Tyler Poelking
April 27th 2018
Stats 3303

Statistics 3303 Final Exam

Part I

Model and Variable Definition:

Model Used for predicting K9C9 status, with infected as outcome.

Let $y = \{y_{ic} ; i = 1, \ldots N^p, c = 1, \ldots N^c\}$ where N^p corresponds to the number of subjects tested in each country (100), and N^c corresponds to the number of countries in the clinical trial (10).

$$p(\underline{y} | \underline{\alpha}, \underline{\beta}) = \prod_{c=1}^{N^c} \prod_{i=1}^{N^p} p(y_{ic} | \alpha_c, \beta_c)$$

$$y_{ic} | \alpha_c, \beta_c \sim Bern(\theta_{ic}) \text{ and } \text{logit}(\theta_{ic}) = \alpha_c + \beta_c E2K_{ic}$$

$\underline{\alpha} = \{\alpha_1, \ldots \alpha_{N^c}\}$ captures the country specific log odds that a subject whose E2K test was negative (0) is infected according to the diagnostic test.

$\underline{\beta} = \{\beta_1, \ldots \beta_{N^c}\}$ captures the country specific difference in log odds of a subject who scored positive on the E2K test's showing as infected on the diagnostic test relative to subjects who scored negative on the E2K test.

$E2K_{ic}$ corresponds to subject i in country c's E2K status. 1 = positive, 0 = negative

Joint Distribution: $p(\underline{\alpha}, \underline{\beta} | \mu_\alpha, \sigma^2_\alpha, \mu_\beta, \sigma^2_\beta) = \prod_{c=1}^{N^c} p(\alpha_c | \mu_\alpha, \sigma^2_\alpha) p(\beta_c | \mu_\beta, \sigma^2_\beta)$

$\alpha_c | \mu_\alpha, \sigma^2_\alpha \sim N(\mu_\alpha, \sigma^2_\alpha)$, $\beta_c | \mu_\beta, \sigma^2_\beta \sim N(\mu_\beta, \sigma^2_\beta)$, $p(\mu_\alpha, \sigma^2_\alpha, \mu_\beta, \sigma^2_\beta) = p(\mu_\alpha) p(\sigma^2_\alpha) p(\mu_\beta) p(\sigma^2_\beta)$ ↳ prior independence

and $\mu_\alpha \sim N(0, 9)$, $\mu_\beta \sim N(0, 9)$, $\sigma^2_\alpha \sim Unif(0, 9)$, $\sigma^2_\beta \sim Unif(0, 9)$

μ_α: mean (across country) of the log odds that a E2K negative subject is infected.
μ_β: mean (across country) of the change in log odds that an E2K positive subject is infected relative to an E2K negative subject.
σ^2_α: captures the deviation across countries in the country specific log odds that a E2K negative subject is infected about the overall mean.
σ^2_β: captures the deviation across countries in the country specific change in log odds that an E2K positive subject is infected relative to an E2K negative subject about the overall mean.

Tyler Poelking
April 27th 2018
Stats 3303

Model Details:

The initial values were: alpha = 0 for all 10 countries, beta = 0 for all 10 countries, mu_alpha = 0, mu_beta = 0, sigma2_alpha = 1, sigma2_beta = 1

The number of iterations for tuning and burn-in were set to 5000. Two chains ran for an additional 2,500 iterations each. The trace plots show both chains appear to be sampling from the same distributions and are evenly bouncing around a centralized value with all apparent patterns repeating, thus providing evidence the algorithm converged.

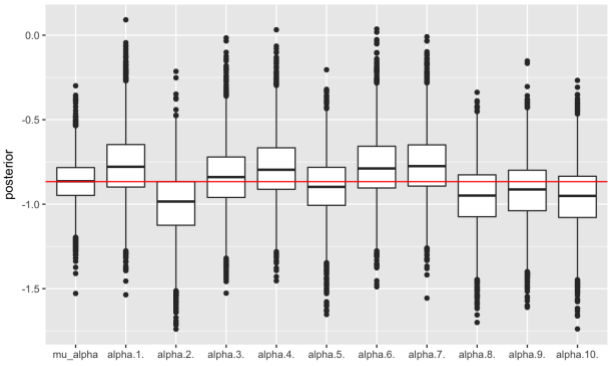
Conditional Dependence:
In this model, we assume that conditional on Country and EZK test result, patient's infected statuses are independent. Furthermore, we assume that conditional on their corresponding mean's and variances, alpha and beta follow normal distributions. We also assume that the priors on mu_beta and mu_alpha (see interpretations above) follow a normal with mean 0 and variance 9 and that this is reasonable and allows for convergence to the true posterior.

We know the probability of infection of an individual at a particular country through:

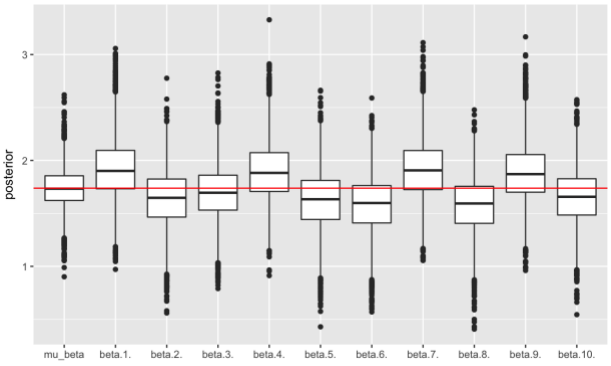
1. The particular country's log odds that a subject whose EZK test was negative is infected.
2. The particular country's difference in log odds of a subject who scored positive on the EZK test's showing as infected relative to subjects who scored negative
3. The individual's EZK test score.

Interpretations of results

Alpha Boxplots



Beta Boxplots



**Note: When referring to a patient as being infected or not infected, we refer to the infected status derived from the highly accurate diagnositc test.*

The alphas capture the country-specific change in log odds that a subject whose EZK test is negative is infected. As a country's alpha increases, the likelihood of a subject who scored negative on the EKZ test actually having influenza increases.

The betas capture the country-specific change in log odds that a subject whose EZK test is positive is infected, relative to the subject whose EZK test is negative. Adding a country

specific's alpha and beta together derives the country specific log odds that, given the subject scored positive on the EZK test, the subject is infected.

The mean of mu_alpha is -0.867. Taking the inverse logit, we get that an estimated 29.6% of subjects who scored negative on the EZK test actually have influenza. Country A had a mean alpha of -0.77 which corresponds to 31.7% of subjects who scored negative on EZK actually having influenza. Therefore, country A's performance in this regard was worse than average. Other countries who had mean alphas above average were C, D, F and G. F and G were the worst at 31.5% and 31.7% respectively. Inversely, countries with lower than average alphas performed better in this regard. Countries, B, E, H, I, and J were in this category, with country B having the lowest percentage of 27.1%.

The mean of mu_alpha + mu_beta is 0.87. Taking the inverse logit, we get that an estimated 70.6% of subjects who scored positive on the EZK test actually have influenza. Country A had a mean alpha+beta corresponding to 76.1% of subjects who scored positive on EZK actually having influenza.

Therefore, country A's performance in this regard was better than average. Other countries who had mean alpha+betas above average were A, D, G, and I. A and D were the best at 76.1% and 75.5% respectively. Inversely, counties with lower than average alpha+betas performed worse in this regard. Counties B, C, E, F, H, and J were in this category, with country H and B having the lowerest percentage at 65.1% and 65.4% respectively.

The variances of alpha, beta, and their corresponding means/variances is shown below. All of these parameters varied quite little. This fact allows us to be more confident in the conclusions we draw for the insurance company.

alpha.1.	alpha.2.	alpha.3.	alpha.4.	alpha.5.	alpha.6.	alpha.7.	alpha.8.	alpha.9.	alpha.10.
0.036217279	0.036394138	0.029783604	0.033587261	0.030916763	0.032489806	0.034621165	0.033530698	0.031829840	0.035641905
beta.1.	beta.2.	beta.3.	beta.4.	beta.5.	beta.6.	beta.7.	beta.8.	beta.9.	beta.10.
0.082132402	0.076291855	0.066666851	0.076466615	0.078606460	0.074137925	0.083597662	0.073420604	0.077813738	0.069302538
mu_alpha	mu_beta	sigma2_alpha	sigma2_beta						
0.015927759	0.035552429	0.004886335	0.027791559						

Part II

Using the proper parameters in the model output, we obtain:

P(Infected = 1| EZK = 0) = inv.logit(mean(alpha.4.)) = 0.312
P(Infected = 0| EZK = 0) = 1- P(Infected = 1| EZK = 0) = 0.688
P(Infected = 0| EZK = 1) = inv.logit(mean(alpha.4.) + mean(beta.4.)) = 0.752
P(Infected = 0| EZK = 0) = 1- P(Infected = 0| EZK = 1) = 0.248

Below we use the ER (expected reward) methodology that we have done in class to compute the expected loss per patient for the two potential actions the insurance company can take. For each potential outcome, we sum the product of the probability and cost associated with each outcome.

If the insurance provider decides to treat patients who test positive for K9C9 using the the EZK test, the estimate cost/person is: 0.312 * 1490 + 0.688*0 + 0.752 *457 + 0.248*457 = **$921.42.**

Tyler Poelking
April 27th 2018
Stats 3303

To find the estimated cost when the insurance provider decides to NOT treat patients who test positive for K9C9 using the EZK test, we need the estimated proportion of people in Country D who are infected. This can be found by taking the mean of each of the 100 thetas corresponding to the subjects from Country D and finding the proportion of these means that are above 0.5, which correspond to the person, on average, being diagnosed as infected by the model. This proportion ends up being 0.51. We can then multiply that proportion by 1490 to get an estimated cost = 0.51 * 1490 = **$879.10.**

The insurance company should therefore not treat patients who test positive for K9C9 using the EZK test if they want to minimize costs.

Appendix

```
---
title: "Final"
author: "Tyler Poelking"
date: "4/21/2018"
output:
  word_document: default
  html_document: default
---

```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
```


```{r Setup, include =FALSE}
set.seed(23422432)
library(rjags)
library(boot)
library(ggplot2)
library(grid)
library(gridExtra)
library(reshape2)
library(plyr)
library(dplyr)

```

Part I

```{r Read In Data and Set Some Parameters}
```
```

Tyler Poelking
April 27th 2018
Stats 3303

```
fluData <- read.table( "~/Desktop/All Stuff/School Stuff/STATS/3303/final/flu.txt", header=T )
attach(fluData)

nTotal <- length(Infected)
nC <- length(unique(Country))
countryNumeric = as.numeric(Country)
```

```{r Jags Model Setup}
# create objects for JAGS
dataList <- list( "Infected" = Infected,
                  "EZK" = EZK,
                  "country" = countryNumeric,
                  "nC" = nC,
                  "nTotal" = nTotal)
# list of parameters to be monitored
parameters <- c(
                "alpha",
                "beta",
                "mu_alpha",
                "mu_beta",
                "sigma2_alpha",
                "sigma2_beta",
                "theta")
# set initial values
initsValues <- list(
                   "alpha" = rep(0, nC),
                    "beta" = rep(0, nC),
                   "mu_alpha" = 0,
                    "mu_beta" = 0,
                    "sigma2_alpha" = 1,
                    "sigma2_beta" = 1)

# number of iteration for "tuning"
adaptSteps <- 5000
# number of iterations for "burn-in"
burnInSteps <- 5000
# number of chains to run
nChains <- 2
# total number of iterations to save
numSavedSteps <- 5000
# "thinning" (1 = keep every interation)
thinSteps <- 1
# iterations per chain
ITER <- ceiling( (numSavedSteps * thinSteps) / nChains )
```

Tyler Poelking
April 27th 2018
Stats 3303

```
# -------------
# Run JAGS
# -------------
# create, initialize, and adapt the model
jagsModel <- jags.model( "finalModel.txt",
                    data = dataList,
                    inits = initsValues,
                    n.chains = nChains,
                    n.adapt = adaptSteps )
```

```{r Run Jags}
# burn-in the algorithm
update( jagsModel,
      n.iter = burnInSteps )
# run algorithm to get interations for inference
codaSamples <- coda.samples( jagsModel,
                        variable.names = parameters,
                        n.iter = ITER,
                        thin = thinSteps )

# -------------
# Look at posterior samples
# -------------
# make a dataframe with the posterior samples
mcmcChainDF <- data.frame( as.matrix( codaSamples,
                              iters = T,
                              chains = T ) )
# create a vector with the variable names
varNames <- names( mcmcChainDF )[3:( 26 )]
# number of variables
nVars <- length( varNames )
mcmcChainDF$CHAIN <- as.factor(mcmcChainDF$CHAIN)
# construct trace plots
p <- list()
for( k in 1:nVars )
{
  plot_frame <- mcmcChainDF
  plot_frame$dep_var <- mcmcChainDF[ , varNames[k]]
  p[[k]] <- ggplot( plot_frame,
                aes( x = ITER,
                    y = dep_var))  +
    geom_line( aes( color = CHAIN ) ) +
    labs( y = varNames[k] )
}
```

Tyler Poelking
April 27th 2018
Stats 3303

```{r Trace Plots, fig.width = 8, fig.height=20}
do.call( grid.arrange, c( p, list("ncol" = 1) ) )
```

The initial values are:

alpha = 0 for all 10 countries
beta = 0 for all 10 countries
mu_alpha = 0
mu_beta = 0
sigma2_alpha = 1
sigma2_beta = 1

The number of iterations for tuning and burn-in were set to 5000. Two chains ran for an additional 2,500 iterations each. The trace plots above show both chains appear to be sampling from the same distributions and are evenly bouncing around a centralized value with all apparent patterns repeating, thus providing evidence the algorithm converged.

Conditional Dependence:
In this model, we assume that if we know both the probability of infection of an individual at a particular country, the responses are independent.

We know the probability of infeciton of an individual at a particular country through the particular country's log odds that a subject whose EZK test was negative is infected, the particular country's difference in log offs of a subjet who scored positive on the EZK test's showing as infected relative to subjects who scored negative, and the individual's EZK test score.

```{r Results}
#boxplot of alphas and intercept
alphaPostDFreshape <- melt( mcmcChainDF,
                    id.vars = "ITER",
                    measure.vars = c("mu_alpha",
                                     "alpha.1.",
                                     "alpha.2.",
                                     "alpha.3.",
                                     "alpha.4.",
                                     "alpha.5.",
                                     "alpha.6.",
                                     "alpha.7.",
                                     "alpha.8.",
                                     "alpha.9.",
                                     "alpha.10."))
ggplot(alphaPostDFreshape,
      aes(x = variable, y = value )) +
```

```
  geom_boxplot() +
  ylab( "posterior" ) +
  xlab( "" ) + geom_hline(yintercept = mean(mcmcChainDF$mu_alpha), color = 'red')

#boxplot of betas
betaPostDFreshape <- melt( mcmcChainDF,
                    id.vars = "ITER",
                    measure.vars = c("mu_beta",
                                     "beta.1.",
                                     "beta.2.",
                                     "beta.3.",
                                     "beta.4.",
                                     "beta.5.",
                                     "beta.6.",
                                     "beta.7.",
                                     "beta.8.",
                                     "beta.9.",
                                     "beta.10."))
ggplot(betaPostDFreshape,
       aes(x = variable, y = value )) +
  geom_boxplot() +
  ylab( "posterior" ) +
  xlab( "" )+ geom_hline(yintercept =  mean(mcmcChainDF$mu_beta), color = 'red')
```

*Note: When referring to a patient as being infected or not infected, we refer to the infected status derived from the highly accurate diagnositc test.

The alphas capture the country-specific change in log odds that a subject whose EZK test is negative is infected. As a country's alpha increases, the likelihood of a subject who scored negative on the EKZ test actually having influenza increases.

```{r Confirming Alpha Interpretations}

#Get alpha based on entire data
numer = fluData %>% filter(Infected ==1 & EZK ==0)
denom = fluData %>% filter(EZK==0)
logit(length(numer[[1]])/length(denom[[1]]))
inv.logit(mean(mcmcChainDF$mu_alpha))

#iterate through countries and get their true 'alpha '
for (c in c("A", "B", "C","D", "E", "F", "G", "H","I", "J")){
   country = subset(fluData, Country == c)
   country_num <- country %>%
        filter(EZK ==0 & Infected ==1)
   country_den <- country %>%
        filter(EZK==0)
```

```
   print(length(country_num[[1]])/length(country_den[[1]]))

}

#Find corresponding values for each country to compare
print("Country's estimated inv.logit(alpha) based on model: ")
cat("Country A: ",inv.logit(mean(mcmcChainDF$alpha.1.)))
cat("Country B: ",inv.logit(mean(mcmcChainDF$alpha.2.)))
cat("Country C: ",inv.logit(mean(mcmcChainDF$alpha.3.)))
cat("Country D: ",inv.logit(mean(mcmcChainDF$alpha.4.)))
cat("Country E: ",inv.logit(mean(mcmcChainDF$alpha.5.)))
cat("Country F: ",inv.logit(mean(mcmcChainDF$alpha.6.)))
cat("Country G: ",inv.logit(mean(mcmcChainDF$alpha.7.)))
cat("Country H: ",inv.logit(mean(mcmcChainDF$alpha.8.)))
cat("Country I: ",inv.logit(mean(mcmcChainDF$alpha.9.)))
cat("Country J: ",inv.logit(mean(mcmcChainDF$alpha.10.)))
```

The betas capture the country-specific change in log odds that a subject whose EZK test is positive is infected, relative to the subject whose EZK test is negative. Adding a country specific's alpha and beta together derives the country specific log odds that, given the subject scored positive on the EZK test, the subject is infected.

```{r Confirming Beta Interpretations}
#For entire data
with_infl = length(subset(fluData, EZK == 1)[[1]])
with_infl_and_ezk_positive = length(subset(fluData, Infected == 1 & EZK ==1)[[1]])
pos_ezk_success_rate = with_infl_and_ezk_positive/with_infl
pos_ezk_success_rate

inv.logit(mean(mcmcChainDF$mu_alpha) + mean(mcmcChainDF$mu_beta))

#iterate through countries and get their true 'alpha + beta'
for (c in c("A", "B", "C","D", "E", "F", "G", "H","I", "J")){
   #subset country A. Based on data
   country = subset(fluData, Country == c)
   country_with_infl = length(subset(country, EZK == 1)[[1]])
   country_with_infl_and_ezk_positive = length(subset(country, Infected == 1 & EZK
==1)[[1]])
   country_success_rate = country_with_infl_and_ezk_positive/country_with_infl
   print(country_success_rate)
}

#Find corresponding values for each country to compare
print("Country's estimated inv.logit(alpha+beta) based on model: ")
```

```
cat("Country A: ", inv.logit(mean(mcmcChainDF$alpha.1.) + mean(mcmcChainDF$beta.1.)))
cat("Country B: ", inv.logit(mean(mcmcChainDF$alpha.2.) + mean(mcmcChainDF$beta.2.)))
cat("Country C: ",inv.logit(mean(mcmcChainDF$alpha.3.) + mean(mcmcChainDF$beta.3.)))
cat("Country D: ",inv.logit(mean(mcmcChainDF$alpha.4.) + mean(mcmcChainDF$beta.4.)))
cat("Country E: ",inv.logit(mean(mcmcChainDF$alpha.5.) + mean(mcmcChainDF$beta.5.)))
cat("Country F: ",inv.logit(mean(mcmcChainDF$alpha.6.) + mean(mcmcChainDF$beta.6.)))
cat("Country G: ",inv.logit(mean(mcmcChainDF$alpha.7.) + mean(mcmcChainDF$beta.7.)))
cat("Country H: ",inv.logit(mean(mcmcChainDF$alpha.8.) + mean(mcmcChainDF$beta.8.)))
cat("Country I: ",inv.logit(mean(mcmcChainDF$alpha.9.) + mean(mcmcChainDF$beta.9.)))
cat("Country J: ",inv.logit(mean(mcmcChainDF$alpha.10.) + mean(mcmcChainDF$beta.10.)))

```

```{r Variance Exploration}
#column variances
apply(mcmcChainDF, 2, var)
```

The interpretation confirmations were done to ensure the accuracy of my understanding of the model. I compared what my model predicted with what the true population was to ensure my understanding. It can be observed that if the alpha as determined by the original data is above the mean alpha across the whole data, the predicted value of alpha will also be above the mean alpha across the whole data. Same goes for values below the mean. This can also be said for my interrpretations of beta+alpha. If the beta+alpha as determined by the original data is above the mean alpha+beta across the whole data, the predicted value of alpha+beta will also be above the mean alpha+beta across the whole data. Same goes for valeus below the mean. Note that this was merely for confirmation of my understanding. No inferences were made using the original data.

The mean of mu_alpha is -0.867. Taking the inverse logit, we get that an estimated 29.6% of subjects who scored negative on the EZK test actually have influenza. Country A had a mean alpha of -0.77 which corresponds to 31.7% of subjects who scored negative on EZK actually having influenza. Therefore, country A's performance in this regard was worse than average. Other countries who had mean alphas above average were C, D, F and G. F and G were the worst at 31.5% and 31.7% respectively. Inversely, countries with lower than average alphas performed better in this regard. Countries, B, E, H, I, and J were in this category, with country B having the lowest percentage of 27.1%.

The mean of mu_alpha + mu_beta is 0.87. Taking the inverse logit, we get that an estimated 70.6% of subjects who scored positive on the EZK test actually have influenza. Country A had a mean alpha+beta corresponding to 76.1% of subjects who scored positive on EZK actually having influenza.
Therefore, country A's performance in this regard was better than average. Other countries who had mean alpha+betas above average were A, D, G, and I. A and D were the best at 76.1% and 75.5% respectively. Inversely, counties with lower than average alpha+betas performed worse in this regard. Counties B, C, E, F, H, and J were in this category, with country H and B having the lowerest percentage at 65.1% and 65.4% respectively.

##Part 2

Tyler Poelking
April 27th 2018
Stats 3303

```{r Finding Necessary Probabilities}
country_D = subset(fluData, Country == "D")

#Given EZK = 0, probability that Infected = 1
p_infected_ezk_0 = inv.logit(mean(mcmcChainDF$alpha.4.))
cat("Given EZK = 0, probability that Infected = 1: ",p_infected_ezk_0)

#Given EZK = 0, probability that Infected = 0
p_not_infected_ezk_0 = 1-p_infected_ezk_0
cat("Given EZK = 0, probability that Infected = 0: ",p_not_infected_ezk_0)

#Given EZK = 1, probability that Infected = 1
p_infected_ezk_1 = inv.logit(mean(mcmcChainDF$alpha.4.) + mean(mcmcChainDF$beta.4.))
cat("Given EZK = 1, probability that Infected = 1: ", p_infected_ezk_1)

#Given EZK = 1, probability that Infected = 0
p_note_infected_ezk_1 = 1-p_infected_ezk_1
cat("Given EZK = 1, probability that Infected = 0: ",p_note_infected_ezk_1)

cat("If the insurance provider decides to treat patients who test positive for K9C9 using the EZK test, the estimated cost/person is: $", round((p_infected_ezk_0 * 1490 + p_not_infected_ezk_0*0 + p_infected_ezk_1*457 + p_note_infected_ezk_1*457),2))

#Percentage of Country D infected
country_D_thetas = mcmcChainDF[327:426]
predictions = data.frame(apply(country_D_thetas, 2, mean))
predictions = predictions[,1]
count_infected = length(predictions[predictions>0.5])
prop_infected_d = count_infected/100
cat("The estimated proportion of country D's population that is infected: ", prop_infected_d)

cat("Using this proportion, we can estimate that, if the insurance provider decides NOT to treat patients who test positive for K9C9 using  the EZK test, the estimated cost/person is: $", prop_infected_d * 1490)


```