Homework 1

Gaurav Kandlikar & Camila Madeiros

October 8, 2016

Question 1

1. Explain what your measurements will be.
   We will be exploring Gaurav's running speeds from his last ten runs.

Question 2

2. Deciding on some priors
   We estimate a mean speed (μ_0) of 8 minutes and 10 seconds (i.e. 490 seconds) per mile. The basis for this is that on good days, I (Gaurav) generally try to run 8-minute (480 seconds) miles, and I know that over the past ~2 weeks I've been running slower due to a weird knee. We estimate a standard deviation (τ) of 30 seconds because I've done both road runs and treadmill runs, and I know that I run somewhat differently in the two conditions.

Question 3

3. Report the data and sample mean and variance (n-1) denominator.

```
runs <- read.csv("running_data.csv")
head(runs)
```

```
##   sample distance time_min time_sec speed_sec_per_mi
## 1      1      3.5       27     1620          462.857
## 2      2      3.1       24     1440          464.516
## 3      3      7.1       58     3480          490.141
## 4      4      4.0       35     2100          525.000
## 5      5      5.4       48     2880          533.333
## 6      6     13.5      130     7800          577.778
```

```
sample_mean <- mean(runs$speed_sec_per_mi)
sample_mean # get the sample mean
```

```
## [1] 513.719
```

```
# Calculate sample variance
sample_var <- sum((runs$speed_sec_per_mi-sample_mean)^2)/(nrow(runs)-1)
```

The mean of our sample is 513.719132; the variance of our sample is 2133.81829.

Question 4

4. Now specify the sampling standard deviation. Since we are doing a one parameter model, and since this value is usually not known, we need to do something because we are working with such a simple model.

We know that the speed estimates from the run tracking app are fairly accurate: the speeds from the app have closely matched my race speeds recorded independently. We don't think that the sampling standard deviation is higher than the sample σ of 46.193271, so we will proceed in the analysis assuming the sampling standard deviation is the same as the sd of the data .

Question 5

5. Calculate the posterior mean, variance, and SD.

Because we are working with a normal-normal model (mean is normally distributed with, in this case, a known sampling error), we can use the following formulae to collect posterior mean and variance:

$$\bar{\theta} = \frac{\frac{n}{\sigma^2}}{\frac{n}{\sigma^2}+\frac{1}{\tau^2}}\bar{y} + \frac{\frac{1}{\tau^2}}{\frac{n}{\sigma^2}+\frac{1}{\tau^2}}\theta_0$$

$$\text{var}(\theta \mid \text{data}) = \left(\frac{n}{\sigma^2}+\frac{1}{\tau^2}\right)^{-1}$$

```
prior_mean = 490 #seconds
prior_sd   = 30 #seconds
sampling_sd = sqrt(sample_var)
n = nrow(runs)
ybar = sample_mean

# calculate the posterior mean using the formula
posterior_mean <- (n/(sampling_sd^2))/((n/sampling_sd^2)+(1/prior_sd^2)) * ybar +
  (1/prior_sd^2)/((n/sampling_sd^2)+(1/prior_sd^2)) * prior_mean

# calculate the posterior variance using the formula
posterior_var <- ((n/(sampling_sd^2))+(1/(prior_sd^2)))^-1
posterior_sd <- sqrt(posterior_var)
```

The posterior mean is $\bar{\mu} = 509.173313$;
The posterior var is $V = 172.48678$;
The posterior sd is $sd = 13.133422$.

Question 6

6. The prior predictive density is the density that you predict for a single observation before seeing any data.
   It sure is!

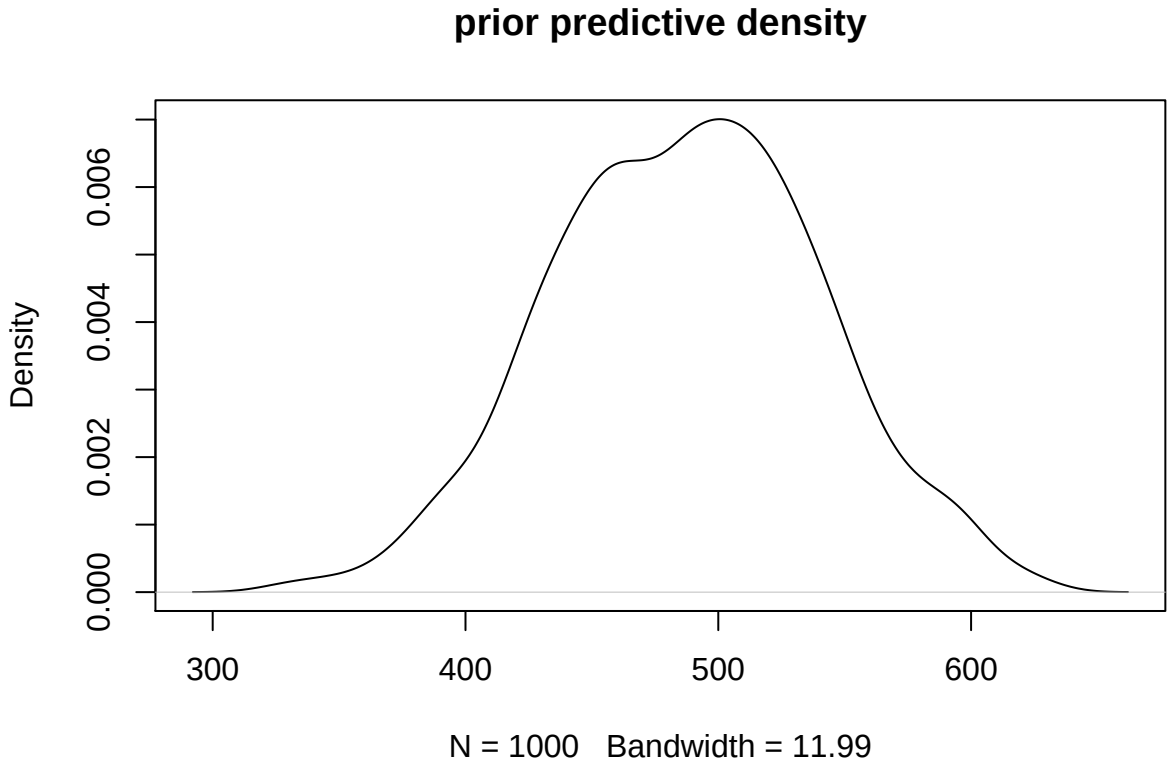
```
prior_mean; prior_sd; sampling_sd
```

```
## [1] 490
```

```
## [1] 30
```

```
## [1] 46.1933
```

```
# Sample from a normal distribution with the parameters above
plot(density(rnorm(1000, prior_mean, sqrt(prior_sd^2+sampling_sd^2))),
     main = "prior predictive density")
```



Question 7

7. Construct a table with means, sds and vars for the (i) posterior for mu, (ii) the prior for mu, (iii) the prior predictive for y, and (iv) the likelihood of mu.

```
posterior_row  <- c(posterior_mean, posterior_sd, posterior_var)
prior_row      <- c(prior_mean, prior_sd, prior_sd^2)
prior_pred_row <- c(prior_mean, sqrt(prior_sd^2+sampling_sd^2),
                    prior_sd^2+sampling_sd^2)
likelihood_row <- c(sample_mean, sqrt(sample_var/n), sample_var/n)

to_print <- rbind(posterior_row, prior_row, prior_pred_row, likelihood_row)
rownames(to_print) <- c("Posterior", "Prior", "Prior Predictive", "Likelihood")
colnames(to_print) <- c("Mean", "SD", "Variance")
knitr::kable(to_print, caption = "Posterior, Prior, Prior Predictive, and Likelihood calculated from th
```

Table 1: Posterior, Prior, Prior Predictive, and Likelihood calculated from their definitions

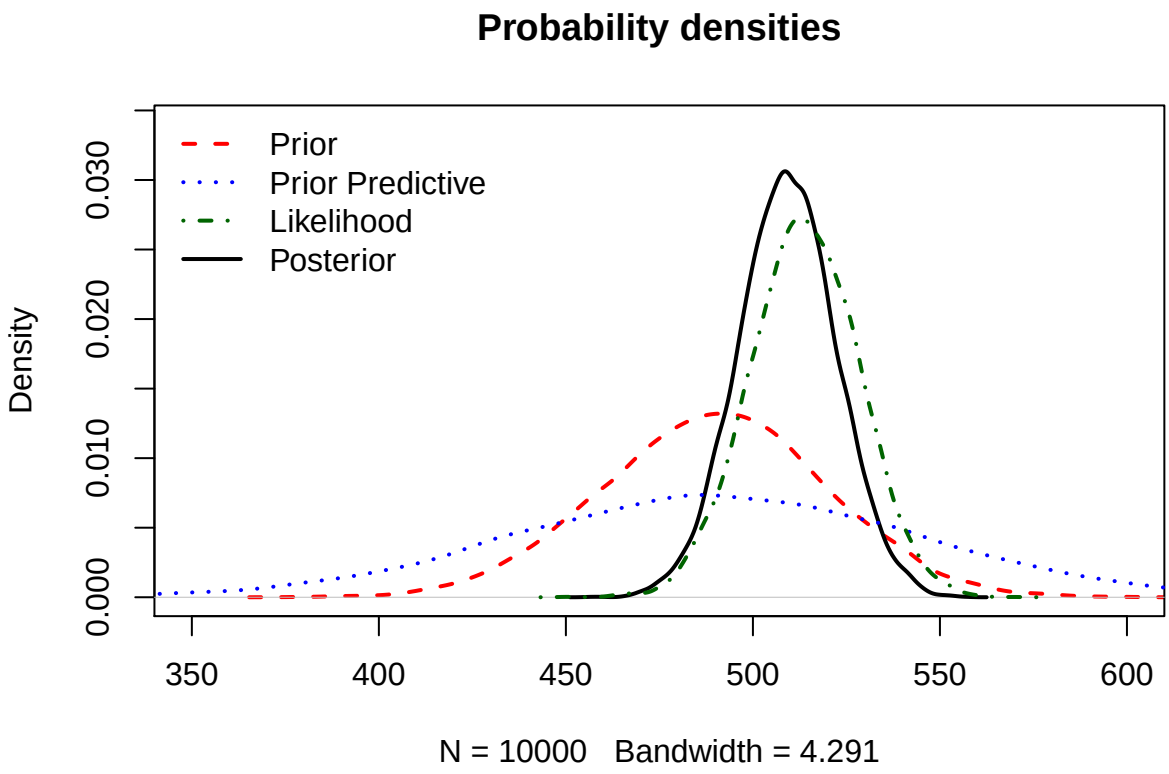
	Mean	SD	Variance
Posterior	509.173	13.1334	172.487
Prior	490.000	30.0000	900.000
Prior Predictive	490.000	55.0801	3033.818
Likelihood	513.719	14.6076	213.382

Question 8

8. Plot on a single plot the (i) posterior for mu, (ii) the prior for mu, (iii) the prior predictive for y, and (iv) the likelihood of mu (suitably normalized so it looks like a density, ie a normal with mean y-bar and variance sigma2 /n) all on the same graph. Interpret the plot.

```
posterior_vec <- rnorm(10000, posterior_mean, posterior_sd)
prior_vec <- rnorm(10000, prior_mean, prior_sd)
prior_pred_vec <- rnorm(10000, prior_mean, sqrt(prior_sd^2+sampling_sd^2))
likelihood_vec <- rnorm(10000, sample_mean, sqrt(sample_var/n))

plot(density(prior_vec), lty = 2, col = "red", main = "Probability densities",
     ylim = c(0, 0.034), lwd = 2, xlim = c(350, 600))
lines(density(posterior_vec), lwd = 2)
lines(density(prior_pred_vec), col = "blue", lty = 3, lwd = 2)
lines(density(likelihood_vec), col = "darkgreen", lty = 4, lwd = 2)
legend("topleft", lty = c(2, 3, 4, 1), col = c("red", "blue", "darkgreen", "black"), lwd = 2,
       legend = c("Prior", "Prior Predictive", "Likelihood", "Posterior"), bty = "n")
```



Interpretation of plot: The posterior distribution of mean running speed is in between the prior mean and the likelihood estimate; in other words, the posterior is a compromise between the likelihood and the prior, which shrank the likelihood. The prior predictive has a mean equal to the prior mean but has a much wider distribution, which comes from our uncertainty in measurements as well as the prior estimate for variation in running speed. Our posterior distribution is quite similar to the likelihood distribution, which happens because the weight given to our prior mean is lower than the weight given to the sample mean in the posterior calculation. Practically, this means that I (Gaurav) have been running much slower than I thought over the past three weeks or so- my average speed is about twenty seconds a mile slower than I thought. Time to fix that!

Question 9

9. Write R/WinBUGS programs to sample from the posterior of mu.

```
#sink("running_model.txt")
cat("model {
    for (i in 1:N) {
        x[i] ~ dnorm(mu, tau)
    }
    mu ~ dnorm(prior_mean, prior_tau) # change this to dnorm(prior_mean and prior_tau)
    sigma <- sampling_sd              # change this to be sampling_sd
    tau <- 1/(sigma^2)             # tau equal to 1/sigma^2
}", fill = TRUE)
```

```
## model {
```

```
##  for (i in 1:N) {
##      x[i] ~ dnorm(mu, tau)
##  }
##  mu ~ dnorm(prior_mean, prior_tau) # change this to dnorm(prior_mean and prior_tau)
##  sigma <- sampling_sd              # change this to be sampling_sd
##  tau <- 1/(sigma^2)             # tau equal to 1/sigma^2
## }
```

```
#sink()
```

```
# parameters
jags.params = c("mu", "sigma", "tau")

# data
x = runs$speed_sec_per_mi
N = length(runs$speed_sec_per_mi)
prior_tau = 1/(prior_sd^2)
jags.data = list("x", "N", "prior_mean", "sampling_sd", "prior_tau")

# initials
jags.inits = function(){
    list("mu" = 0) # part of algorithm!
}
```

Now that we have set up the model, data, and parameters, we can run the model:

```
#sampling from the posterior
hw1.sim = jags(jags.data, jags.inits, jags.params,
               model.file = "running_model.txt",
               n.chains = 3, n.iter = 11000, n.burnin = 1000)
```

```
## module glm loaded

## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 10
##    Unobserved stochastic nodes: 1
##    Total graph size: 21
##
## Initializing model
```

We can summarize the output of the model:

```
to_print <- hw1.sim$BUGSoutput$summary[2:4,c("mean", "sd", "2.5%", "97.5%")]
knitr::kable(to_print, caption = "Summary of posterior from JAGS run")
```

Table 2: Summary of posterior from JAGS run

	mean	sd	2.5%	97.5%
mu	509.486074	13.4088	483.271988	536.283907
sigma	46.193271	0.0000	46.193271	46.193271
tau	0.000469	0.0000	0.000469	0.000469

Question 10

10. Adapt your BUGS program to sample from the prior and prior predictive. Do this by not loading your data, rather, in loading the initial values, move the data y over to the init list instead. There is an example at the end of Homework 2 for a Poisson-gamma likelihood/prior. [Helpful step: set keyword DIC=F in the call to bugs, as WinBUGS can not calculate DIC for prior predictions.]

```
#Sampling from the prior:
# parameters
jags.params = c("mu", "sigma", "tau")

# data
x = runs$speed_sec_per_mi
N = length(runs$speed_sec_per_mi)
prior_tau = 1/(prior_sd)^2
jags.data.prior = list("N", "prior_mean", "sampling_sd", "prior_tau")

# initials
jags.inits_prior = function(){
    list("mu"=0, "x"=x) # part of algorithm!
}

hw1.sim2 = jags(jags.data.prior, jags.inits_prior, jags.params,
              model.file = "running_model.txt",
              n.chains = 3, n.iter = 11000, n.burnin = 1000, DIC=F)
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 0
##    Unobserved stochastic nodes: 11
##    Total graph size: 21
##
## Initializing model
```

```
to_print_prior <- hw1.sim2$BUGSoutput$summary[,c("mean", "sd", "2.5%", "97.5%")]
knitr::kable(to_print_prior, caption = "Summary of prior from JAGS run")
```

Table 3: Summary of prior from JAGS run

	mean	sd	2.5%	97.5%
mu	490.217382	30.3367	429.768699	548.396511
sigma	46.193271	0.0000	46.193271	46.193271
tau	0.000469	0.0000	0.000469	0.000469

```
#Sampling from prior predictive
# parameters
jags.params = c("mu","tau", "sigma")

# data
x = runs$speed_sec_per_mi
N = length(runs$speed_sec_per_mi)
prior_tau = 1/((prior_sd^2)+(sampling_sd^2))
jags.data_prior_predictive = list("N", "prior_mean", "sampling_sd", "prior_tau")

# initials
jags.inits_prior_predictive = function(){
    list("mu"= 0, "x"=x) # part of algorithm!
}

hw1.sim3 = jags(jags.data_prior_predictive, jags.inits_prior_predictive, jags.params,
                model.file = "running_model.txt",
                n.chains = 3, n.iter = 11000, n.burnin = 1000,
                DIC=F)
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 0
##    Unobserved stochastic nodes: 11
##    Total graph size: 21
##
## Initializing model
```

```
to_print_prior_predictive <- hw1.sim3$BUGSoutput$summary[,c("mean", "sd", "2.5%", "97.5%")]
knitr::kable(to_print_prior_predictive,  caption = "Summary of prior predictive from JAGS run")
```

Table 4: Summary of prior predictive from JAGS run

	mean	sd	2.5%	97.5%
mu	491.074349	55.1473	384.782157	599.904515
sigma	46.193271	0.0000	46.193271	46.193271
tau	0.000469	0.0000	0.000469	0.000469

Question 11

11. Adapt your BUGS program to sample from the likelihood.

```
###Sampling from likelihood
# parameters
jags.params = c("mu","tau", "sigma")

# data
x = runs$speed_sec_per_mi
N = length(runs$speed_sec_per_mi)
prior_tau = 1/(((prior_sd^2)/N))^2
jags.data_likelihood = list("x", "N", "prior_mean", "sampling_sd", "prior_tau")

# initials
jags.inits = function(){
    list("mu"= 0) # part of algorithm!
}

hw1.sim4 = jags(jags.data_likelihood, jags.inits, jags.params,
              model.file = "running_model.txt",
              n.chains = 3, n.iter = 11000, n.burnin = 1000)
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 10
##    Unobserved stochastic nodes: 1
##    Total graph size: 21
##
## Initializing model
```

```
to_print_likelihood <- hw1.sim4$BUGSoutput$summary[,c("mean", "sd", "2.5%", "97.5%")]
knitr::kable(to_print_likelihood, caption = "Summary of likelihood from JAGS run")
```

Table 5: Summary of likelihood from JAGS run

	mean	sd	2.5%	97.5%
deviance	104.978358	1.31638	104.036275	108.588478
mu	513.163286	14.17592	485.077380	540.717136
sigma	46.193271	0.00000	46.193271	46.193271
tau	0.000469	0.00000	0.000469	0.000469

Question 12

12. Report your WinBUGS models and R code, data, and inits. Use at least samples of size 10000. R code, model, data, and inits are reported in questions 9-11.

Question 13

13. Construct a table with means, sds and vars for the (i) posterior for mu,(ii) the prior for mu, (iii) the prior predictive for y, and (iv) the likelihood of mu from the WinBUGS output.

```
Posterior_mu   <- hw1.sim$BUGSoutput$summary[2, c("mean", "sd")]
Prior_mu       <- hw1.sim2$BUGSoutput$summary[1,c("mean", "sd")]
Prior_predictive_y <- hw1.sim3$BUGSoutput$summary[1,c("mean", "sd")]
Likelihood_mu <- hw1.sim4$BUGSoutput$summary[2,c("mean", "sd")]

to_print <- rbind(Posterior_mu, Prior_mu, Prior_predictive_y, Likelihood_mu)
print(to_print)
```

```
##                        mean      sd
## Posterior_mu       509.486 13.4088
## Prior_mu           490.217 30.3367
## Prior_predictive_y 491.074 55.1473
## Likelihood_mu      513.163 14.1759
```

```
var <- as.matrix((to_print[,2])^2)
results <- cbind(to_print, var)
rownames(results) <- c("Posterior", "Prior", "Prior predictive", "Likelihood")
colnames(results) <- c("Mean", "SD", "Variance")
knitr::kable(results, caption = "Posterior, Prior, Prior Predictive, and Likelihood from JAGS simulatio
```

Table 6: Posterior, Prior, Prior Predictive, and Likelihood from JAGS simulations

	Mean	SD	Variance
Posterior	509.486	13.4088	179.797
Prior	490.217	30.3367	920.315
Prior predictive	491.074	55.1473	3041.222
Likelihood	513.163	14.1759	200.957

Question 14

14. Plot on a single plot the (i) posterior for mu, (ii) the prior for mu (iii) the prior predictive for y, and (iv) the likelihood of mu (suitably normalized so it looks like a density, ie a normal with mean y-bar and variance sigma2/n) all on the same graph. All from the WinBUGS output. Interpret the plot.

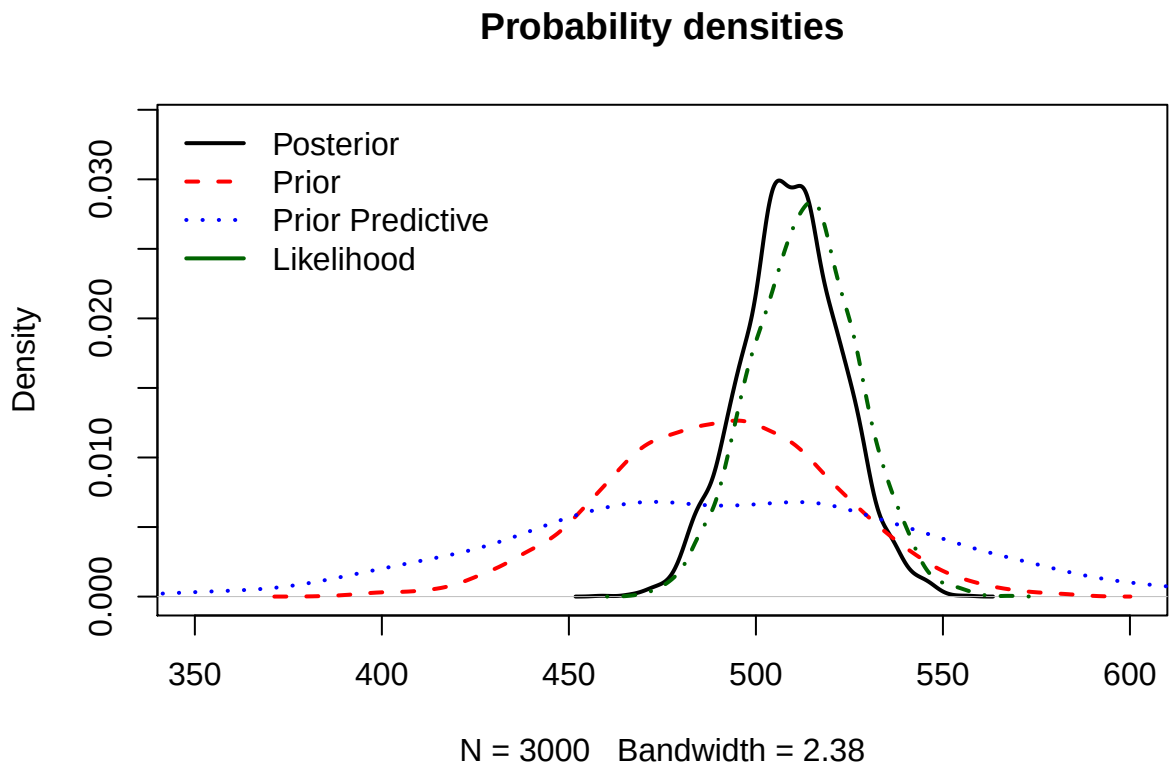
```
# Extract the posterior from the BUGS run
posterior = apply(hw1.sim$BUGSoutput$sims.array, 3, unlist)
posterior <- posterior[,2]
# Extract the prior from the BUGS run
prior = apply(hw1.sim2$BUGSoutput$sims.array, 3, unlist)
prior <- prior[,1]

# Extract the prior predictive from the BUGS run
```

```r
prior_predictive = apply(hw1.sim3$BUGSoutput$sims.array, 3, unlist)
prior_predictive <- prior_predictive[,1]

# Extract the likelihood from the BUGS run
likelihood = apply(hw1.sim4$BUGSoutput$sims.array, 3, unlist)
likelihood <- likelihood[,2]

plot(density(posterior), main = "Probability densities", xlim= c(350, 600), ylim= c(0, 0.034), lwd = 2)
lines(density(prior), lty = 2, lwd = 2, col = "red")
lines(density(prior_predictive),lty = 3, lwd = 2, col = "blue")
lines(density(likelihood),lty = 4, lwd = 2, col = "darkgreen")
legend("topleft", lty = c(1, 2, 3), col = c("black", "red", "blue", "darkgreen"), lwd = 2,
       legend = c("Posterior","Prior", "Prior Predictive", "Likelihood"), bty = "n")
```



Interpretation of the plot: As expected this plot is identical (more or less) to the plot generated from the same distributions in Question 8. As before, we see that the posterior distribution represents something of a compromise between the prior and likelihood distributions. Practically, this means that I (Gaurav) have been running much slower than I thought over the past three weeks or so- my average speed is about twenty seconds a mile slower than I thought. Time to fix that!