 1:1000){
  postmatrix428[i,] <- rbinom(20, 448, 21/22)
}

for(i in 1:1000){
  postmatrix439[i,] <- rbinom(20, 459, 21/22)
}

for(i in 1:1000){
  postmatrix396[i,] <- rbinom(20, 416, 21/22)
}

par(mfrow=c(2,2))
sds388 <- apply(postmatrix388, 1, sd)
hist(sds388, main="Approx Post Pred Dist for Sample SD", nclass=50)
abline(v=obssds[1], lwd=3, col="red")

sds428 <- apply(postmatrix428, 1, sd)
hist(sds428, main="Approx Post Pred Dist for Sample SD", nclass=50)
abline(v=obssds[2], lwd=3, col="red")

sds439 <- apply(postmatrix439, 1, sd)
hist(sds439, main="Approx Post Pred Dist for Sample SD", nclass=50)
abline(v=obssds[3], lwd=3, col="red")

sds396 <- apply(postmatrix396, 1, sd)
hist(sds396, main="Approx Post Pred Dist for Sample SD", nclass=50)
abline(v=obssds[4], lwd=3, col="red")

```

```

max <- apply(prematrix, 1, max)
stripchart(max, method="stack", at=0.15, offset=0.1, xlab="", main="Approx Post Pred Dist for Sample Max", cex=
abline(v=max(x.vec), lwd=3, col="red")

```

```

par(mfrow=c(2,2))
max388 <- apply(postmatrix388, 1, max)
stripchart(max388, method="stack", at=0.15, offset=0.03, xlab="", main="Approx Post Pred Dist for Sample Max",
abline(v=obsmax[1], lwd=3, col="red")

max428 <- apply(postmatrix428, 1, max)
stripchart(max428, method="stack", at=0.15, offset=0.03, xlab="", main="Approx Post Pred Dist for Sample Max",
abline(v=obsmax[2], lwd=3, col="red")

max439 <- apply(postmatrix439, 1, max)
stripchart(max439, method="stack", at=0.15, offset=0.03, xlab="", main="Approx Post Pred Dist for Sample Max",
abline(v=obsmax[3], lwd=3, col="red")

max396 <- apply(postmatrix396, 1, max)
stripchart(max396, method="stack", at=0.15, offset=0.03, xlab="", main="Approx Post Pred Dist for Sample Max",
abline(v=obsmax[4], lwd=3, col="red")

```