GLMS: Regression for Binary and Count Data

Dichotomous Predicted Variable

- This section focus on dichotomous predicted variables: such as whether a basketball player will get a hit or if a bird will be located in a spatial grid.

- Traditionally, these types of methods are generally implemented with logistic regression.

- The model can be written as:

where the logistic function is $logistic(x) = 1/(1 + exp(-x))$.

- *Q:* What are the three components of a GLM and what are they in this specific setting?
  1. Sampling Model:

```
2. Linear Combination of Predictors:
```

```
3. (inverse) Link function:
```

- Q: To fit this in a Bayesian framework, we need to specify priors. What parameters require priors and what distributions would be reasonable?

We will explore the Swiss birds dataset for this lab to construct a logistic regression model for presence of the Willow Tit.

```
swiss.birds <- read.csv('http://www.math.montana.edu/ahoegh/teaching/stat491/data/willowtit2013.csv')
kable(head(swiss.birds))
```

siteID	elev	rlength	forest	birds	searchDuration
Q001	450	6.4	3	0	160
Q002	450	5.5	21	0	190
Q003	1050	4.3	32	1	150
Q004	950	4.5	9	0	180
Q005	1150	5.4	35	0	200
Q006	550	3.6	2	0	115

This dataset contains 242 sites and 6 variables:

```
- siteID, a unique identifier for the site, some were not sampled during this period
- elev, mean elevation of the quadrant in meters
- rlength, the length of the route walked by the birdwatcher, in kilometers
- forest, percent forest cover
- birds, binary variable for whether a bird is observed, 1 = yes
- searchDuration, time birdwatcher spent searching the site, in minutes
```

```
model.string <- 'model {
  for (i in 1:Ntotal) {
    y[i] ~ dbern(mu[i])
    mu[i] <- ilogit(beta0 + sum( beta[1:Nx] * x[i,1:Nx] ))
  }

  beta0 ~ dnorm(0,1/5^2)
  for (j in 1:Nx){
    beta[j] ~ dnorm(0, 1/5^2)
  }
}'

# Fit Model
jags.logistic <-  jags.model(textConnection(model.string),
                  data=list(y=swiss.birds$birds,
                            Ntotal=nrow(swiss.birds),
                            Nx = 4,
                            x= swiss.birds[,c('elev','rlength','forest','searchDuration')]),
                  n.chains =2, n.adapt = 5000)
```

JAGS Code

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 242
##    Unobserved stochastic nodes: 5
##    Total graph size: 2434
##
## Initializing model
```

```r
update(jags.logistic, 5000)
samples <- coda.samples(jags.logistic, variable.names = c('beta','beta0'), n.iter = 20000)

summary(samples)
```

```
##
## Iterations = 10001:30000
## Thinning interval = 1
## Number of chains = 2
## Sample size per chain = 20000
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##               Mean        SD  Naive SE Time-series SE
## beta[1]  0.002251 0.0004114 2.057e-06      1.419e-05
## beta[2] -0.061192 0.1555577 7.778e-04      5.477e-03
## beta[3]  0.057721 0.0086547 4.327e-05      2.548e-04
## beta[4]  0.007198 0.0033027 1.651e-05      1.280e-04
## beta0   -7.412010 1.2459131 6.230e-03      6.733e-02
##
## 2. Quantiles for each variable:
##
##                2.5%       25%       50%       75%    97.5%
## beta[1]  0.0014851  0.001968  0.002238  0.002519  0.00310
## beta[2] -0.3708866 -0.165840 -0.060349  0.044501  0.24148
## beta[3]  0.0414779  0.051784  0.057420  0.063499  0.07533
## beta[4]  0.0007751  0.005005  0.007206  0.009372  0.01366
## beta0   -9.9151984 -8.223392 -7.385460 -6.580070 -4.98843
```

Interpretation of Logistic Regression Coefficients

Recall this model can be written as: $logit(\mu) = \beta_0 + \beta_1 x_1 + \dots \beta_p x_p$.

- When x_i increases or decreases by 1 unit, then logit(μ) increases or decreases by β_i.

- The logit function for μ can be expressed as $logit(\mu) = \log(\frac{\mu}{1-\mu})$.

- In this setting, μ is the probability of y = 1, so

$$logit(\mu) = \log(\frac{Pr[y=1]}{1 - Pr[y=1]}) = \log(\frac{Pr[y=1]}{Pr[y=0]})$$

- This ratio: $\frac{Pr[y=1]}{Pr[y=0]}$ is known as the odds, so $logit(\mu)$ is the log-odds of an outcome in favor of 1 rather than 0.

- Suppose the logistic regression for the Swiss birds had the following coefficients: $\beta_0 = -7$, $\beta_{elev} = .002$, and $\beta_{forest} = .06$.

  - Compute the probability for observing a bird in a quadrant with: elevation = 1500 meters and forest cover of 60 %. $1/(1 + exp(-(-7 + .002 * 1500 + .06 * 60))) = 0.4013123$

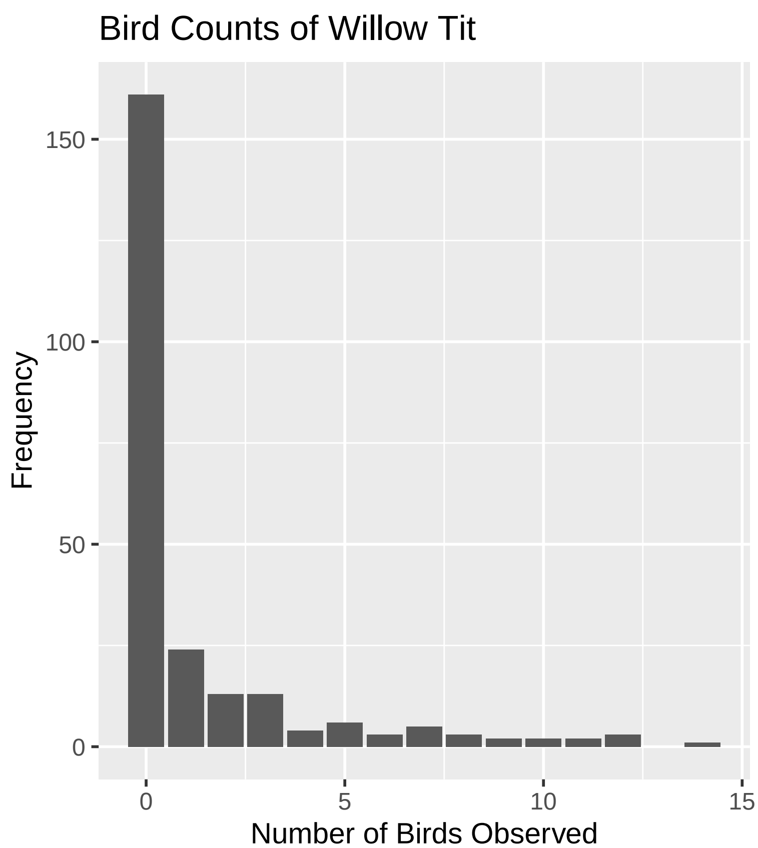
  - How do we interpret the meaning for the coefficient β_{forest}? Each unit increases the log-odds by .06. So for instance, if the forest cover was 70% rather than the 60% above the log-odds of observing a bird would increase by 0.6.
  - The log-odds are different than a probability as in this case (conditional on the elevation) the probability increases to: $1/(1 + exp(-(-6 + .002 * 1500 + .06 * 70))) = 0.7685248$

- Note that an unit increase in the log-odds does not have a unit increase in the probability.

Count Predicted Variable

- Now reconsider the willow tit dataset and consider modeling not just the presence / absence of birds, but directly modeling the number of birds observed in each spatial region.

```
##   siteID elev rlength forest bird.count searchDuration
## 1   Q001  450     6.4      3          0            160
## 2   Q002  450     5.5     21          0            190
## 3   Q003 1050     4.3     32          3            150
## 4   Q004  950     4.5      9          0            180
## 5   Q005 1150     5.4     35          0            200
## 6   Q006  550     3.6      2          0            115
```



1. Sampling Model: In general the Poisson model will be used as a sampling model for count data: -$y|\mu \sim$

2. Linear Combination of Predictors: The same principles apply here as the other GLM settings.

3. Link Function: