

Sensitivity/Specificity Analyses - Canine Leishmaniosis

Marie Ozanne

March 3, 2020

Exploratory Analyses

| | | | |
|----|----------|----------|----------|
| ## | | DPP | |
| ## | PCR | Negative | Positive |
| ## | Negative | 743 | 16 |
| ## | Positive | 1 | 11 |

Models

Angela's paper (Toepp et al., 2019, <https://doi.org/10.1371/journal.pntd.0007058>) uses logistic regression, with age, sex, and variables that have to do with diagnostic tests as explanatory variables. They are something like this:

Model A 1: $\text{logit}(\pi_k) = \beta_0 + \beta_1 \text{Age}_k + \beta_2 \text{Sex}_k + \beta_3 Y_k$, where Y_k is diagnostically positive (as defined in Model 1 below), but for the mom and π_k is the probability of disease for individual k

Model A 2: $\text{logit}(\pi_k) = \beta_0 + \beta_1 \text{Age}_k + \beta_2 \text{Sex}_k + \beta_3 T_{1k} + \beta_4 T_{2k}$, where T_{jk} is the result for Test j (as defined in Model 1 below), but for the mom and π_k is the probability of disease for individual k

Note, these models were fit with a log link function, presumably so that relative risks could be recovered?

We plan to evaluate similar models for our data and to then incorporate sensitivity and specificity of the tests into these models. Then we will compare the model performance to that of other methods. Hopefully we will see an improvement/some details that we miss when we do not include the sensitivity and specificity for the tests.

In all these models, we will assume that the observations are independent.

Separate Diagnostic Tests in Data Model:

Data Model

$$T_{1k}|D_k \sim \text{Bernoulli}(\pi_{1k})$$

where $\pi_{1k} = P(T_{1k} = 1|D_k) = D_k \times \underbrace{P(T_{1k} = 1|D_k = 1)}_{\text{sensitivity}} + (1 - D_k) \times \underbrace{(1 - P(T_{1k} = 0|D_k = 0))}_{1 - \text{specificity}}.$

$$T_{2k}|D_k \sim \text{Bernoulli}(\pi_{2k})$$

where $\pi_{2k} = P(T_{2k} = 1|D_k) = D_k \times \underbrace{P(T_{2k} = 1|D_k = 1)}_{\text{sensitivity}} + (1 - D_k) \times \underbrace{(1 - P(T_{2k} = 0|D_k = 0))}_{1 - \text{specificity}}.$

Process Model

$$D_k \sim \text{Bernoulli}(\delta_k)$$

where $\delta_k = \text{logit}^{-1}(\text{logit}(\rho) + \mathbf{x}_k^T \boldsymbol{\beta} + \epsilon_k)$, and ϵ_k is a random effect for individual k .

Prior Model

Prevalence

$$\text{logit}(\rho) \sim \text{Normal}(\mu_\rho, \tau^2)$$

where $\mu_\rho = \text{logit}^{-1}(\rho^*)$ and $\tau^2 = 100$.

Random intercept

$$\epsilon \sim \text{Normal}(0, 5000)$$

OpenBUGS Implementation (a):

- no individual fixed effects
- sens=spec=0.999

```
## ranges of sensitivities and specificities
sens.pcr <- 0.999
sens.dpp <- 0.999
spec.pcr <- 0.999
spec.dpp <- 0.999

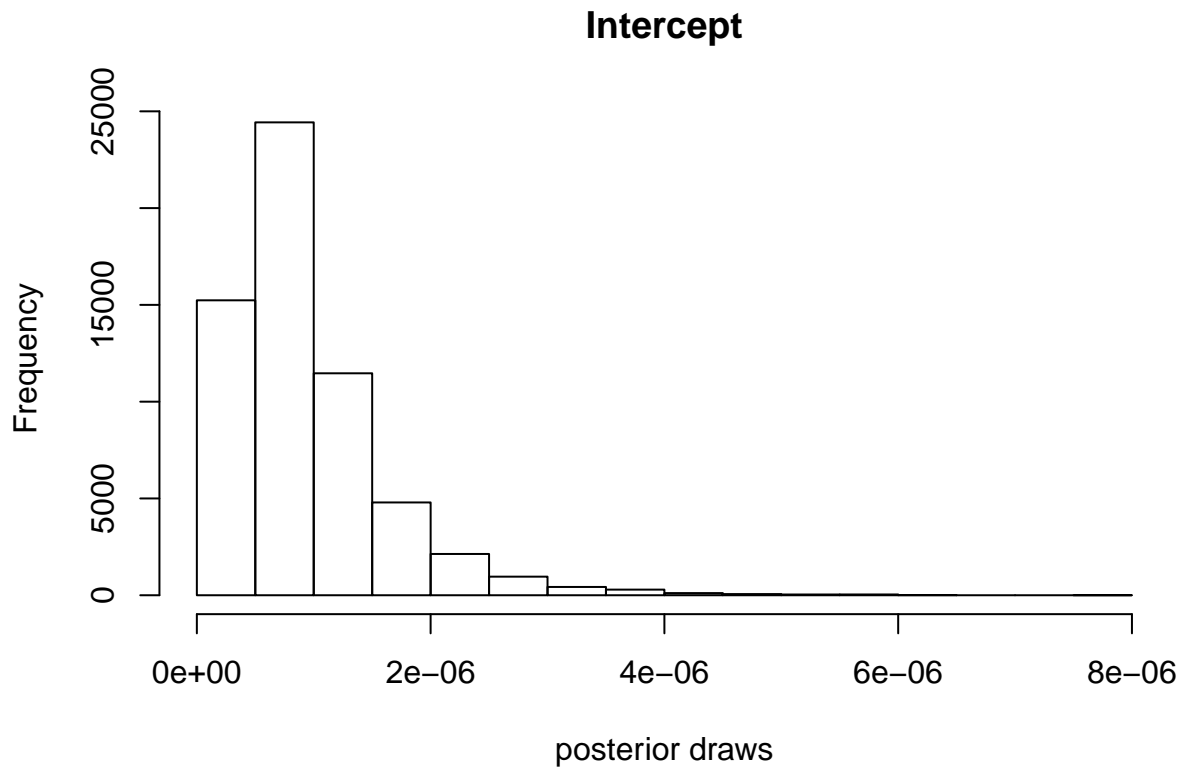
## range of prevalence for visceral leishmaniasis
prev.range <- c(0.05,0.10)
prev <- mean(prev.range)
```

- Mixing seems to be much improved (for those individuals where the trace plots is anything other than 0) over previous model.

Posterior Distributions (a)

```
# par(mfrow=c(2,2))
## Graphical summaries of posterior distributions
# hist(exp(model1a_df$lp1)/(1+exp(model1a_df$lp1)), main="prevalence", xlab="posterior draws")
# abline(v=mean(exp(model1a_df$lp1)/(1+exp(model1a_df$lp1))), lty="dashed", col="red")

hist(exp(model1a_df$b1)/(1+exp(model1a_df$b1)), main="Intercept", xlab="posterior draws")
abline(v=mean(model1a_df$b1), lty="dashed", col="red")
```



```
# hist(model1a_df$b2, main="Sex", xlab="posterior draws")
# abline(v=mean(model1a_df$b2), lty="dashed", col="red")
# hist(model1a_df$b3, main="Age/Sex Interaction", xlab="posterior draws")
# abline(v=mean(model1a_df$b3), lty="dashed", col="red")

summary(exp(model1a_df$b1)/(1+exp(model1a_df$b1)))
```

```
##      Min.   1st Qu.   Median     Mean   3rd Qu.    Max.
## 6.428e-08 4.993e-07 7.676e-07 9.384e-07 1.180e-06 7.656e-06
```

Disease State Classification (a)

```
## Set up storage for model results
pred_df_m1a <- data.frame(obs=1:nind,
                          D=rep(NA,nind), ## average estimate
                          SD=rep(NA,nind),
                          LB=rep(NA,nind), ## 2.5th percentile
                          UB=rep(NA,nind), ## 97.5th percentile
                          model_assignment=rep(NA,nind),
                          Clinical_status=ss_data2$ClinicalStatus,
                          Diagnostic_status=ss_data2$Diagnostically_positive)

## Calculate probabilities of compartment membership for each posterior draw
```

```

pred_df_m1a$D <- apply(model1a_df[,grep("D", names(model1a_df))], 2, mean)
pred_df_m1a$SD <- apply(model1a_df[,grep("D", names(model1a_df))], 2, sd)
pred_df_m1a$LB <- apply(model1a_df[,grep("D", names(model1a_df))], 2,
  quantile, probs=0.025)
pred_df_m1a$UB <- apply(model1a_df[,grep("D", names(model1a_df))], 2,
  quantile, probs=0.975)

summary(pred_df_m1a)

```

```

##      obs          D          SD          LB
## Min.   : 1.0    Min.   :0.00000    Min.   :0.000000    Min.   :0.00000
## 1st Qu.:193.5    1st Qu.:0.00000    1st Qu.:0.000000    1st Qu.:0.00000
## Median :386.0    Median :0.00000    Median :0.000000    Median :0.00000
## Mean   :386.0    Mean   :0.01784    Mean   :0.008182    Mean   :0.01427
## 3rd Qu.:578.5    3rd Qu.:0.00000    3rd Qu.:0.000000    3rd Qu.:0.00000
## Max.   :771.0    Max.   :1.00000    Max.   :0.384551    Max.   :1.00000
##      UB      model_assignment Clinical_status Diagnostic_status
## Min.   :0.00000    Mode:logical      A: 0           Negative:743
## 1st Qu.:0.00000    NA's:771          N:736          Positive: 28
## Median :0.00000          S: 35
## Mean   :0.03632
## 3rd Qu.:0.00000
## Max.   :1.00000

```

```

## Apply a cut off of point estimate of 0.5; if pi.D > 0.5, classify as S (symptomatic), otherwise as N
## Summarize in a table (clinical status versus diagnostic status)
table(pred_df_m1a[pred_df_m1a$D > 0.5,]$Clinical_status,
  pred_df_m1a[pred_df_m1a$D > 0.5,]$Diagnostic_status)

```

```

##
##      Negative Positive
## A          0          0
## N          0          0
## S          0         11

```

```

## Print summary table of clinical status versus diagnostic status from the original data
table(pred_df_m1a$Clinical_status, pred_df_m1a$Diagnostic_status)

```

```

##
##      Negative Positive
## A          0          0
## N         728          8
## S          15         20

```

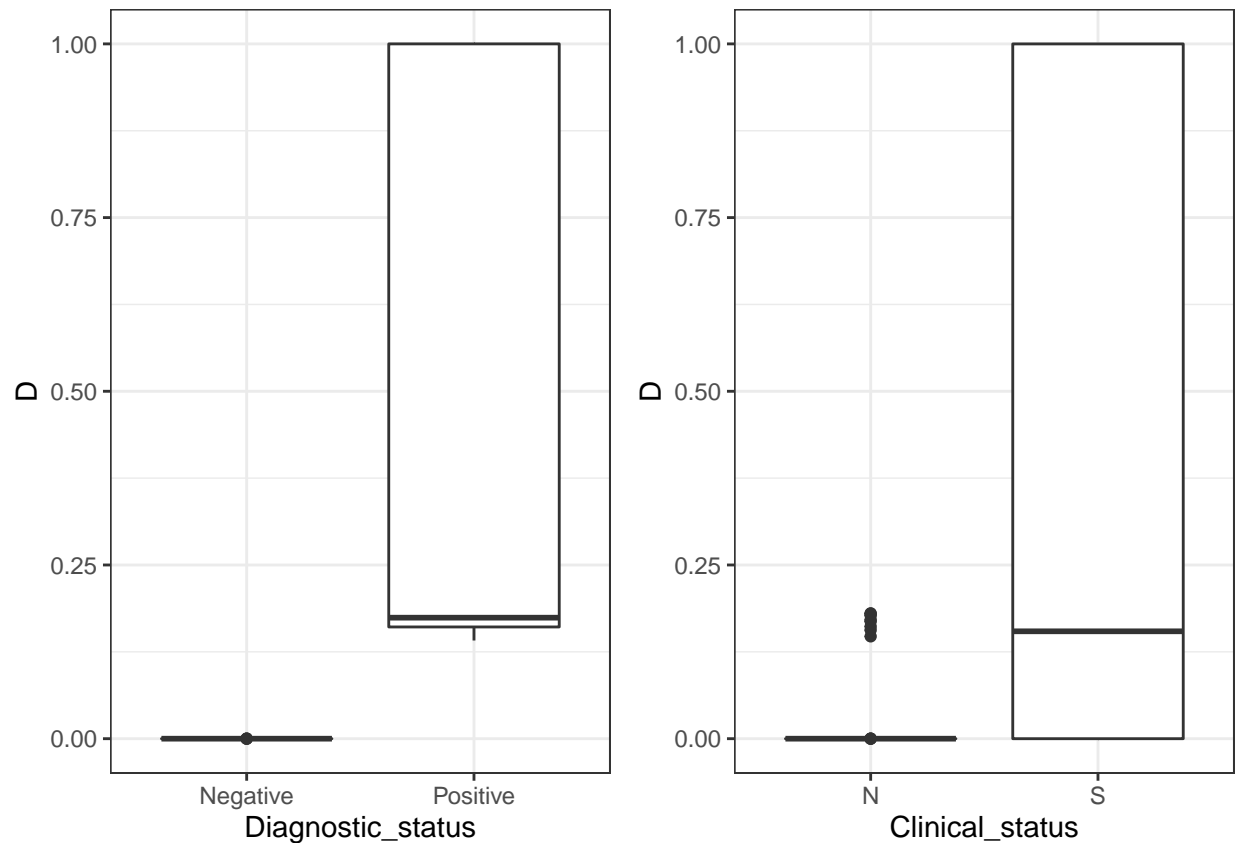
```

## boxplots
p1 <- (ggplot(data=pred_df_m1a, aes(x=Diagnostic_status, y=D))
  + geom_boxplot()
  + theme_bw())
p2 <- (ggplot(data=pred_df_m1a, aes(x=Clinical_status, y=D))
  + geom_boxplot())

```

```
+ theme_bw())

ggarrange(p1,p2, nrow=1)
```



OpenBUGS Implementation (b)

Same as (a), but without fixing the sensitivities and specificities to be 0.999.

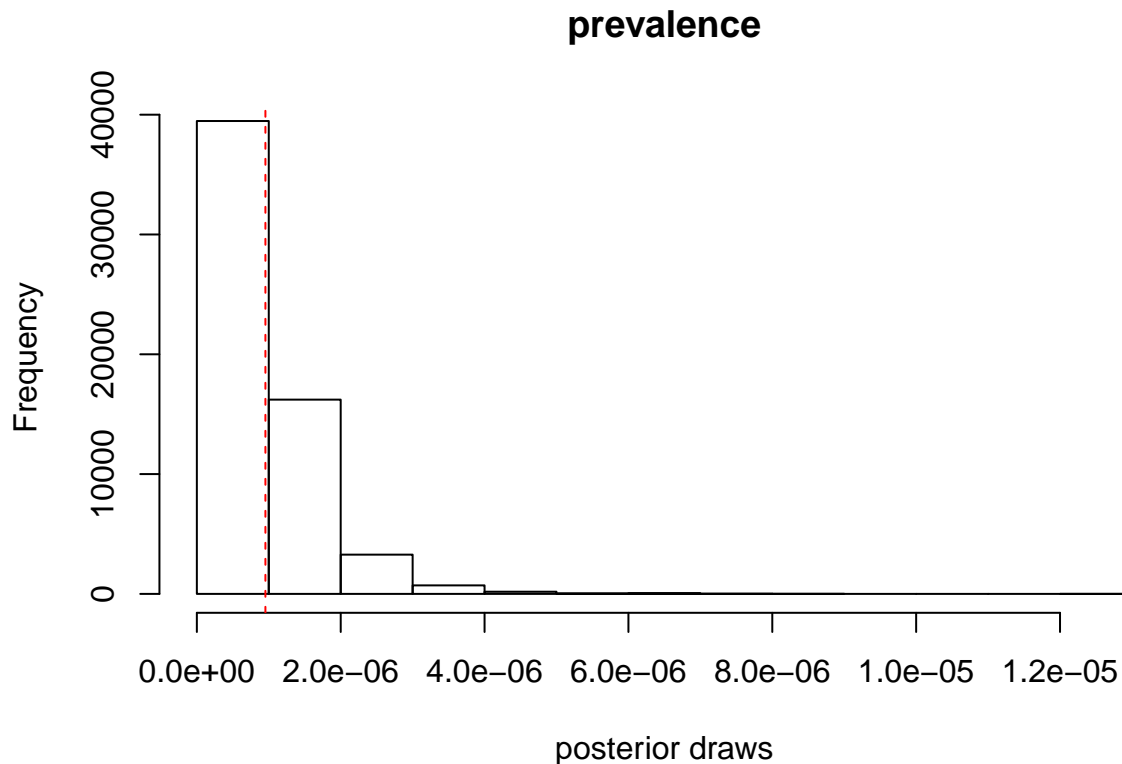
```
## ranges of sensitivities and specificities
sens.pcr.range <- c(0.839, 0.990)
sens.dpp.range <- c(0.832, 0.930)
spec.pcr.range <- c(0.871, 0.970)
spec.dpp.range <- c(0.682, 0.951)

sens.pcr <- mean(sens.pcr.range)
sens.dpp <- mean(sens.dpp.range)
spec.pcr <- mean(spec.pcr.range)
spec.dpp <- mean(spec.dpp.range)

## range of prevalence for visceral leishmaniasis
prev.range <- c(0.05,0.10)
prev <- mean(prev.range)
```

Posterior Distributions (b)

```
# par(mfrow=c(2,2))
## Graphical summaries of posterior distributions
hist(exp(model1b_df$b1)/(1+exp(model1b_df$b1)), main="prevalence", xlab="posterior draws")
abline(v=mean(exp(model1b_df$b1)/(1+exp(model1b_df$b1))), lty="dashed", col="red")
```



```
# hist(model1b_df$b1, main="Age", xlab="posterior draws")
# abline(v=mean(model1b_df$b1), lty="dashed", col="red")
# hist(model1b_df$b2, main="Sex", xlab="posterior draws")
# abline(v=mean(model1b_df$b2), lty="dashed", col="red")
# hist(model1b_df$b3, main="Age/Sex Interaction", xlab="posterior draws")
# abline(v=mean(model1b_df$b3), lty="dashed", col="red")
```

Disease State Classification (b)

```
## Set up storage for model results
pred_df_m1b <- data.frame(obs=1:nind,
                           D=rep(NA,nind), ## average estimate
                           SD=rep(NA,nind),
                           LB=rep(NA,nind), ## 2.5th percentile
                           UB=rep(NA,nind), ## 97.5th percentile)
```

```

        model_assignment=rep(NA,nind),
        Clinical_status=ss_data2$ClinicalStatus,
        Diagnostic_status=ss_data2$Diagnostically_positive)

## Calculate probabilities of compartment membership for each posterior draw
pred_df_m1b$D <- apply(model1b_df[,grep("D", names(model1b_df))], 2, mean)
pred_df_m1b$SD <- apply(model1b_df[,grep("D", names(model1b_df))], 2, sd)
pred_df_m1b$LB <- apply(model1b_df[,grep("D", names(model1b_df))], 2,
                        quantile, probs=0.025)
pred_df_m1b$UB <- apply(model1b_df[,grep("D", names(model1b_df))], 2,
                        quantile, probs=0.975)

summary(pred_df_m1b)

```

```

##      obs      D      SD      LB
## Min.   : 1.0   Min.   :0.000650   Min.   :0.02549   Min.   :0
## 1st Qu.:193.5 1st Qu.:0.002050   1st Qu.:0.04523   1st Qu.:0
## Median :386.0 Median :0.002550   Median :0.05043   Median :0
## Mean   :386.0 Mean   :0.017507   Mean   :0.05841   Mean   :0
## 3rd Qu.:578.5 3rd Qu.:0.003208   3rd Qu.:0.05655   3rd Qu.:0
## Max.   :771.0 Max.   :0.931867   Max.   :0.42445   Max.   :0
##      UB      model_assignment Clinical_status Diagnostic_status
## Min.   :0.00000   Mode:logical      A: 0           Negative:743
## 1st Qu.:0.00000   NA's:771          N:736          Positive: 28
## Median :0.00000           S: 35
## Mean   :0.03632
## 3rd Qu.:0.00000
## Max.   :1.00000

```

```

## Apply a cut off of point estimate of 0.5; if pi.D > 0.5, classify as S (symptomatic), otherwise as N
## Summarize in a table (clinical status versus diagnostic status)
table(pred_df_m1b[pred_df_m1b$D > 0.5,]$Clinical_status,
      pred_df_m1b[pred_df_m1b$D > 0.5,]$Diagnostic_status)

```

```

##
##      Negative Positive
## A      0      0
## N      0      0
## S      0     11

```

```

## Print summary table of clinical status versus diagnostic status from the original data
table(pred_df_m1b$Clinical_status, pred_df_m1b$Diagnostic_status)

```

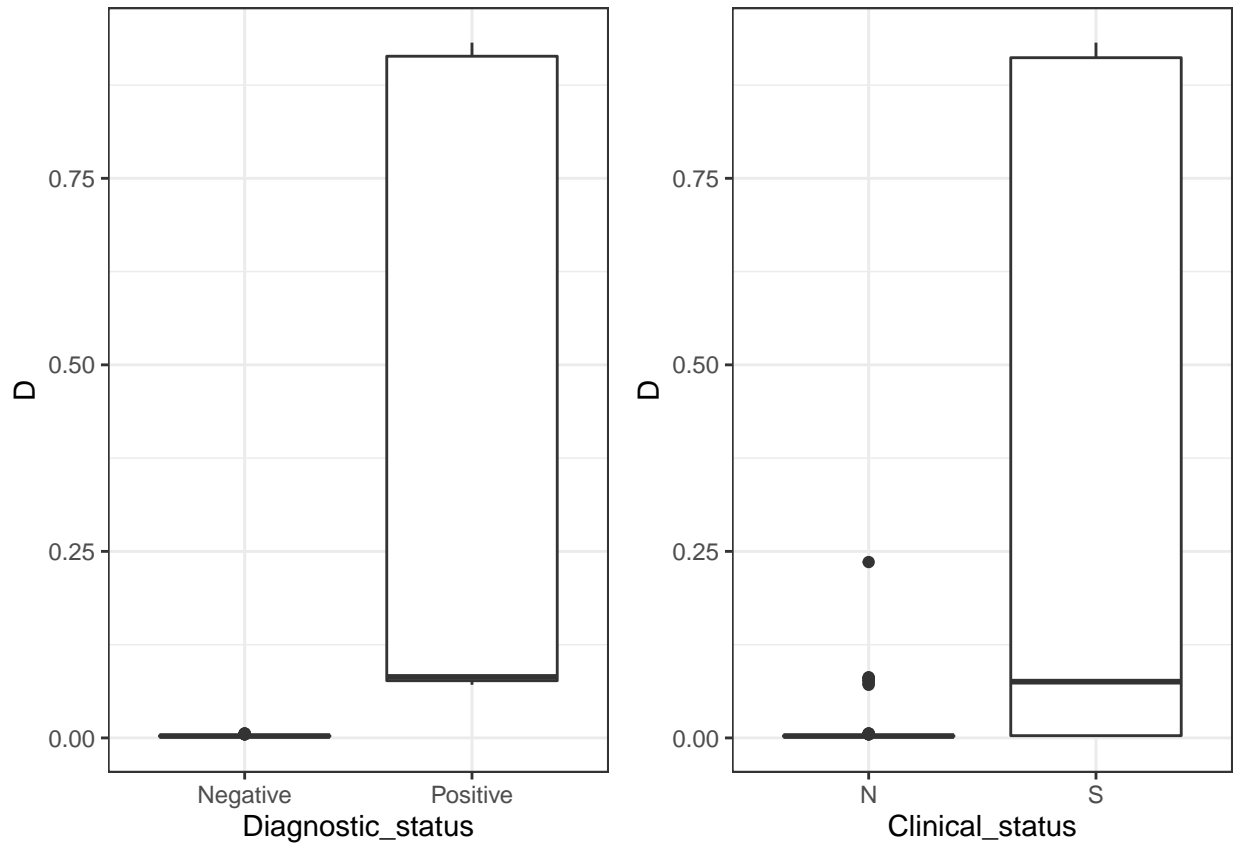
```

##
##      Negative Positive
## A      0      0
## N     728      8
## S     15     20

```

```
## boxplots
p1 <- (ggplot(data=pred_df_m1b, aes(x=Diagnostic_status, y=D))
      + geom_boxplot()
      + theme_bw())
p2 <- (ggplot(data=pred_df_m1b, aes(x=Clinical_status, y=D))
      + geom_boxplot()
      + theme_bw())

ggarrange(p1,p2, nrow=1)
```



- On the plus side, the mixing looks much better with this model.

OpenBUGS implementation (c)

Linear predictors

The regression parameters are $\beta = (\beta_{age}, \beta_{sex}, \beta_{age*sex})^T$;

$$\beta \sim \text{Normal}(\mu_{\beta}, \Sigma_{\beta})$$

where $\mu_{\beta} = \mathbf{0}$ and $\Sigma_{\beta} = \mathbf{I}$ in our code.

```
## ranges of sensitivities and specificities
sens.pcr.range <- 0.999
```



```

sens.dpp.range <- 0.999
spec.pcr.range <- 0.999
spec.dpp.range <- 0.999

## range of prevalence for visceral leishmaniasis
prev.range <- c(0.05,0.10)
prev <- mean(prev.range)

```

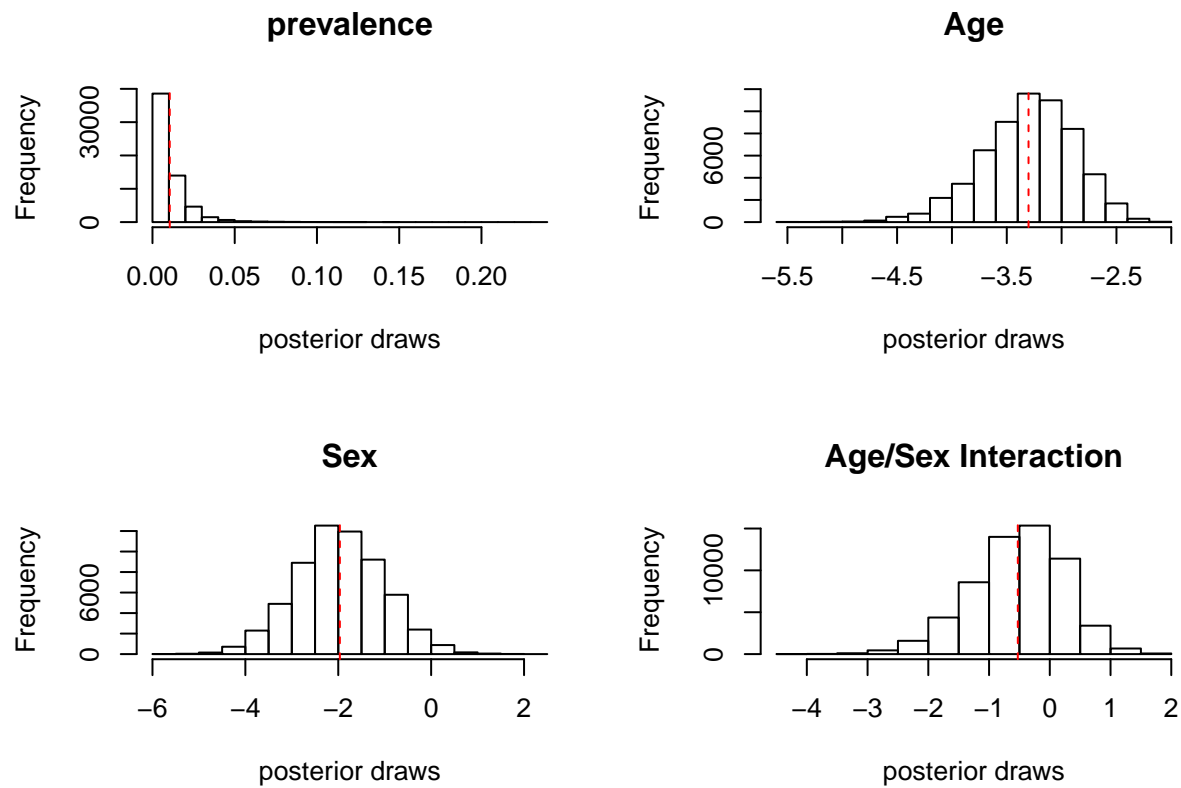
Posterior Distributions (c)

```

par(mfrow=c(2,2))
## Graphical summaries of posterior distributions
hist(exp(model1c_df$b1)/(1+exp(model1c_df$b1)), main="prevalence", xlab="posterior draws")
abline(v=mean(exp(model1c_df$b1)/(1+exp(model1c_df$b1))), lty="dashed", col="red")

hist(model1c_df$b2, main="Age", xlab="posterior draws")
abline(v=mean(model1c_df$b2), lty="dashed", col="red")
hist(model1c_df$b3, main="Sex", xlab="posterior draws")
abline(v=mean(model1c_df$b3), lty="dashed", col="red")
hist(model1c_df$b4, main="Age/Sex Interaction", xlab="posterior draws")
abline(v=mean(model1c_df$b4), lty="dashed", col="red")

```



Disease State Classification (c)

```
## Set up storage for model results
pred_df_m1c <- data.frame(obs=1:nind,
                          D=rep(NA,nind), ## average estimate
                          SD=rep(NA,nind),
                          LB=rep(NA,nind), ## 2.5th percentile
                          UB=rep(NA,nind), ## 97.5th percentile
                          model_assignment=rep(NA,nind),
                          Clinical_status=ss_data2$ClinicalStatus,
                          Diagnostic_status=ss_data2$Diagnostically_positive)

## Calculate probabilities of compartment membership for each posterior draw
pred_df_m1c$D <- apply(model1c_df[,grep("D", names(model1c_df))], 2, mean)
pred_df_m1c$SD <- apply(model1c_df[,grep("D", names(model1c_df))], 2, sd)
pred_df_m1c$LB <- apply(model1c_df[,grep("D", names(model1c_df))], 2,
                        quantile, probs=0.025)
pred_df_m1c$UB <- apply(model1c_df[,grep("D", names(model1c_df))], 2,
                        quantile, probs=0.975)

summary(pred_df_m1c)
```

```
##      obs      D      SD      LB
## Min.   : 1.0   Min.   :0.0000000   Min.   :0.000000   Min.   :0
## 1st Qu.:193.5 1st Qu.:0.0000167   1st Qu.:0.004083   1st Qu.:0
## Median :386.0 Median :0.0001000   Median :0.010000   Median :0
## Mean   :386.0 Mean   :0.0059332   Mean   :0.027964   Mean   :0
## 3rd Qu.:578.5 3rd Qu.:0.0004167   3rd Qu.:0.020408   3rd Qu.:0
## Max.   :771.0 Max.   :0.5866833   Max.   :0.498793   Max.   :0
##      UB      model_assignment Clinical_status Diagnostic_status
## Min.   :0.00000   Mode:logical      A: 0           Negative:743
## 1st Qu.:0.00000   NA's:771          N:736          Positive: 28
## Median :0.00000           S: 35
## Mean   :0.01946
## 3rd Qu.:0.00000
## Max.   :1.00000
```

```
## Apply a cut off of point estimate of 0.5; if pi.D > 0.5, classify as S (symptomatic), otherwise as N
## Summarize in a table (clinical status versus diagnostic status)
table(pred_df_m1c[pred_df_m1c$D > 0.5,]$Clinical_status,
      pred_df_m1c[pred_df_m1c$D > 0.5,]$Diagnostic_status)
```

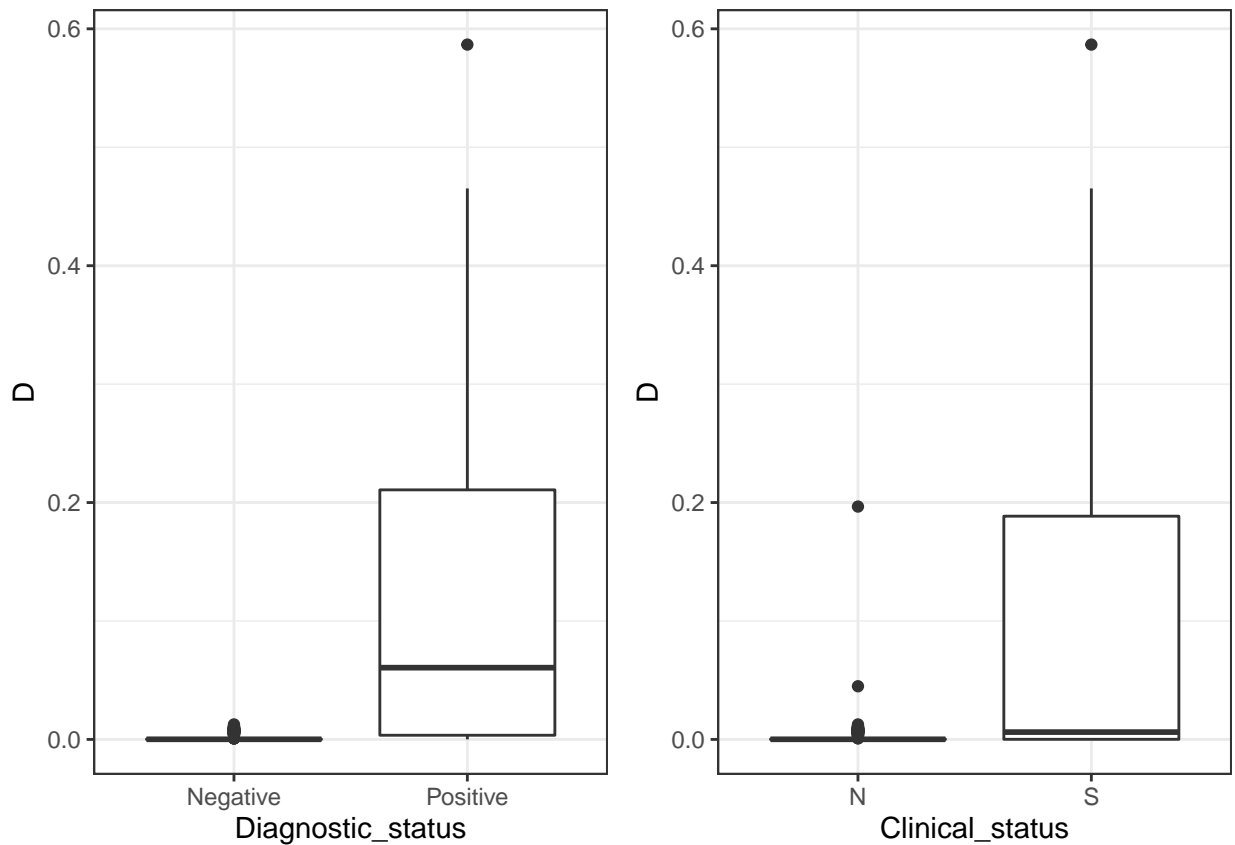
```
##
##      Negative Positive
## A      0      0
## N      0      0
## S      0      1
```

```
## Print summary table of clinical status versus diagnostic status from the original data
table(pred_df_m1c$Clinical_status, pred_df_m1c$Diagnostic_status)
```

```
##
##      Negative Positive
##  A      0      0
##  N     728     8
##  S      15     20
```

```
## boxplots
p1 <- (ggplot(data=pred_df_m1c, aes(x=Diagnostic_status, y=D))
      + geom_boxplot()
      + theme_bw())
p2 <- (ggplot(data=pred_df_m1c, aes(x=Clinical_status, y=D))
      + geom_boxplot()
      + theme_bw())

ggarrange(p1,p2, nrow=1)
```



OpenBUGS implementation (d)

Linear predictors

The regression parameters are $\beta = (\beta_{age}, \beta_{sex}, \beta_{age*sex})^T$;

$$\beta \sim Normal(\mu_{\beta}, \Sigma_{\beta})$$

where $\mu_{\beta} = \mathbf{0}$ and $\Sigma_{\beta} = \mathbf{I}$ in our code.

```

## ranges of sensitivities and specificities
sens.pcr.range <- c(0.839, 0.990)
sens.dpp.range <- c(0.832, 0.930)
spec.pcr.range <- c(0.871, 0.970)
spec.dpp.range <- c(0.682, 0.951)

sens.pcr <- mean(sens.pcr.range)
sens.dpp <- mean(sens.dpp.range)
spec.pcr <- mean(spec.pcr.range)
spec.dpp <- mean(spec.dpp.range)

## range of prevalence for visceral leishmaniasis
prev.range <- c(0.05, 0.10)
prev <- mean(prev.range)

```

OpenBUGS Model 1 Implementation

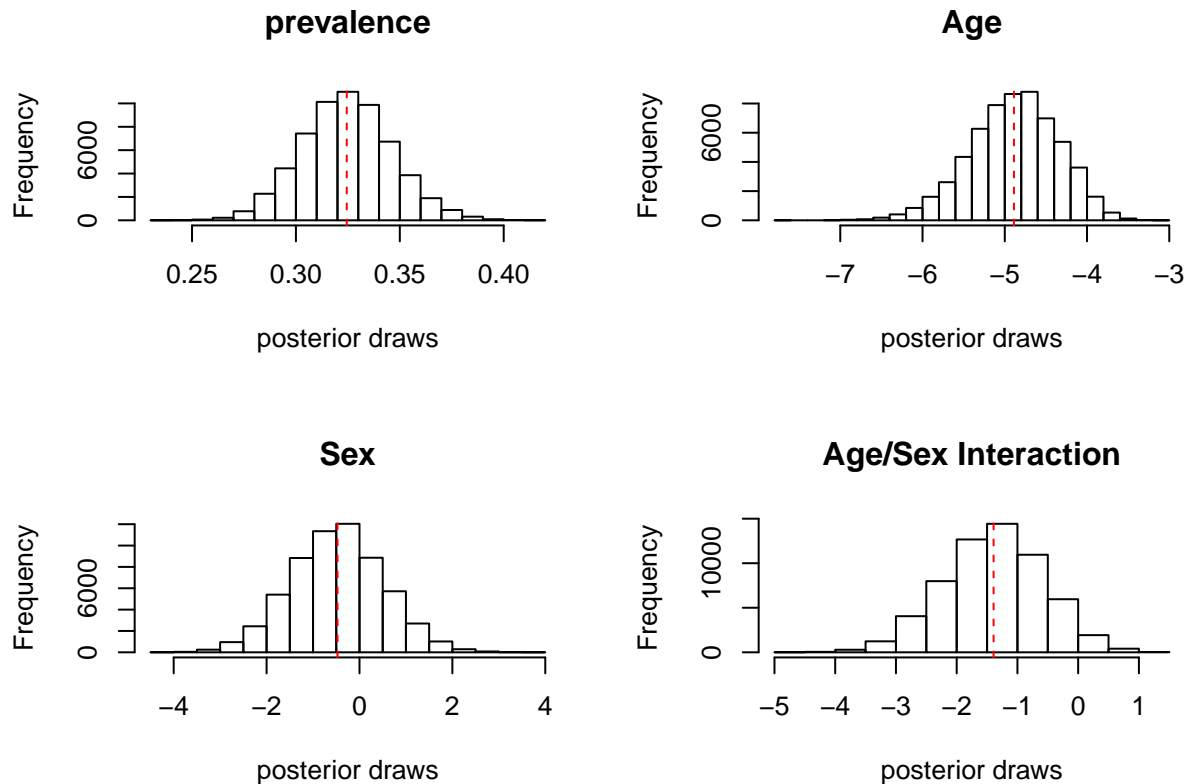
OpenBUGS Model 1 Posterior Distributions

```

par(mfrow=c(2,2))
## Graphical summaries of posterior distributions
hist(exp(model1_df$lpi)/(1+exp(model1_df$lpi)), main="prevalence", xlab="posterior draws")
abline(v=mean(exp(model1_df$lpi)/(1+exp(model1_df$lpi))), lty="dashed", col="red")

hist(model1_df$b1, main="Age", xlab="posterior draws")
abline(v=mean(model1_df$b1), lty="dashed", col="red")
hist(model1_df$b2, main="Sex", xlab="posterior draws")
abline(v=mean(model1_df$b2), lty="dashed", col="red")
hist(model1_df$b3, main="Age/Sex Interaction", xlab="posterior draws")
abline(v=mean(model1_df$b3), lty="dashed", col="red")

```



OpenBUGS Model 1 Disease State Classification

```
## Set up storage for model results
pred_df_m1 <- data.frame(obs=1:nind,
                        D=rep(NA,nind), ## average estimate
                        SD=rep(NA,nind),
                        LB=rep(NA,nind), ## 2.5th percentile
                        UB=rep(NA,nind), ## 97.5th percentile
                        model_assignment=rep(NA,nind),
                        Clinical_status=ss_data2$ClinicalStatus,
                        Diagnostic_status=ss_data2$Diagnostically_positive)

## Calculate probabilities of compartment membership for each posterior draw
pred_df_m1$D <- apply(model1_df[,grep("D", names(model1_df))], 2, mean)
pred_df_m1$SD <- apply(model1_df[,grep("D", names(model1_df))], 2, sd)
pred_df_m1$LB <- apply(model1_df[,grep("D", names(model1_df))], 2,
                       quantile, probs=0.025)
pred_df_m1$UB <- apply(model1_df[,grep("D", names(model1_df))], 2,
                       quantile, probs=0.975)

summary(pred_df_m1)
```

| ## | obs | D | SD | LB | UB |
|----|-----|---|----|----|----|
|----|-----|---|----|----|----|

```
## Min.      : 1.0      Min.      :0.000e+00      Min.      :0.000000      Min.      :0
## 1st Qu.:193.5      1st Qu.:0.000e+00      1st Qu.:0.000000      1st Qu.:0
## Median :386.0      Median :0.000e+00      Median :0.000000      Median :0
## Mean    :386.0      Mean    :1.358e-03      Mean    :0.010641      Mean    :0
## 3rd Qu.:578.5      3rd Qu.:6.667e-05      3rd Qu.:0.008165      3rd Qu.:0
## Max.    :771.0      Max.    :2.726e-01      Max.    :0.445283      Max.    :0
##          UB          model_assignment Clinical_status Diagnostic_status
## Min.      :0.000000      Mode:logical      A: 0              Negative:743
## 1st Qu.:0.000000      NA's:771          N:736             Positive: 28
## Median :0.000000                                S: 35
## Mean    :0.009079
## 3rd Qu.:0.000000
## Max.    :1.000000
```

```
## Apply a cut off of point estimate of 0.5; if pi.D > 0.5, classify as S (symptomatic), otherwise as N
## Summarize in a table (clinical status versus diagnostic status)
```

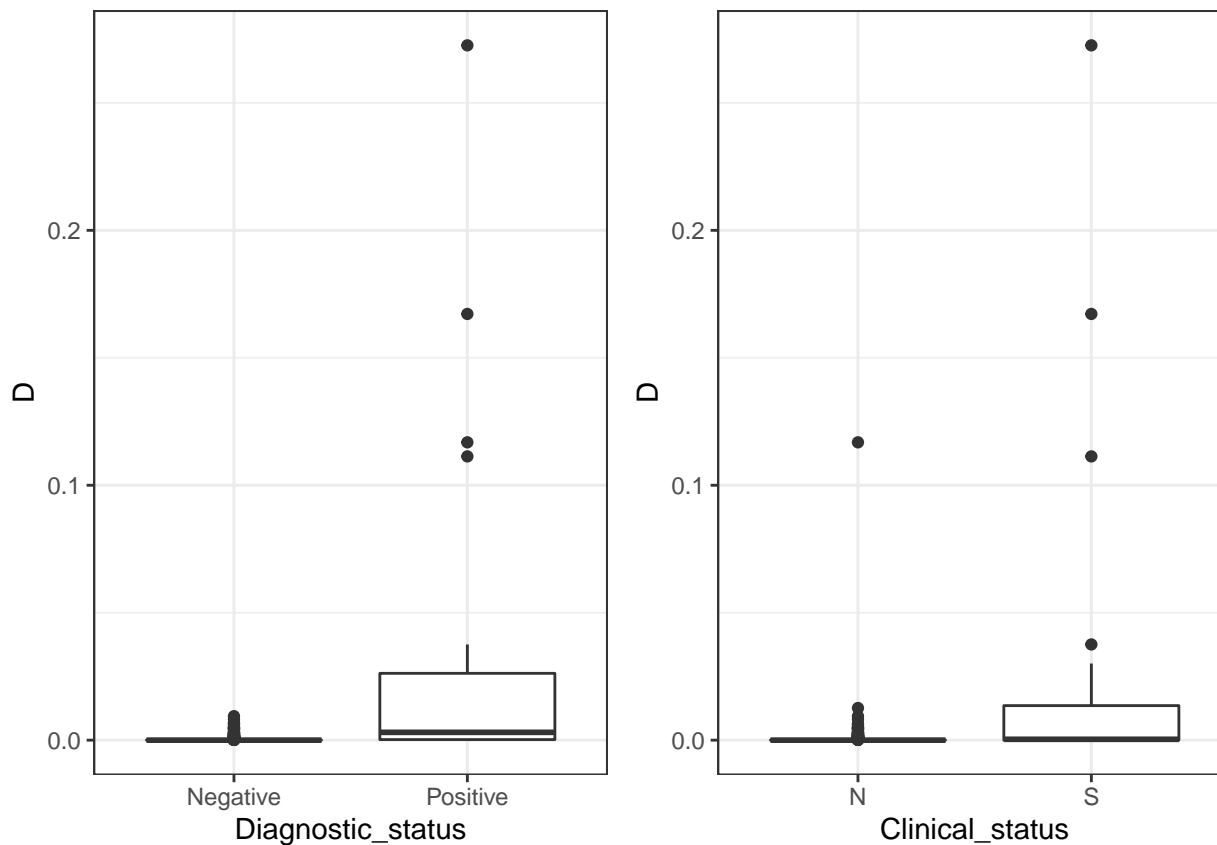
```
table(pred_df_m1[pred_df_m1$D > 0.5,]$Clinical_status,
      pred_df_m1[pred_df_m1$D > 0.5,]$Diagnostic_status)
```

```
##
##      Negative Positive
## A         0         0
## N         0         0
## S         0         0
```

```
## Print summary table of clinical status versus diagnostic status from the original data
table(pred_df_m1$Clinical_status, pred_df_m1$Diagnostic_status)
```

```
##
##      Negative Positive
## A         0         0
## N        728         8
## S         15        20
```

```
## boxplots
p1 <- (ggplot(data=pred_df_m1, aes(x=Diagnostic_status, y=D))
      + geom_boxplot()
      + theme_bw())
p2 <- (ggplot(data=pred_df_m1, aes(x=Clinical_status, y=D))
      + geom_boxplot()
      + theme_bw())
ggarrange(p1,p2, nrow=1)
```



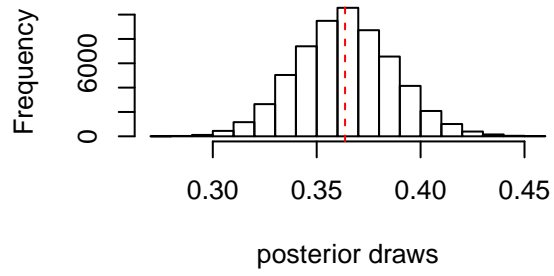
OpenBUGS Model 1 Implementation - Informative Prior Prevalence

OpenBUGS Model 1 Posterior Distributions - Informative Prior Prevalence

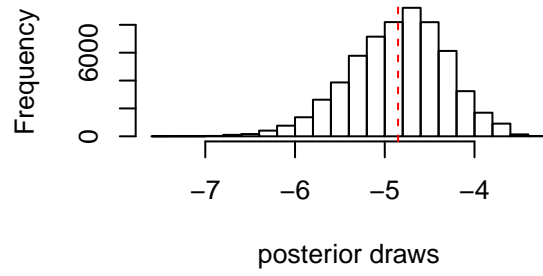
```
par(mfrow=c(2,2))
## Graphical summaries of posterior distributions
hist(exp(model1b_df$lp1)/(1+exp(model1b_df$lp1)), main="prevalence", xlab="posterior draws")
abline(v=mean(exp(model1b_df$lp1)/(1+exp(model1b_df$lp1))), lty="dashed", col="red")

hist(model1b_df$b1, main="Age", xlab="posterior draws")
abline(v=mean(model1b_df$b1), lty="dashed", col="red")
hist(model1b_df$b2, main="Sex", xlab="posterior draws")
abline(v=mean(model1b_df$b2), lty="dashed", col="red")
hist(model1b_df$b3, main="Age/Sex Interaction", xlab="posterior draws")
abline(v=mean(model1b_df$b3), lty="dashed", col="red")
```

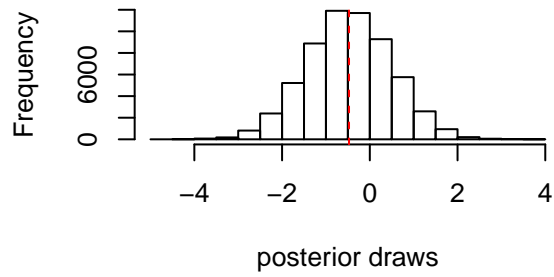
prevalence



Age



Sex



Age/Sex Interaction

