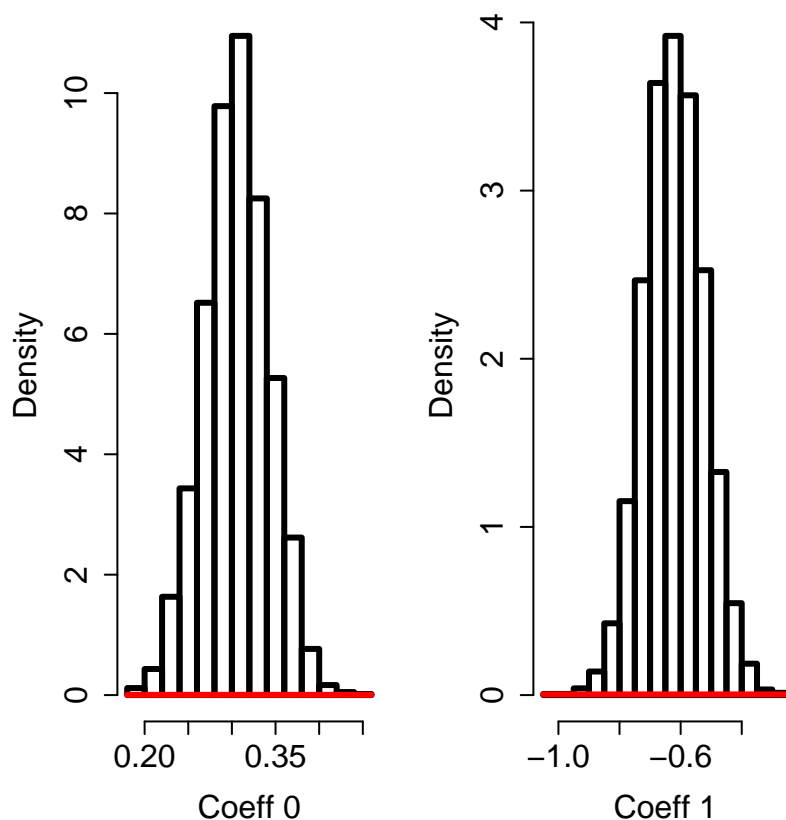


Homework 6

Casey Gibson

3/2/2018

Problem 1



```
print ("B0")  
  
## [1] "B0"  
  
print (signif(mean(mod2$BUGSoutput$sims.array[,,"b0"]),3))  
  
## [1] 0.305  
  
print (signif(quantile(mod2$BUGSoutput$sims.array[,,"b0"],c(.025,.975)),3))  
  
## 2.5% 97.5%  
## 0.234 0.376  
  
print ("B1")  
  
## [1] "B1"  
  
print (signif(mean(mod2$BUGSoutput$sims.array[,,"b1"]),3))  
  
## [1] -0.622
```

```
print (signif(quantile(mod2$BUGSoutput$sims.array[,,"b1"],c(.025,.975)),3))
```

```
##    2.5%  97.5%
## -0.819 -0.427
```

```
"
model {
  for( i in 1:n) {
    logit(p.i[i]) <- (
      # different notation; using betas for main effects and b's for varying effects
      b0
      + b1*c.dist100[i]
    )
    y.i[i] ~ dbern(p.i[i])
  }

  b0 ~ dnorm(0.0,1.0E-4)
  b1 ~ dnorm(0.0,1.0E-4)
} # end model
"
```

In terms of odds ratios we see that there is a $\exp(.305) = 1.36$ expected multiplicative effect on the odds of switching wells for households that are \bar{d} meters away from the nearest safe well. This is because when $d_i = \bar{d}$, β_{a1} drops out. We expect to see an effect between $(\exp(.231), \exp(.378)) = (1.26, 1.46)$ 95% of the time.

Similarly, we expect a $\exp(-0.620) = .5379$ multiplicative effect on the odds ratio of switching for every unit increase in distance from the nearest safe well. We expect to see an effect somewhere between $(\exp(-0.814), \exp(-0.438)) = (.443, .645)$ 95% of the time.

Problem 2

a)

```
"
model {
  for( i in 1:n) {
    logit(p.i[i]) <- (
      # different notation; using betas for main effects and b's for varying effects
      a0 + a1*c.dist100[i] + a2*c.arsenic[i]
    )
    y.i[i] ~ dbern(p.i[i])
  }

  a0 ~ dnorm(0.0,1.0E-4)
  a1 ~ dnorm(0.0,1.0E-4)
  a2 ~ dnorm(0.0,1.0E-4)

} # end model
"
```

```
"
model {
  for( i in 1:n) {
    logit(p.i[i]) <- (
```

```

# different notation; using betas for main effects and b's for varying effects
b0 + b1*c.dist100[i] + b2*c.arsenic[i] + b3*c.arsenic[i]*c.dist100[i]
)
y.i[i] ~ dbern(p.i[i])
}

b0 ~ dnorm(0.0,1.0E-4)
b1 ~ dnorm(0.0,1.0E-4)
b2 ~ dnorm(0.0,1.0E-4)
b3 ~ dnorm(0.0,1.0E-4)

} # end model
"

```

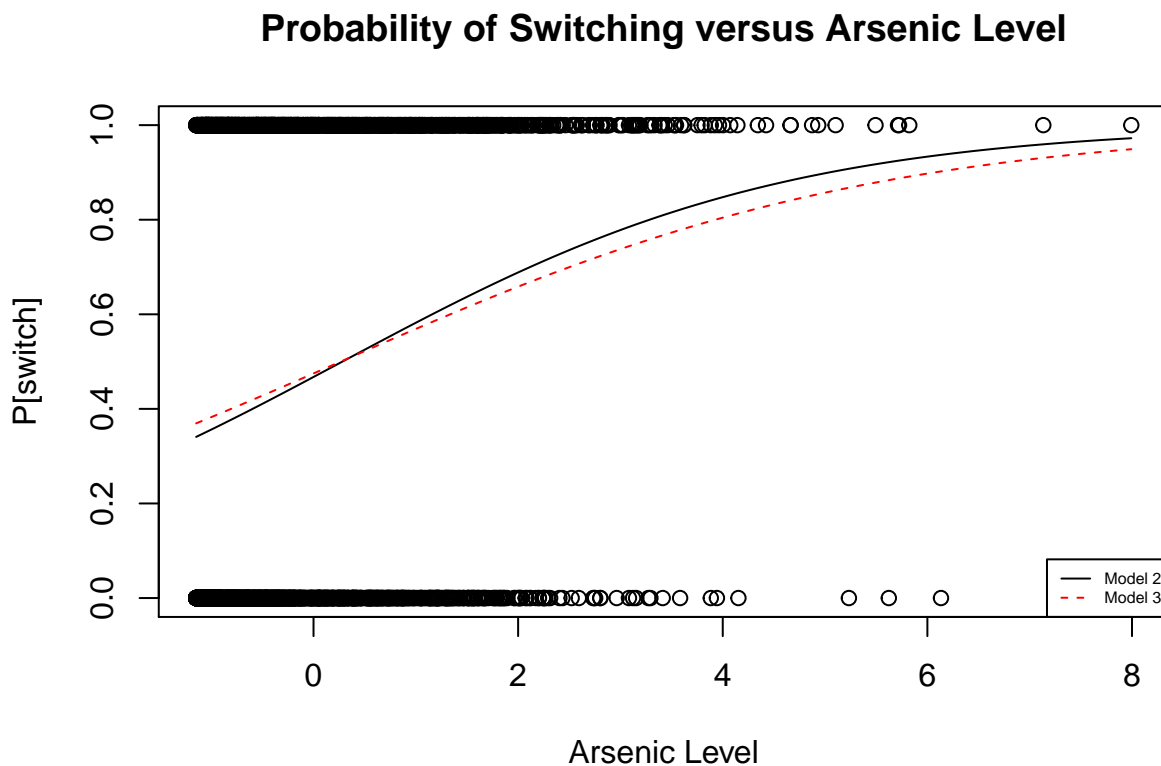
b)

```

invlogit <- function(x) {exp(x)/(1 + exp(x))}

plot(x = c.arsenic, y = y.i, ylab = "P[switch]", xlab = "Arsenic Level", main = "Probability of Switching versus Arsenic Level",
      curve(invlogit(a0_mean + a1_mean*(1-mean(dist100)) + a2_mean*x), add=TRUE)
      curve(invlogit(b0_mean + b1_mean*(1-mean(dist100)) + b2_mean*x + b3_mean*(1-mean(dist100))*x), add=TRUE)
      legend("bottomright", legend = c("Model 2", "Model 3"), col = c(1,2), lty = c(1,2), cex = .5)

```



We can see that the interaction term accounts for the fact that the importance of arsenic levels decrease as distance increases. That is people who are further away from safe wells consider arsenic less important when making the decision to switch.

Problem 4

a)

```
"
model {
  for( i in 1:n) {
    logit(p.i[i]) <- (
      # different notation; using betas for main effects and b's for varying effects
      b0 + b1*c.dist100[i] + b2*c.log.arsenic[i] + alpha.j[getj1.i[i]]
    )
    y.i[i] ~ dbern(p.i[i])
  }
  for (j in 1:J){
    alpha.j[j] ~ dnorm(0, 1/sigma_alpha^2)
  }
  b0 ~ dnorm(0.0,1.0E-4)
  b1 ~ dnorm(0.0,1.0E-4)
  b2 ~ dnorm(0.0,1.0E-4)
  sigma_alpha ~ dunif(0,200)

} # end model

"
```

We can see that the variance of the first group mean is smaller than that of the second group mean

```
print ("Grouping 1 sigma alpha")
```

```
## [1] "Grouping 1 sigma alpha"
```

```
print (sigma_alpha_mean_1)
```

```
## [1] 0.1334997
```

```
print ("Grouping 2 sigma alpha")
```

```
## [1] "Grouping 2 sigma alpha"
```

```
print (sigma_alpha_mean_2)
```

```
## [1] 1.049601
```

This make sense because we have reduced the variance within each group by putting households that are close to each other in the dataset , under the same village cluster. As long as there is any correlation between row i and row i+1 (such as survey collectors going from one town, to the town adjacent) this will reduce the variance within the cluster.

b)

```
linpred <-b0_mean + (.5-mean(dist100))*b1_mean +(log(1.2)-mean(log.arsenic))*b2_mean
```

```
invlogit <- function(x) {exp(x)/(1 + exp(x))}
```

```
print ("J = 1")
```

```
## [1] "J = 1"
```

```
quantile(invlogit(linpred+mcmc.array[,,"alpha.j[1]"]),c(.025,.975))
```

```
##      2.5%      97.5%  
## 0.3302953 0.9038543
```

```
print ("J = 2")
```

```
## [1] "J = 2"
```

```
quantile(invlogit(linpred+mcmc.array[,,"alpha.j[2]"]),c(.025,.975))
```

```
##      2.5%      97.5%  
## 0.4954345 0.9539468
```

```
print ("J = 3")
```

```
## [1] "J = 3"
```

```
quantile(invlogit(linpred+mcmc.array[,,"alpha.j[3]"]),c(.025,.975))
```

```
##      2.5%      97.5%  
## 0.6456555 0.9651341
```

```
print ("J = 4")
```

```
## [1] "J = 4"
```

```
quantile(invlogit(linpred+mcmc.array[,,"alpha.j[4]"]),c(.025,.975))
```

```
##      2.5%      97.5%  
## 0.5268797 0.9274944
```

```
print ("J = 5")
```

```
## [1] "J = 5"
```

```
quantile(invlogit(linpred+mcmc.array[,,"alpha.j[5]"]),c(.025,.975))
```

```
##      2.5%      97.5%  
## 0.4599345 0.8672441
```