

Ec/ACM/CS 112 - PS4

Ben Juarez

preliminaries

```
data = read.csv("~/Desktop/MetabolicRate.csv")
logMrate = log(data$Mrate)
logBodySize = log(data$BodySize)
Instar = data$Instar

stepSize = function(grid) {
  if (length(grid)==1) {
    step = 1
  }
  else {
    step = (max(grid) - min(grid)) / (length(grid) - 1)
  }
  return(step)
}

buildPriorMultivar = function(beta0Grid,beta1Grid,beta2Grid,sigmaGrid) {
  nBeta0Grid = length(beta0Grid)
  nBeta1Grid = length(beta1Grid)
  nBeta2Grid = length(beta2Grid)
  nSigmaGrid = length(sigmaGrid)
  prior = array( rep(1, nBeta0Grid * nBeta1Grid * nBeta2Grid * nSigmaGrid ),
    dim = c(nBeta0Grid, nBeta1Grid, nBeta2Grid, nSigmaGrid ))
  for (nB0 in 1:nBeta0Grid) {
    for (nB1 in 1:nBeta1Grid) {
      for (nB2 in 1:nBeta2Grid) {
        for (nSig in 1:nSigmaGrid) {
          prior[nB0,nB1,nB2, nSig] = 1 / nSig^2
        }
      }
    }
  }
  return(prior)
}

likelihoodMultivar = function(y,x1, x2, b0L, b1L, b2L, sL){
  loglike = sum(log(dnorm(y-b0L-b1L*x1-b2L*x2, mean = 0, sd=sL)))
  like = exp(loglike)
  return(like)
}

compPostMultivar = function(y,x1, x2, prior, beta0Grid,beta1Grid,beta2Grid,sigmaGrid) {
```

```

nBeta0Grid = length(beta0Grid)
nBeta1Grid = length(beta1Grid)
nBeta2Grid = length(beta2Grid)
nSigmaGrid = length(sigmaGrid)
post = array(rep(-1, nBeta0Grid * nBeta1Grid * nBeta2Grid * nSigmaGrid ),
             dim = c(nBeta0Grid, nBeta1Grid, nBeta2Grid, nSigmaGrid))
for (nBeta0 in 1:nBeta0Grid) {
  b0 = beta0Grid[nBeta0]
  for (nBeta1 in 1:nBeta1Grid) {
    b1 = beta1Grid[nBeta1]
    for (nBeta2 in 1:nBeta2Grid) {
      b2 = beta2Grid[nBeta2]
      for (nSigma in 1:nSigmaGrid) {
        s = sigmaGrid[nSigma]
        post[nBeta0,nBeta1,nBeta2,nSigma] =
          likelihoodMultivar(y,x1,x2,b0,b1,b2,s) *
          prior[nBeta0,nBeta1,nBeta2,nSigma]
      }
    }
  }
}
post = post / (sum(post) * stepSize(beta0Grid) * stepSize(beta1Grid) *
              stepSize(beta2Grid) * stepSize(sigmaGrid))
return(post)
}

```

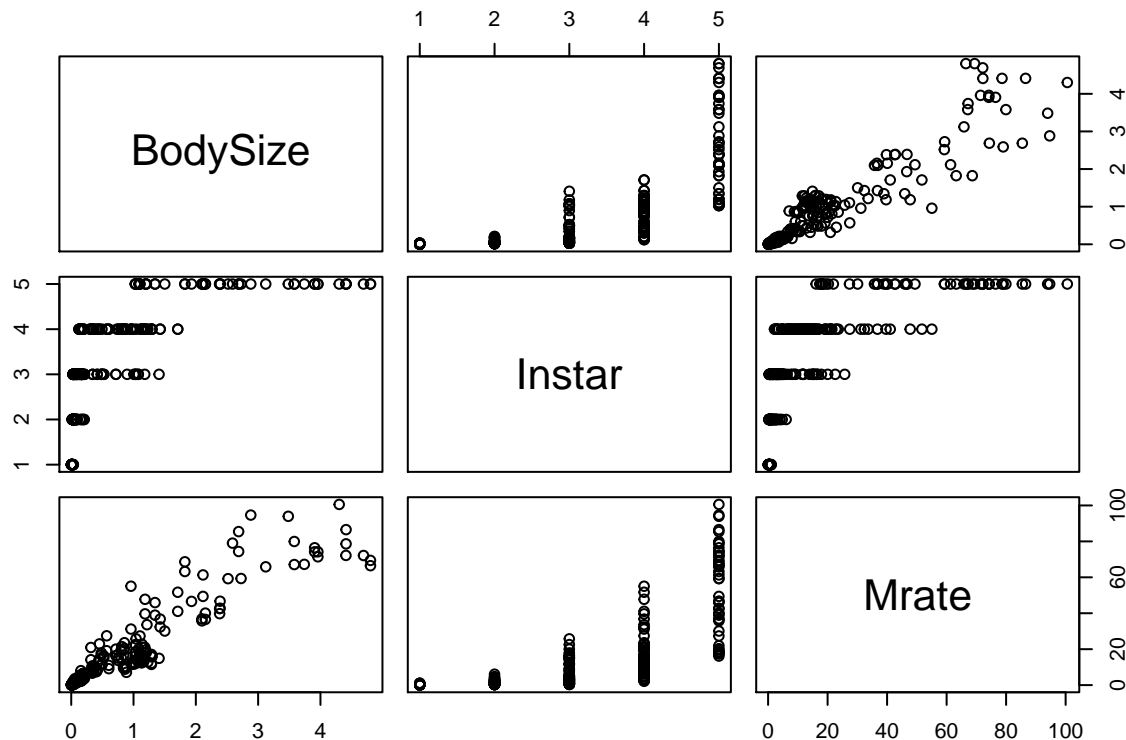
Question 1

Step 1

```

pairs(data)

```



This scatterplot matrix gives us insight into the correlations between our variables. Using `pairs()` it appears that there is positive correlation between each of the variables `BodySize`, `Instar`, and `Mrate`. In our case, we are constructing a model using `BodySize` and `Instar` as our independent variables. However, this information raises potential concern because `BodySize` and `Instar` are seemingly correlated, which goes against any independence assumptions.

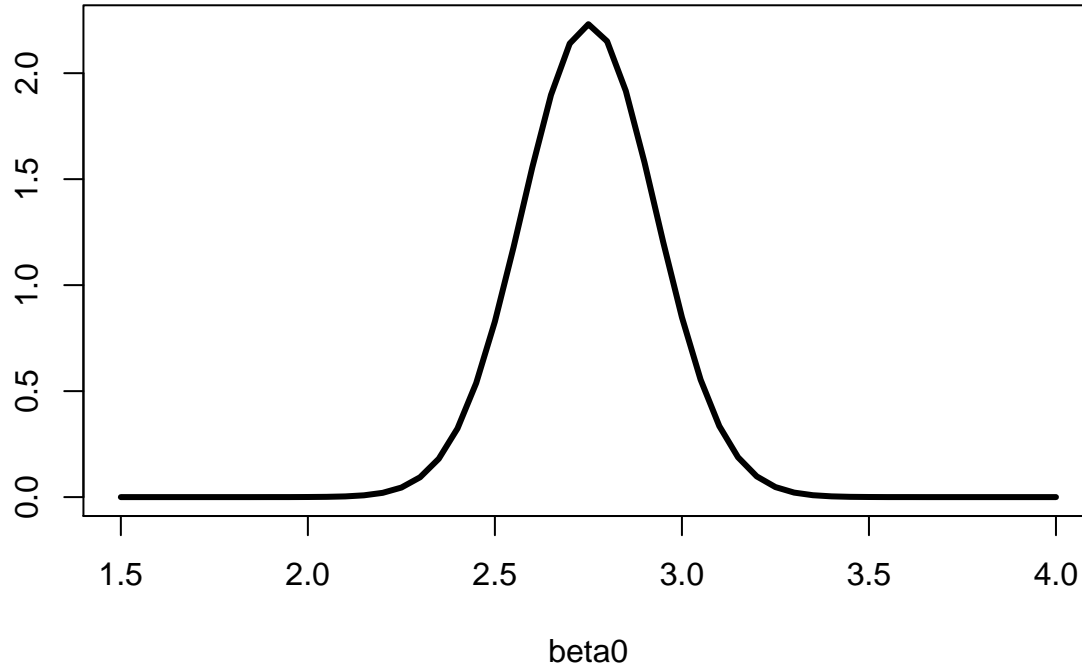
Step 2

```
stepBeta0Grid = 0.05
stepBeta1Grid = 0.01
stepBeta2Grid = 0.01
stepSigmaGrid = 0.01
beta0Grid = seq(1.5, 4, by = stepBeta0Grid)
beta1Grid = seq(0.6, 1.2, by = stepBeta1Grid)
beta2Grid = seq(-0.2, 0.4, by = stepBeta2Grid)
sigmaGrid = seq(0.25, 0.55, by = stepSigmaGrid)

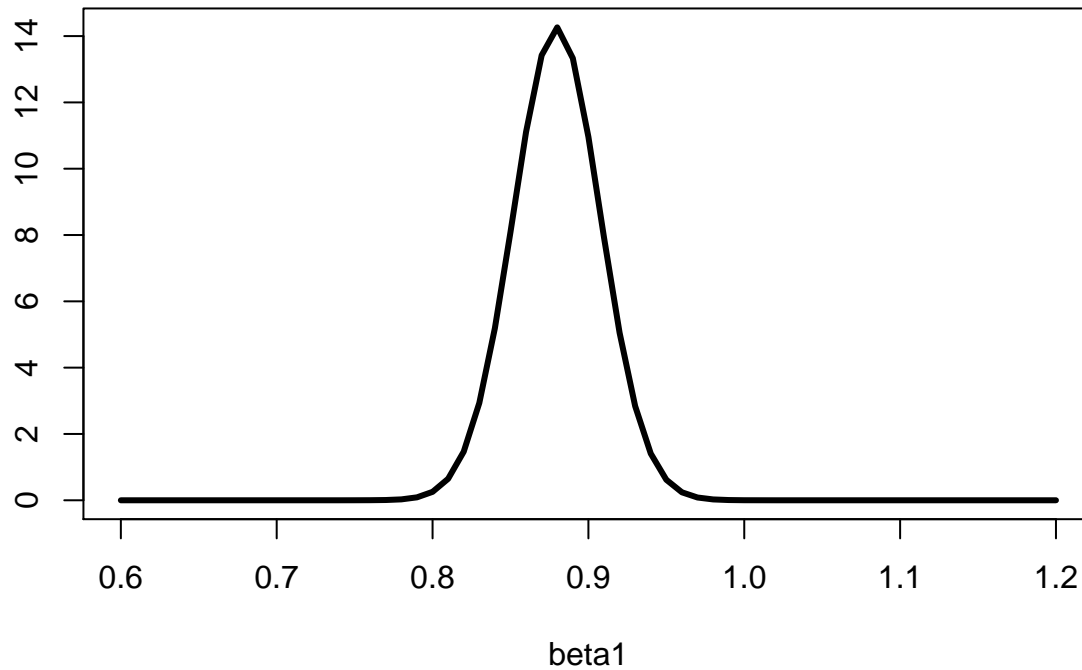
priorM3 = buildPriorMultivar(beta0Grid, beta1Grid, beta2Grid, sigmaGrid)
postM3 = compPostMultivar(logMrate, logBodySize, Instar, priorM3, beta0Grid,
                          beta1Grid, beta2Grid, sigmaGrid)

margPostBeta0M3 = apply(postM3, c(1), sum)
margPostBeta0M3 = margPostBeta0M3 / (sum(margPostBeta0M3) * stepSize(beta0Grid))
margPostBeta1M3 = apply(postM3, c(2), sum)
margPostBeta1M3 = margPostBeta1M3 / (sum(margPostBeta1M3) * stepSize(beta1Grid))
margPostBeta2M3 = apply(postM3, c(3), sum)
margPostBeta2M3 = margPostBeta2M3 / (sum(margPostBeta2M3) * stepSize(beta2Grid))
margPostSigmaM3 = apply(postM3, c(4), sum)
margPostSigmaM3 = margPostSigmaM3 / (sum(margPostSigmaM3) * stepSize(sigmaGrid))
```

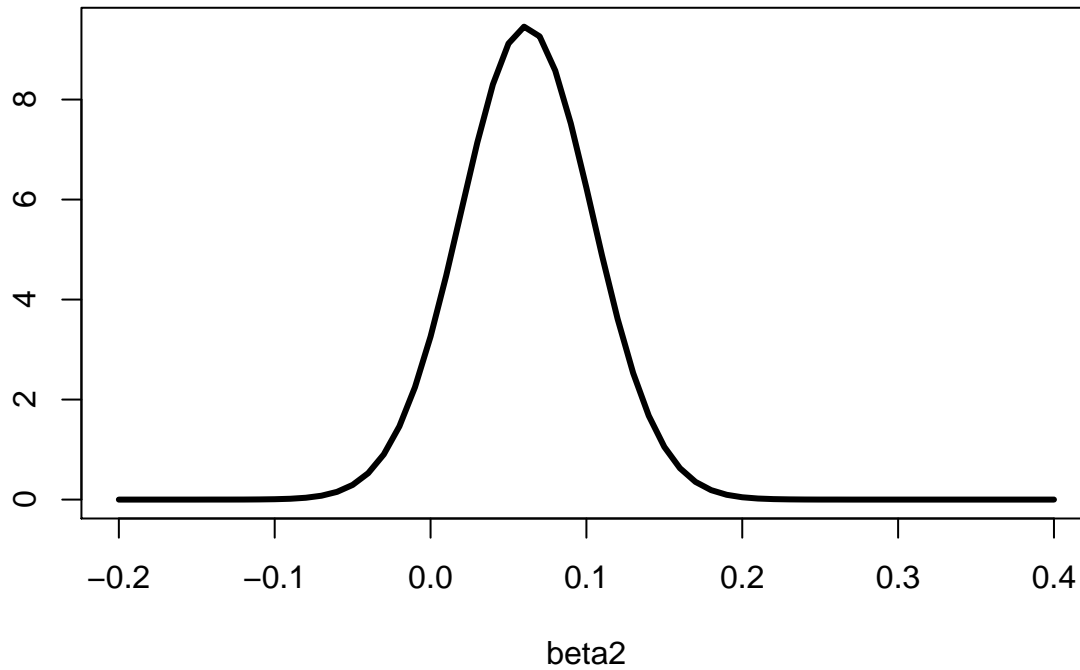
```
plot(beta0Grid, margPostBeta0M3, xlab = "beta0", ylab="", type = "l", lwd = 3)
```



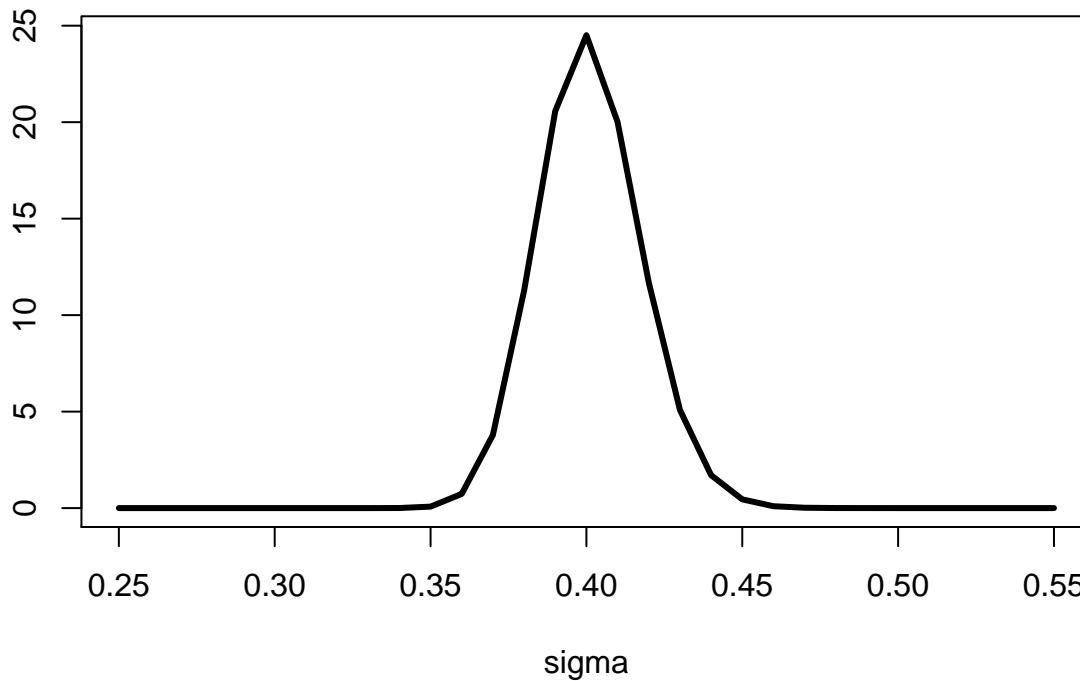
```
plot(beta1Grid, margPostBeta1M3, xlab = "beta1", ylab="", type = "l", lwd = 3)
```



```
plot(beta2Grid, margPostBeta2M3, xlab = "beta2", ylab="", type = "l", lwd = 3)
```



```
plot(sigmaGrid, margPostSigmaM3, xlab = "sigma", ylab="", type = "l", lwd = 3)
```



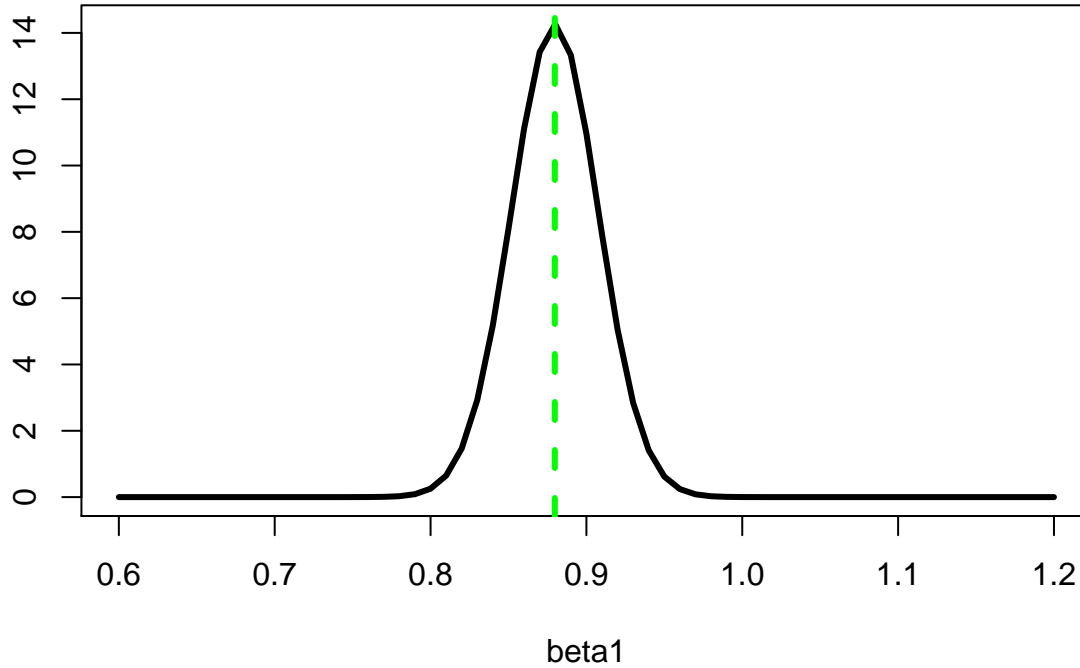
β_0 final grid region: $\{1.50, 1.55, \dots, 3.95, 4.00\}$ \setminus β_1 final grid region: $\{0.60, 0.61, \dots, 1.19, 1.2\}$ \setminus β_2 final grid region: $\{-0.20, -0.19, \dots, 0.39, 0.40\}$ \setminus σ final grid region: $\{.25, 0.26, \dots, 0.54, 0.55\}$ \setminus

Step 3

```
meanBeta1 = sum(margPostBeta1M3 * beta1Grid * stepBeta1Grid)
meanBeta2 = sum(margPostBeta2M3 * beta2Grid * stepBeta2Grid)

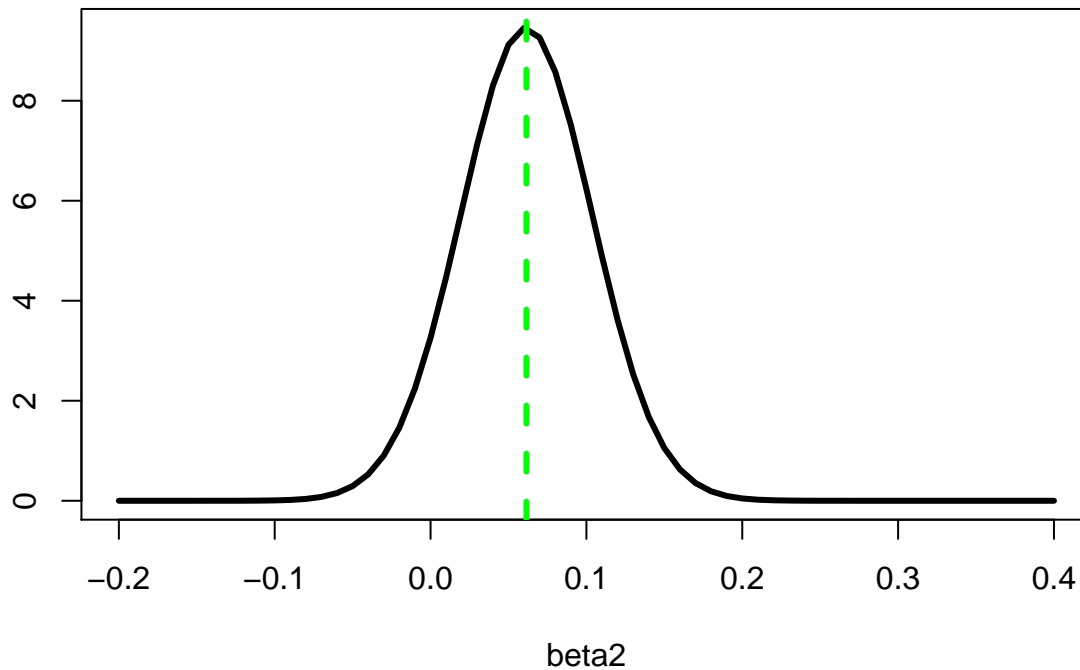
plot(beta1Grid, margPostBeta1M3, xlab = "beta1", ylab="", type = "l", lwd = 3)
```

```
abline(v = meanBeta1, lwd = 3, lty = 2, col = "green")
```



```
plot(beta2Grid, margPostBeta2M3, xlab = "beta2", ylab="", type = "l", lwd = 3)
abline(v = meanBeta2, lwd = 3, lty = 2, col = "green")
```

```
library(lattice)
```

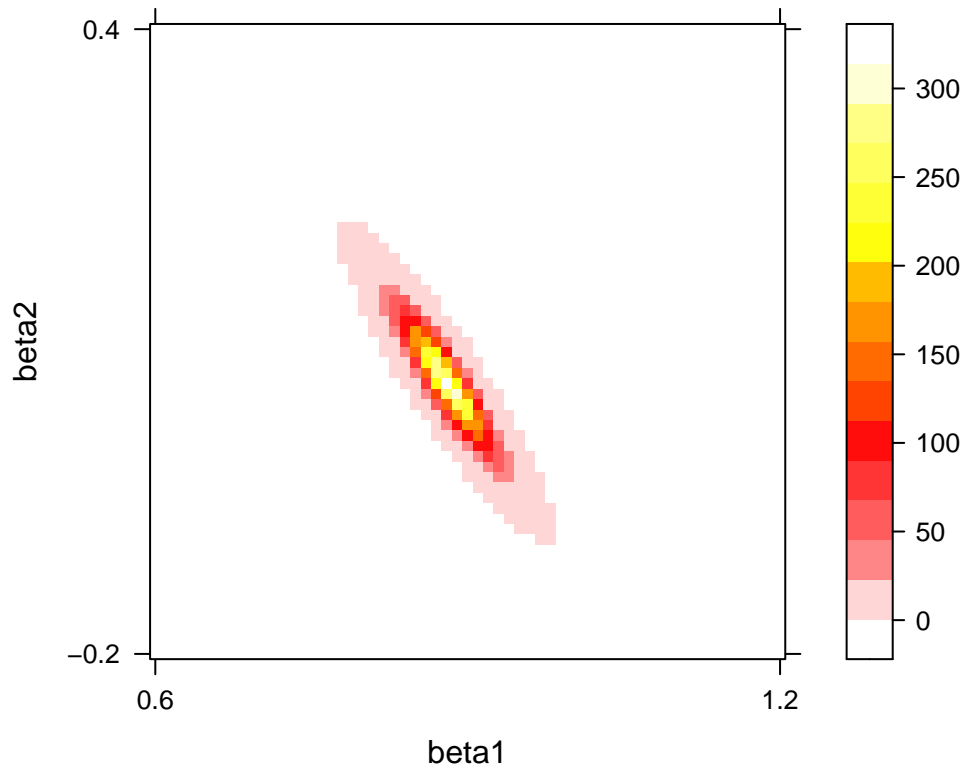


```
new.palette = colorRampPalette(c("white","red","yellow","white"), space="rgb")
jointPost = apply(postM3, c(2,3), sum)
jointPost = jointPost / (stepBeta1Grid * stepBeta2Grid * sum(jointPost))
levelplot(jointPost, col.regions=new.palette(20),
```

```

xlab = "beta1", ylab = "beta2", scales=list(
  x=list(at=c(1,length(beta1Grid)),labels=c(0.6, 1.2)),
  y=list(at=c(1,length(beta2Grid)),labels=c(-0.2, 0.4)))

```



Step 4

This joint posterior has this shape because we know that BodySize and Instar are correlated. So, considering our likelihood definition such that β_1 is associated with $\log(\text{BodySize})$ and β_2 is associated with Instar, we can see that if beta1 is greater then beta2 needs to decrease in order to account for the increased positive shift (and vice versa).

Step 5

```

prob = 0
for (i in 1:length(beta1Grid)) {
  for (j in 1:length(beta2Grid)) {
    if (beta1Grid[i] > 0) {
      if (beta2Grid[j] > 0) {
        prob = prob + jointPost[i,j]
      }
    }
  }
}
prob = prob / sum(jointPost)
round(prob, 3)

```

```
## [1] 0.91
```

Question 2

Step 1

It seems like we should expect that this model would lead to a posterior distribution of β_1 that is shifted upwards because we no longer consider BodySize with another β , and since β_1 and β_2 were mostly positive in the previous model, it makes sense that this new β_1 will need to make up for the upward shift from their positive impact. Logically, the new posterior distribution associated with Instar needs to make up for the positive effects we saw from both Instar and BodySize previously.

Step 2

```
data = read.csv("~/Desktop/MetabolicRate.csv")
logMrate = log(data$Mrate)
logBodySize = log(data$BodySize)
Instar = data$Instar

stepSize = function(grid) {
  if (length(grid)==1) {
    step = 1
  }
  else {
    step = (max(grid) - min(grid)) / (length(grid) - 1)
  }
  return(step)
}

buildPriorUnivar = function(beta0Grid,betaPGrid,sigmaGrid) {
  nBeta0Grid = length(beta0Grid)
  nBetaPGrid = length(betaPGrid)
  nSigmaGrid = length(sigmaGrid)
  prior = array( rep(1, nBeta0Grid * nBetaPGrid * nSigmaGrid ),
    dim = c(nBeta0Grid, nBetaPGrid, nSigmaGrid ))
  for (nB0 in 1:nBeta0Grid) {
    for (nBP in 1:nBetaPGrid) {
      for (nSig in 1:nSigmaGrid) {
        prior[nB0,nBP, nSig] = 1 / nSig^2
      }
    }
  }
  return(prior)
}

likelihoodUnivar = function(y,xP, b0L, bPL, sL){
  loglike = sum(log(dnorm(y-b0L-bPL*xP, mean = 0, sd=sL)))
  like = exp(loglike)
  return(like)
}

compPostUnivar = function(y,xP, prior, beta0Grid,betaPGrid,sigmaGrid) {
  nBeta0Grid = length(beta0Grid)
  nBetaPGrid = length(betaPGrid)
  nSigmaGrid = length(sigmaGrid)
  post = array( rep(-1, nBeta0Grid * nBetaPGrid * nSigmaGrid ),
```



```

        dim = c(nBeta0Grid, nBetaPGrid, nSigmaGrid ))
    for (nBeta0 in 1:nBeta0Grid) {
        b0 = beta0Grid[nBeta0]
        for (nBetaP in 1:nBetaPGrid) {
            bP = betaPGrid[nBetaP]
            for (nSigma in 1:nSigmaGrid) {
                s = sigmaGrid[nSigma]
                post[nBeta0,nBetaP,nSigma] = likelihoodUnivar(y,xP,b0,bP,s) *
                    prior[nBeta0,nBetaP,nSigma]
            }
        }
    }
    post = post / (sum(post) * stepSize(beta0Grid) * stepSize(betaPGrid) *
        stepSize(sigmaGrid))
    return(post)
}

stepBeta0Grid = 0.05
stepBeta1Grid = 0.01
stepSigmaGrid = 0.01
beta0Grid = seq(-5, 2, by = stepBeta0Grid)
beta1Grid = seq(0.5, 2, by = stepBeta1Grid)
sigmaGrid = seq(0.0, 2, by = stepSigmaGrid)

priorM1 = buildPriorUnivar(beta0Grid, beta1Grid, sigmaGrid)
postM1 = compPostUnivar(logMrate, Instar, priorM1, beta0Grid, beta1Grid,
    sigmaGrid)

margPostBeta0M1 = apply(postM1,c(1),sum)
margPostBeta0M1 = margPostBeta0M1 / (sum(margPostBeta0M1) * stepSize(beta0Grid))
margPostBeta1M1 = apply(postM1,c(2),sum)
margPostBeta1M1 = margPostBeta1M1 / (sum(margPostBeta1M1) * stepSize(beta1Grid))
margPostSigmaM1 = apply(postM1,c(3),sum)
margPostSigmaM1 = margPostSigmaM1 / (sum(margPostSigmaM1) * stepSize(sigmaGrid))

```

Step 3

```
paste("Bivariate model mean:", round(meanBeta2, 3))
```

```
## [1] "Bivariate model mean: 0.062"
```

```
newMeanBeta1 = sum(margPostBeta1M1 * beta1Grid * stepBeta1Grid)
```

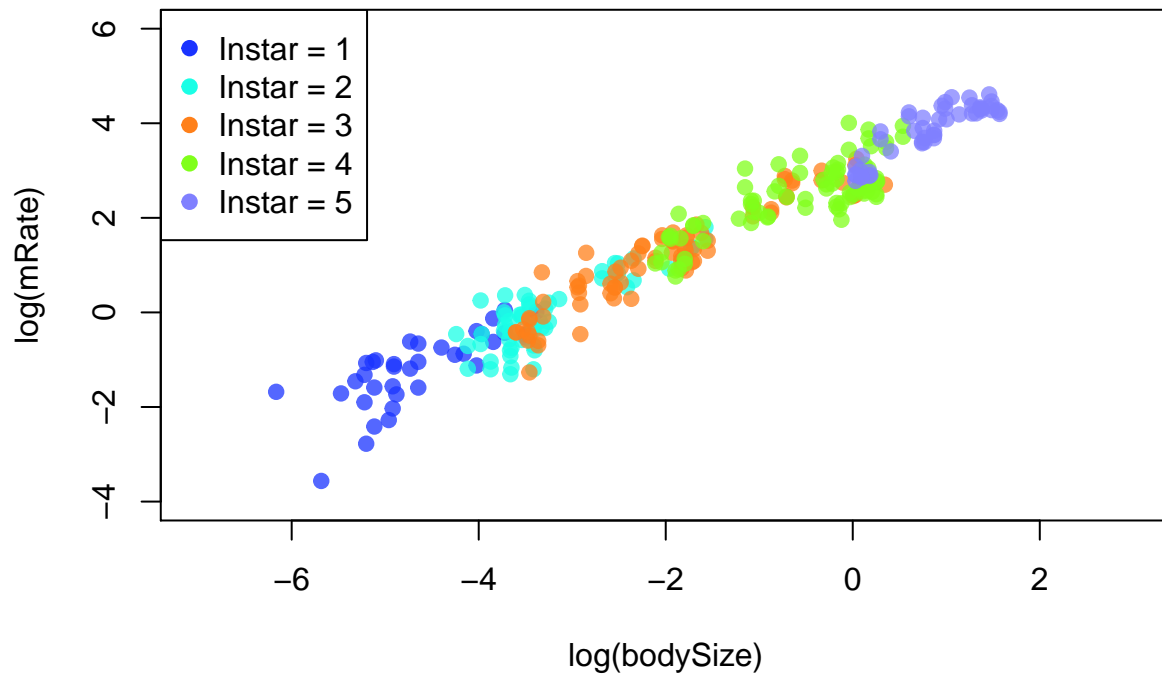
```
paste("Univariate model mean:", round(newMeanBeta1, 3))
```

```
## [1] "Univariate model mean: 1.254"
```

We see that the the mean marginal posterior for this beta is much more positive in the univariate model compared to the bivariate model. Again, this makes sense because this beta needs to account for the positive impact that both beta1 and beta2 have in the bivariate model. Thus, beta1 in the univariate model needs to have a much stronger positive impact relative to the beta2 in the bivariate model.

Step 4

```
new_data = subset(data, data$Instar == 1)
plot(log(new_data$BodySize), log(new_data$Mrate),
     xlab = "log(bodySize)",
     ylab = "log(mRate)",
     xlim = c(-7, 3),
     ylim = c(-4, 6),
     pch = 19,
     col = rgb(red = 0.1, green = 0.2, blue = 1.0, alpha = 0.75))
new_data = subset(data, data$Instar == 2)
points(log(new_data$BodySize), log(new_data$Mrate),
       pch = 19,
       col = rgb(red = 0.1, green = 1.0, blue = 0.9, alpha = 0.75))
new_data = subset(data, data$Instar == 3)
points(log(new_data$BodySize), log(new_data$Mrate),
       pch = 19,
       col = rgb(red = 1.0, green = 0.5, blue = 0.1, alpha = 0.75))
new_data = subset(data, data$Instar == 4)
points(log(new_data$BodySize), log(new_data$Mrate),
       pch = 19,
       col = rgb(red = 0.5, green = 1.0, blue = 0.1, alpha = 0.75))
new_data = subset(data, data$Instar == 5)
points(log(new_data$BodySize), log(new_data$Mrate),
       pch = 19,
       col = rgb(red = 0.5, green = 0.5, blue = 1.0, alpha = 0.75))
legend("topleft",
      c("Instar = 1", "Instar = 2", "Instar = 3", "Instar = 4", "Instar = 5"),
      col = c(
        rgb(red = 0.1, green = 0.2, blue = 1.0, alpha = 1),
        rgb(red = 0.1, green = 1.0, blue = 0.9, alpha = 1),
        rgb(red = 1.0, green = 0.5, blue = 0.1, alpha = 1),
        rgb(red = 0.5, green = 1.0, blue = 0.1, alpha = 1),
        rgb(red = 0.5, green = 0.5, blue = 1.0, alpha = 1)
      ),
      pch = c(19))
```



Following the scatterplot, we see that both life stage and caterpillar body size both positively determine metabolic rate (increase in body size corresponds to increase in metabolic rate, increase in life stage corresponds to increase in metabolic rate). Furthermore, we confirm that instar and bodySize are positively correlated with mRate. Therefore, both variables are significant in determining metabolic rate, but it seems that choosing bodySize in a univariate model over instar would produce a better model.