Final Project Report

Peike Wang

2023-04-25

A new strain of influenza called K9C9 has emerged and has been detected in humans in 10 countries. To quickly assess the diagnostic ability of an inexpensive diagnostic test named "EZK" for this new strain of influenza, the World Health Organization sponsored a small clinical trial in each of the 10 countries where the virus is endemic. The EZK test is not perfect and can result in false positives and false negatives. To evaluate the diagnostic ability of EZK, 100 subjects were randomly selected in each country and tested for K9C9 using both the highly accurate diagnostic test and the EZK test. The data collected from the clinical trials included variables indicating the K9C9 infection status of each subject according to the highly accurate diagnostic test and the results of the EZK test, as well as the country of residence of the subject. To account for potential genetic variation in the virus, a Bayesian hierarchical model was developed with country-specific parameters to model the probability of a subject having the virus based on the EZK test results.

Then, in order to fit the Bayesian hierarchical model, I define the following variables and unknown parameters. At each N^c country, there are N^i subjects, while $N^c = 10$ and $N^i = 100$.

Let Y_{ic} be the infection status of each subject and X_{ic} be the EZK test result of each subject, for i = 1,...,N^i and c = 1,...,N^c. First, I assume that

$Y_{ic}|\alpha_c, \beta_c \sim$ Bernoulli(θ_c), for i = 1,...,N^i, c = 1,...,N^c.

$\text{logit}(\theta_c) = \alpha_c + \beta_c X_{ic}$, for i = 1,. . . ,$N^i$, c = 1,. . . ,$N^c$.

$\alpha_c \sim \text{Normal}(\mu_\alpha, \sigma_\alpha^2)$, for c = 1,. . . ,$N^c$.

$\beta_c \sim \text{Normal}(\mu_\beta, \sigma_\beta^2)$, for c = 1,. . . ,$N^c$.

where $\mu_\alpha \sim$ Normal(0,9), $\mu_\beta \sim$ Normal(0,9), $\sigma_\alpha \sim$ Uniform(0,3), $\sigma_\beta \sim$ Uniform(0,3).

α_c is the country-specific intercept when the EZK test is negative (X = 0). β_c is the effect of a positive EZK test (X = 1) on the log-odds of being infected with the K9C9 virus, across all countries. μ_α is the average country-specific intercept when the EZK test is negative. σ_α is the standard deviation of the country-specific intercepts. μ_β is the average effect of a positive EZK test (X = 1) on the log-odds of being infected with the K9C9 virus, across all countries. σ_β is the standard deviation of the country-specific effects of a positive EZK test.

I choose Bernoulli model to fit the Y_{ic} because it represents the infection status which is binary with a certain probability. So Bernoulli is the best model and I can define the probability of Bernoulli to show the relationships between infections and EZK. Therefore, the next assumption I make is define the logit of Bernoulli probability as a linear function of X_{ic}, then I can evaluate the diagnostic ability of EZK by estimating the parameter α_c, β_c. In order to estimating these two parameters, I assume they are normally distributed with four specific parameters $\mu_\alpha, \sigma_\alpha^2$ and $\mu_\beta, \sigma_\beta^2$. Finally, I make the prior assumption that these four parameters are both normally or uniformly distributed, because these are relatively non-informative priors that allow for a wide range of possible values.

Then I fit the Bayesian hierarchical model using JAGS in R (model codes can be found in the appendix). Starting value are $\mu_\alpha = 0, \sigma_\alpha = 1, \mu_\beta = 0, \sigma_\beta = 1$. The iterations my algorithm run is 10000 and burns at 2500. I can confirm the convergence of the model by check the

trace and density plots in the appendix. The evidence of convergence is that the trace of parameters converge to their true value and their densities are normally distributed.

Finally, after summarizing the model, I get the mean value of four parameters. $\mu_\alpha = -0.8733, \mu_\beta = 1.742, \sigma_\alpha = 0.1293, \sigma_\beta = 0.21$. Firstly, a negative value of μ_α suggests that, on average, subjects in countries with a negative EZK test are less likely to be infected with the K9C9 virus. Then, a positive value of μ_β suggests that a positive EZK test is associated with an increased likelihood of being infected with the virus. Lastly, small values of $\sigma_\alpha, \sigma_\beta$ indicate that the virus is relatively similar across the countries.

Furthermore, I can calculate the probability. Firstly, if the EZK test is negative, the log-odds of infection is -0.8733, so the probability of infection is 29.46%, so the probability of not infection is 70.54%. Secondly, if the EZK test is positive, the log-odds of infection is 0.8687, so the probability of infection is 70.45%.

In conclusion, from the probability I calculate above, I believe that the diagnostic ability of EZK is not bad, since it has pretty high probability, which is around 70.5%, that give the same results with the highly accurate diagnostic test. Considering its low price, I think EZK can be used as one of the tests to diagnose the infection.

Appendix

```
library(coda)
library(rjags)
```

```
## Linked to JAGS 4.3.1
```

```
## Loaded modules: basemod,bugs
```

```
flu <- read.table("flu.txt", header=T)
```

```
set.seed(99)

# data
y=flu$Infected
nc=10
y=matrix(y,ncol=nc)
ni=nrow(y)
x=flu$EZK
x=matrix(x,ncol=nc)

# Setup the input data list for JAGS:
mydata = list(ni=ni,nc=nc,y=y,x=x)

# Setup parameter initialization for JAGS:
myinit = list(mu_alpha = 0, mu_beta = 0, sigma_alpha = 1, sigma_beta = 1)

# Setup MCMC options for JAGS:
niters=10000  # **total** number of iterations, **including** burn-in
nburns=2500
nadapt=2500
nchains=2

# Specify JAGS model:
mod = "model {
  # likelihood
  for (c in 1:nc) {
    alpha[c] ~ dnorm(mu_alpha, 1/(sigma_alpha*sigma_alpha))
    beta[c] ~ dnorm(mu_beta, 1/(sigma_beta*sigma_beta))

    for (i in 1:ni) {
      logit(theta[i,c]) <- alpha[c] + beta[c]*x[i,c]
      y[i,c] ~ dbern(theta[i,c])
    }
  }

  # priors
  mu_alpha ~ dnorm(0, 1/9)
  mu_beta ~ dnorm(0, 1/9)
  sigma_alpha ~ dunif(0, 3)
  sigma_beta ~ dunif(0, 3)
}"
```

```r
# Now let's setup the model:
fit=jags.model(textConnection(mod),
               data=mydata, inits=myinit, n.chains=nchains, n.adapt=nadapt)
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 1000
##    Unobserved stochastic nodes: 24
##    Total graph size: 2095
##
## Initializing model
```

```r
# And run it:
fit.samples=coda.samples(fit,c("mu_alpha", "mu_beta", "sigma_alpha", "sigma_beta"),n.ite
summary(fit.samples)
```

```
##
## Iterations = 2501:12500
## Thinning interval = 1
## Number of chains = 2
## Sample size per chain = 10000
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##                Mean     SD  Naive SE Time-series SE
## mu_alpha    -0.8714 0.1107 0.0007827       0.004809
## mu_beta      1.7412 0.1642 0.0011612       0.006779
## sigma_alpha  0.1315 0.1030 0.0007286       0.004878
## sigma_beta   0.2257 0.1633 0.0011546       0.007944
##
## 2. Quantiles for each variable:
##
##                 2.5%      25%     50%     75%   97.5%
## mu_alpha    -1.090864 -0.94561 -0.8693 -0.7983 -0.6590
## mu_beta      1.417330  1.63195  1.7406  1.8452  2.0663
## sigma_alpha  0.003568  0.05031  0.1132  0.1881  0.3830
## sigma_beta   0.006997  0.10328  0.1991  0.3156  0.6133
```

```
fit.samples2=coda.samples(fit,c("mu_alpha", "mu_beta"),n.iter=niters)
plot(fit.samples2)
```