

# Bird Keeping Example: Logistic Regression with multiple predictors

In this example, we look at multiple logistic regression including both categorical and continuous predictors.

The birdkeeping data is from Ramsey and Shafer. It is a retrospective case-control study of the relationship between lung cancer (response) and birdkeeping (predictor) but also considering other predictor variables (Sex, Age, etc).

```
library(car)
library(MuMIn)
library(emmeans)
library(ResourceSelection)
BirdData <- read.csv("C:/hess/STAT512/RNotes/MultReg5/MR5_BirdKeeping.csv")
str(BirdData)
```

```
## 'data.frame':    147 obs. of  7 variables:
## $ LC : Factor w/ 2 levels "LungCancer","NoCancer": 1 1 1 1 1 1 1 1 1 1 ...
## $ Sex: Factor w/ 2 levels "Female","Male": 2 2 2 2 2 2 2 2 2 2 ...
## $ SS : Factor w/ 2 levels "High","Low": 2 2 1 2 2 1 1 2 2 1 ...
## $ BK : Factor w/ 2 levels "Bird","NoBird": 1 1 2 1 1 2 1 2 1 2 ...
## $ Age: int   37 41 43 46 49 51 52 53 56 56 ...
## $ YR : int   19 22 19 24 31 24 31 33 33 26 ...
## $ CD : int   12 15 15 15 20 15 20 20 10 25 ...
```

## Summary Table and Chi-square test

The prop.table function is handy for computing proportions. Margin = 1 gives the row proportions. Margin = 2 gives the column proportions. The choice of which you are interested in depends on the research question and how you set up the table!

```
SumTable <- table(BirdData$BK, BirdData$LC)
SumTable
```

```
##
##      LungCancer NoCancer
## Bird           33       34
## NoBird          16       64
```

```
prop.table(SumTable, 1)
```

```
##
##      LungCancer NoCancer
## Bird  0.4925373 0.5074627
## NoBird 0.2000000 0.8000000
```

```
prop.table(SumTable, 2)
```

```
##
##      LungCancer NoCancer
## Bird  0.6734694 0.3469388
## NoBird 0.3265306 0.6530612
```

```
chisq.test(SumTable)
```

```
##  
## Pearson's Chi-squared test with Yates' continuity correction  
##  
## data: SumTable  
## X-squared = 12.756, df = 1, p-value = 0.0003548
```

## Define a new vector of 1s and 0s for logistic regression.

We define Lung Cancer to be the “event” by assigning it a value of 1 (versus a value of 0 for No Cancer).

```
BirdData$resp <- ifelse(BirdData$LC == "LungCancer", 1, 0)  
table(BirdData$LC, BirdData$resp)
```

```
##  
##           0  1  
## LungCancer  0 49  
## NoCancer   98  0
```

## Model Selection

The step() function does AIC based stepwise selection. For this example, we consider only backward selection. (Forward selection would choose the same model; not shown.) dredge() from MuMIn for AIC subsets selection also chooses the same “top” model.

```
FullModel <- glm(resp ~ Sex + SS + BK + Age + YR + CD,  
                 data = BirdData, family=binomial(link="logit"))  
#Backward Elimination  
Model1 <- step(FullModel, direction = "backward", trace = 0)  
summary(Model1)
```

```
##  
## Call:  
## glm(formula = resp ~ BK + YR, family = binomial(link = "logit"),  
##      data = BirdData)  
##  
## Deviance Residuals:  
##      Min       1Q   Median       3Q      Max   
## -1.6093  -0.8644  -0.5283   0.9479   2.0937   
##  
## Coefficients:  
##              Estimate Std. Error z value Pr(>|z|)      
## (Intercept) -1.70460    0.56267  -3.030 0.002450 **    
## BKNoBird    -1.47555    0.39588  -3.727 0.000194 ***   
## YR           0.05825    0.01685   3.458 0.000544 ***   
## ---  
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  
##  
## (Dispersion parameter for binomial family taken to be 1)  
##  
##      Null deviance: 187.14  on 146  degrees of freedom
```

```
## Residual deviance: 158.11 on 144 degrees of freedom
## AIC: 164.11
##
## Number of Fisher Scoring iterations: 4

#MuMin approach
options(na.action = "na.fail")
AllSubsets <- dredge(FullModel, rank = "AIC")

## Fixed term is "(Intercept)"

head(AllSubsets)

## Global model call: glm(formula = resp ~ Sex + SS + BK + Age + YR + CD, family = binomial(link = "logit",
## data = BirdData)
## ---
## Model selection table
##      (Intercept)      Age BK      CD Sex      YR df  logLik    AIC delta weight
## 35 -1.70500      +      0.05825  3 -79.057 164.1  0.00  0.228
## 36  0.34300 -0.04610  +      0.07485  4 -78.108 164.2  0.10  0.217
## 39 -1.91100      + 0.02840  4 -78.374 164.7  0.63  0.166
## 43 -1.56900      +      0.06561  4 -78.549 165.1  0.98  0.139
## 40 -0.07408 -0.04071  + 0.02375  5 -77.658 165.3  1.20  0.125
## 44  0.43050 -0.04533  +      0.08181  5 -77.661 165.3  1.21  0.125
## Models ranked by AIC(x)
```

## Odds Ratios and CIs

Using default ordering the odds ratio gives us non-birdkeepers versus birdkeepers. We might prefer to “reverse” the comparison. Two approaches: (1) Invert the Odds Ratio (and corresponding CI) or (2) Reorder the levels and refit the model.

```
exp(coef(Model1))

## (Intercept)      BKNoBird      YR
##  0.1818444  0.2286526  1.0599797

exp(confint(Model1))

## Waiting for profiling to be done...
##              2.5 %      97.5 %
## (Intercept) 0.0552766 0.5116995
## BKNoBird    0.1025833 0.4876239
## YR          1.0275721 1.0982798

1/0.2286526

## [1] 4.373447

BirdData$BK <- factor(BirdData$BK , levels(BirdData$BK ) [c(2,1)])
table(BirdData$BK, BirdData$LC)

##
##      LungCancer NoCancer
## NoBird      16      64
## Bird       33      34
```

```
Model2 <- glm(resp ~ BK + YR, data = BirdData, family=binomial(link="logit"))
summary(Model2)
```

```
##
## Call:
## glm(formula = resp ~ BK + YR, family = binomial(link = "logit"),
##      data = BirdData)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.6093  -0.8644  -0.5283   0.9479   2.0937
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.18016    0.63640  -4.997 5.82e-07 ***
## BKBird       1.47555    0.39588   3.727 0.000194 ***
## YR           0.05825    0.01685   3.458 0.000544 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 187.14  on 146  degrees of freedom
## Residual deviance: 158.11  on 144  degrees of freedom
## AIC: 164.11
##
## Number of Fisher Scoring iterations: 4
```

```
exp(coef(Model2))
```

```
## (Intercept)      BKBird      YR
##  0.04157919  4.37344710  1.05997966
```

```
exp(confint(Model2))
```

```
## Waiting for profiling to be done...
##              2.5 %    97.5 %
## (Intercept) 0.01063132 0.1311953
## BKBird      2.05076095 9.7481749
## YR          1.02757209 1.0982798
```

## Testing and Pairwise comparisons

The `hoslem.test()` function is from the `ResourceSelection` package. Need to choose the number of groups for testing. The `emmeans()` function from the `emmeans` package can be used for pairwise comparisons. When used with the `type = "response"` option, note that probabilities and odds ratios are returned. **Important Note:** Due to the case control design, it would be appropriate to report the results on the odds ratio scale. Converting to proportion (or probability) scale is just for illustration here.

```
Anova(Model2, type = 3)
```

```
## Analysis of Deviance Table (Type III tests)
##
## Response: resp
```

```
##      LR Chisq Df Pr(>Chisq)
## BK      15.053  1  0.0001046 ***
## YR      14.817  1  0.0001185 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

#H-L Lack of Fit Test
hoslem.test(Model2$y, fitted(Model2), g = 10)

##
## Hosmer and Lemeshow goodness of fit (GOF) test
##
## data:  Model2$y, fitted(Model2)
## X-squared = 7.5439, df = 8, p-value = 0.4792

#EMMeans Odds Ratio Scale
emmeans(Model2, pairwise ~ BK, type = "response")

## $emmeans
##      BK      prob      SE df asymp.LCL asymp.UCL
## NoBird 0.1739509 0.04395539 Inf 0.1036386 0.2772131
## Bird   0.4794293 0.06549858 Inf 0.3550992 0.6063609
##
## Confidence level used: 0.95
## Intervals are back-transformed from the logit scale
##
## $contrasts
##      contrast      odds.ratio      SE df z.ratio p.value
## NoBird / Bird  0.2286526 0.09051813 Inf  -3.727  0.0002
##
## Tests are performed on the log odds ratio scale
```

The rest of the output is primarily for illustration (to compare to the emmeans output above.)

```
#EMMeans Default Logit Scale
emmeans(Model2, pairwise ~ BK)

## $emmeans
##      BK      emmean      SE df asymp.LCL asymp.UCL
## NoBird -1.5578808 0.3059000 Inf -2.1574339 -0.9583278
## Bird   -0.0823293 0.2624385 Inf -0.5966994  0.4320408
##
## Results are given on the logit (not the response) scale.
## Confidence level used: 0.95
##
## $contrasts
##      contrast      estimate      SE df z.ratio p.value
## NoBird - Bird -1.475551 0.3958763 Inf  -3.727  0.0002
##
## Results are given on the log odds ratio (not the response) scale.

#Convert emmeans back to proportion scale
#NoBird
```

```

exp(-1.55788083)/(1+exp(-1.55788083))

## [1] 0.1739509
#Bird
exp(-0.08232932)/(1+exp(-0.08232932))

## [1] 0.4794293
#Remember the emmeans are just predicted values at average x
mean(BirdData$YR)

## [1] 27.85034
NewData <- data.frame(YR = rep(27.85034, 2), BK = c("NoBird","Bird"))
NewData

##           YR      BK
## 1 27.85034 NoBird
## 2 27.85034   Bird
predict(Model2, newdata = NewData)

##           1           2
## -1.55788084 -0.08232933
predict(Model2, newdata = NewData, type = "response")

##           1           2
## 0.1739509 0.4794293

```