# HW5

Nikhil Gopal

3/29/2021

**Question 1**

```
Crabs <- "http://users.stat.ufl.edu/~aa/cat/data/Crabs.dat"

Crabs <- read.table(file=Crabs, header=T)

m1 <- glm(y ~ weight + factor(color), data=Crabs, family=binomial)
```

1a:

```
summary(m1)
```

```
##
## Call:
## glm(formula = y ~ weight + factor(color), family = binomial,
##     data = Crabs)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1908  -1.0144   0.5101   0.8683   2.0751
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)     -3.2572     1.1985  -2.718  0.00657 **
## weight           1.6928     0.3888   4.354 1.34e-05 ***
## factor(color)2   0.1448     0.7365   0.197  0.84410
## factor(color)3  -0.1861     0.7750  -0.240  0.81019
## factor(color)4  -1.2694     0.8488  -1.495  0.13479
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 225.76  on 172  degrees of freedom
## Residual deviance: 188.54  on 168  degrees of freedom
## AIC: 198.54
##
## Number of Fisher Scoring iterations: 4
```

$$logit[\hat{\pi}] = -3.2572 + 1.69289(weight) + 0.1448(color2) + -0.1861(color3) + -1.2694(color4)$$

color 2:

$$logit(\hat{\pi}) = -3.1124 + 1.6928 * weight$$

color 3:

$$logit(\hat{\pi}) = -3.4422 + 1.6298 * weight$$

color 4:

$$logit(\hat{\pi}) = -4.5266 + 1.6298 * weight$$

color 1:

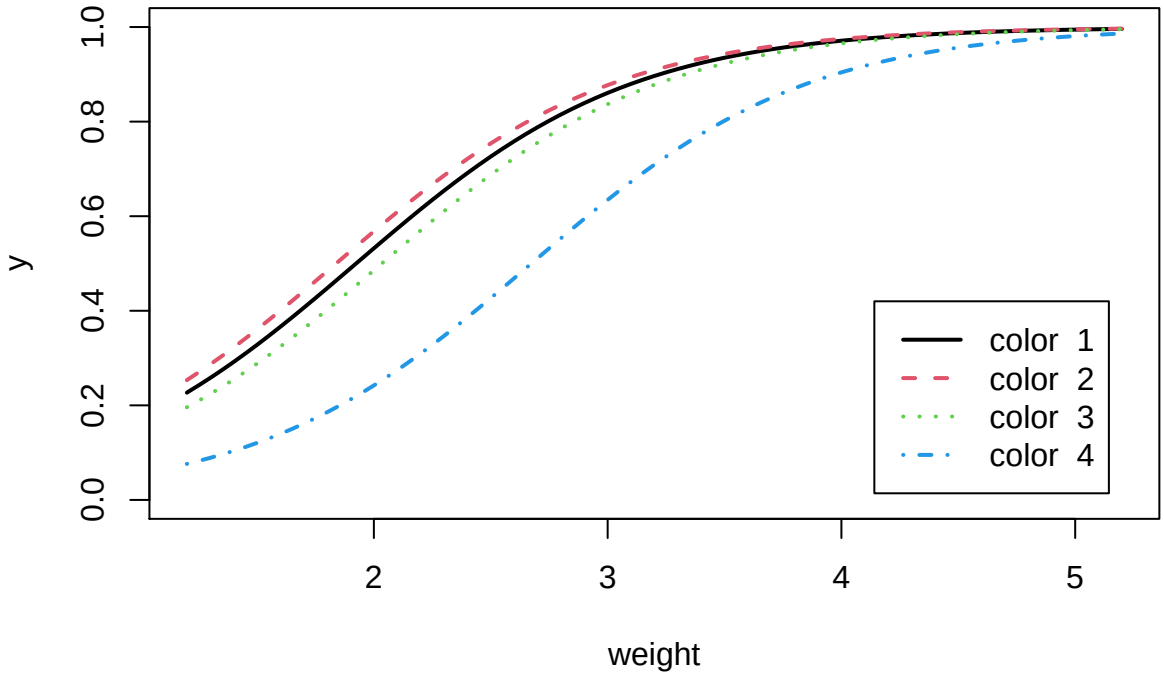
$$logit(\hat{\pi}) = -3.2572 + 1.6928 * weight$$

```
x <- seq(1.2, 5.2, .05)

plot(y ~ weight, data=Crabs, type="n")

for(k in 1:4){
  lines(x, predict(m1, data.frame(weight=x, color=k),type="response"), col=k,
        lty=k, lwd=2)
  }

legend("bottomright", inset=.05, lty=1:4, col=1:4, lwd=2,legend=paste("color ", 1:4))
```




$$logit(\hat{\pi}) = -1.6203 + 1.0483 * weight - 0.832c2 - 6.2964c3 + 0.4335c4 + 0.3613c2weight + 2.7065c3weight - 0.8536c4weight$$

color 2:

$$logit(\hat{\pi}) = -2.4523 + 1.4096 * weight$$

color 3:

$$logit(\hat{\pi}) = -7.9167 + 3.7548 * weight$$

color 4:

$$logit(\hat{\pi}) = -1.1868 + 0.1947 * weight$$

color 1:

$$logit(\hat{\pi}) = -.1.1868 + 0.1947 * weight$$

With the interaction term, the curves start moving differently for each color, and take different shapes. There is still a general trend of satellites being more likely with higher weights, but the curve is much steeper for color 3, and much flatter for curve 4.

1b:

```
m2 <- update(m1, ~ weight*factor(color))


summary(m2)
```

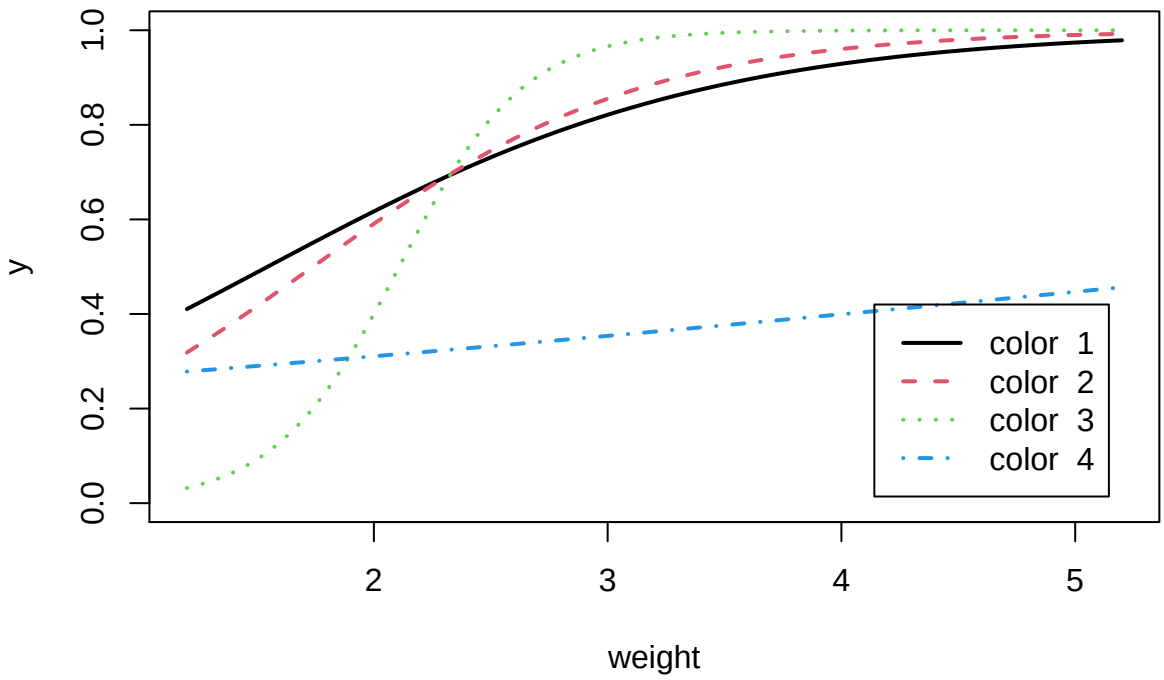
```
##
## Call:
## glm(formula = y ~ weight + factor(color) + weight:factor(color),
##     family = binomial, data = Crabs)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0875  -0.8766   0.5412   0.8399   1.9421
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)            -1.6203     4.8909  -0.331    0.740
## weight                  1.0483     1.8929   0.554    0.580
## factor(color)2         -0.8320     5.0311  -0.165    0.869
## factor(color)3         -6.2964     5.5165  -1.141    0.254
## factor(color)4          0.4335     5.4046   0.080    0.936
## weight:factor(color)2   0.3613     1.9559   0.185    0.853
## weight:factor(color)3   2.7065     2.2284   1.215    0.225
## weight:factor(color)4  -0.8536     2.1551  -0.396    0.692
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 225.76  on 172  degrees of freedom
## Residual deviance: 181.66  on 165  degrees of freedom
## AIC: 197.66
##
## Number of Fisher Scoring iterations: 5
```

$logit[\hat{P}(Y=1)] = -1.6203 + 1.0483(weight) - 0.832(color2) - 6.2964(color3) + 0.4335(color4) + 0.3613(weight:color2) + 2.7065($

```
plot(y ~ weight, data=Crabs, type="n")

for(k in 1:4){lines(x, predict(m2, data.frame(weight=x, color=k),type="response"), col=k, lty=k, lwd=2)}

legend("bottomright", inset=.05, lty=1:4, col=1:4, lwd=2,legend=paste("color ", 1:4))
```




The probability of having satellites increases as weight increases for every color. Colors 1-3 appear to trend toward having similar probabilities as weight increases. Color 4 still shows an increase in probability as weight increases, but the probability is much lower and the increase is much less.

c:

```
library(lmtest)
```

```
## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
lrtest(m1, m2)
```

```
## Likelihood ratio test
##
## Model 1: y ~ weight + factor(color)
## Model 2: y ~ weight + factor(color) + weight:factor(color)
##   #Df  LogLik Df Chisq Pr(>Chisq)
## 1   5 -94.271
## 2   8 -90.828  3 6.886    0.07562 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Our likelihood-ratio test returned a Chi-Square statistic of 6.886 on 3 df. The p value was 0.07562. This means that we cannot definitively say that our interaction model fits the data better than our simple model. The simple model has an AIC of 198.54 and the interaction model has an AIC of 197.66.

**Question 2**

2a:

```
mb <- read.table("http://users.stat.ufl.edu/~aa/cat/data/MBTI.dat", header = T)

mb$p <- mb$drink/mb$n

mb.mod <- glm(p ~ EI + SN + TF + JP, family=binomial, weights=mb$n, data = mb)

modd <- glm(p ~ EI * SN * TF * JP, family=binomial, weights=mb$n, data = mb)


mb.mod$deviance; mb.mod$df.residual
```

```
## [1] 11.14907
```

```
## [1] 11
```

```
1 - pchisq(mb.mod$deviance, mb.mod$df.residual)
```

```
## [1] 0.4308605
```

```
anova(mb.mod, modd, test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: p ~ EI + SN + TF + JP
## Model 2: p ~ EI * SN * TF * JP
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1        11     11.149
## 2         0      0.000 11   11.149   0.4309
```

The residual deviance is 11.14907. The p value for the GOF test is also about 0.43, meaning that we fail to reject our null hypothesis and thus we do not have evidence of a lack of fit, indicating that our simple model is fine compared to the complex model.

2b:

I would remove the JP term, as it was shown to not be statistically significant, and thus we are unsure that it actually has an effect on the proportion who drink.

2c:

```
mb.inter <- glm(p ~ EI + SN + TF + JP + EI:SN + EI:TF + EI:JP + SN:TF + SN:JP + TF:JP, family=binomial,

AIC(mb.mod)
```

```
## [1] 73.98986
```

```
AIC(mb.inter)
```

```
## [1] 78.58169
```

```
library(lmtest)
lrtest(mb.mod, mb.inter)
```

```
## Likelihood ratio test
##
## Model 1: p ~ EI + SN + TF + JP
## Model 2: p ~ EI + SN + TF + JP + EI:SN + EI:TF + EI:JP + SN:TF + SN:JP +
##     TF:JP
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   5 -31.995
## 2  11 -28.291  6 7.4082     0.2847
```

The likelihood ratio test returned a p value of 0.2847, meaning that we cannot be sure that the interaction model fits the data better than the simpler model, meaning we would choose the simpler model based of the likelihood ratio test.

The AIC of the simple model was 73.99 and the AIC of the interactions model was 78.58, meaning that based of AIC we would also choose the simpler model.

2d:

```
library(MASS)
fit7 <- glm(p ~ 1, family = binomial,weights=mb$n, data = mb)

scope <- list(upper=formula(mb.mod), lower=formula(fit7))
scope2 <- list(upper=formula(mb.inter), lower=formula(fit7))

stepAIC(fit7, direction = "forward", scope = scope2)
```

```
## Start:  AIC=85.33
## p ~ 1
##
##        Df Deviance    AIC
## + TF    1   23.683 80.523
## + EI    1   24.036 80.877
## + SN    1   26.832 83.673
```

```
## <none>      30.488 85.329
## + JP    1   29.508 86.348
##
## Step:  AIC=80.52
## p ~ TF
##
##        Df Deviance    AIC
## + EI    1   16.398 75.239
## + SN    1   18.469 77.310
## + JP    1   21.631 80.472
## <none>      23.683 80.523
##
## Step:  AIC=75.24
## p ~ TF + EI
##
##         Df Deviance    AIC
## + SN     1   11.945 72.786
## <none>       16.398 75.239
## + JP     1   14.436 75.277
## + EI:TF  1   14.984 75.825
##
## Step:  AIC=72.79
## p ~ TF + EI + SN
##
##         Df Deviance    AIC
## + SN:TF  1   8.2328 71.074
## <none>      11.9455 72.786
## + EI:TF  1  10.5461 73.387
## + JP     1  11.1491 73.990
## + EI:SN  1  11.3814 74.222
##
## Step:  AIC=71.07
## p ~ TF + EI + SN + TF:SN
##
##         Df Deviance    AIC
## <none>       8.2328 71.074
## + EI:TF  1   7.0895 71.930
## + JP     1   7.4797 72.321
## + EI:SN  1   7.8198 72.661


##
## Call:  glm(formula = p ~ TF + EI + SN + TF:SN, family = binomial, data = mb,
##     weights = mb$n)
##
## Coefficients:
## (Intercept)          TFt          EIi          SNs      TFt:SNs
##    -1.76795      0.07959     -0.55499     -0.86844      0.89962
##
## Degrees of Freedom: 15 Total (i.e. Null);  11 Residual
## Null Deviance:       30.49
## Residual Deviance: 8.233     AIC: 71.07
```

```r
stepAIC(fit7, direction = "forward", scope = scope)
```

```
## Start:  AIC=85.33
## p ~ 1
##
##        Df Deviance    AIC
## + TF    1   23.683 80.523
## + EI    1   24.036 80.877
## + SN    1   26.832 83.673
## <none>      30.488 85.329
## + JP    1   29.508 86.348
##
## Step:  AIC=80.52
## p ~ TF
##
##        Df Deviance    AIC
## + EI    1   16.398 75.239
## + SN    1   18.469 77.310
## + JP    1   21.631 80.472
## <none>      23.683 80.523
##
## Step:  AIC=75.24
## p ~ TF + EI
##
##        Df Deviance    AIC
## + SN    1   11.945 72.786
## <none>      16.398 75.239
## + JP    1   14.436 75.277
##
## Step:  AIC=72.79
## p ~ TF + EI + SN
##
##        Df Deviance    AIC
## <none>      11.945 72.786
## + JP    1   11.149 73.990

##
## Call:  glm(formula = p ~ TF + EI + SN, family = binomial, data = mb,
##     weights = mb$n)
##
## Coefficients:
## (Intercept)          TFt          EIi          SNs
##     -1.9678       0.6601      -0.5518      -0.4843
##
## Degrees of Freedom: 15 Total (i.e. Null);  12 Residual
## Null Deviance:       30.49
## Residual Deviance: 11.95     AIC: 72.79
```

```r
stepAIC(mb.inter, direction = "backward", scope = scope2)
```

```
## Start:  AIC=78.58
## p ~ EI + SN + TF + JP + EI:SN + EI:TF + EI:JP + SN:TF + SN:JP +
```

```
##     TF:JP
##
##         Df Deviance    AIC
## - EI:SN  1   3.7421 76.583
## - SN:JP  1   4.0796 76.920
## - TF:JP  1   4.1179 76.959
## - EI:TF  1   4.1617 77.003
## - EI:JP  1   4.9884 77.829
## - SN:TF  1   5.4760 78.317
## <none>       3.7409 78.582
##
## Step:  AIC=76.58
## p ~ EI + SN + TF + JP + EI:TF + EI:JP + SN:TF + SN:JP + TF:JP
##
##         Df Deviance    AIC
## - SN:JP  1   4.0805 74.921
## - TF:JP  1   4.1211 74.962
## - EI:TF  1   4.1925 75.033
## - EI:JP  1   5.1500 75.991
## - SN:TF  1   5.4928 76.334
## <none>       3.7421 76.583
##
## Step:  AIC=74.92
## p ~ EI + SN + TF + JP + EI:TF + EI:JP + SN:TF + TF:JP
##
##         Df Deviance    AIC
## - EI:TF  1   4.5804 73.421
## - TF:JP  1   4.6241 73.465
## - EI:JP  1   5.5449 74.386
## <none>       4.0805 74.921
## - SN:TF  1   6.1747 75.016
##
## Step:  AIC=73.42
## p ~ EI + SN + TF + JP + EI:JP + SN:TF + TF:JP
##
##         Df Deviance    AIC
## - TF:JP  1   5.2274 72.068
## <none>       4.5804 73.421
## - EI:JP  1   6.7014 73.542
## - SN:TF  1   6.7546 73.595
##
## Step:  AIC=72.07
## p ~ EI + SN + TF + JP + EI:JP + SN:TF
##
##         Df Deviance    AIC
## <none>       5.2274 72.068
## - EI:JP  1   7.4797 72.321
## - SN:TF  1   8.2375 73.078


##
## Call:  glm(formula = p ~ EI + SN + TF + JP + EI:JP + SN:TF, family = binomial,
##     data = mb, weights = mb$n)
##
## Coefficients:
```

```
## (Intercept)          EIi          SNs          TFt          JPp      EIi:JPp
##     -2.0821      -0.2243      -0.8034       0.1659       0.4958      -0.6589
##     SNs:TFt
##      0.8185
##
## Degrees of Freedom: 15 Total (i.e. Null);  9 Residual
## Null Deviance:       30.49
## Residual Deviance: 5.227     AIC: 72.07
```

```
stepAIC(mb.mod, direction = "backward", scope = scope)
```

```
## Start:  AIC=73.99
## p ~ EI + SN + TF + JP
##
##        Df Deviance    AIC
## - JP    1   11.945 72.786
## <none>      11.149 73.990
## - SN    1   14.436 75.277
## - EI    1   17.745 78.586
## - TF    1   20.807 81.648
##
## Step:  AIC=72.79
## p ~ EI + SN + TF
##
##        Df Deviance    AIC
## <none>      11.945 72.786
## - SN    1   16.398 75.239
## - EI    1   18.469 77.310
## - TF    1   21.037 79.878

##
## Call:  glm(formula = p ~ EI + SN + TF, family = binomial, data = mb,
##     weights = mb$n)
##
## Coefficients:
## (Intercept)          EIi          SNs          TFt
##     -1.9678      -0.5518      -0.4843       0.6601
##
## Degrees of Freedom: 15 Total (i.e. Null);  12 Residual
## Null Deviance:       30.49
## Residual Deviance: 11.95     AIC: 72.79
```

The best model is given below, as it has the lowest AIC (71.074):

```
best_mod <- glm(formula = p ~ TF + EI + SN + TF:SN, family = binomial, weights = mb$n, data = mb)

summary(best_mod)
```

```
##
## Call:
## glm(formula = p ~ TF + EI + SN + TF:SN, family = binomial, data = mb,
##     weights = mb$n)
```

```
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.2673  -0.5975  -0.2545   0.5777   1.0198
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.76795    0.22630  -7.812 5.61e-15 ***
## TFt          0.07959    0.38550   0.206  0.83644
## EIi         -0.55499    0.21731  -2.554  0.01065 *
## SNs         -0.86844    0.30356  -2.861  0.00423 **
## TFt:SNs      0.89962    0.47632   1.889  0.05894 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 30.4880  on 15  degrees of freedom
## Residual deviance:  8.2328  on 11  degrees of freedom
## AIC: 71.074
##
## Number of Fisher Scoring iterations: 4
```

### Question 3

```
Dept <- rep(1:6, rep(2,6))

Gender <- rep(c("Male","Female"), 6)

Yes <- c(512,89,353,17,120,202,138,131,53,94,22,24)

No <- c(313,19,207,8,205,391,279,244,138,299,351,317)

Data <- data.frame(Dept=Dept, Gender=Gender, Yes=Yes, No=No)

rm(Dept, Gender, Yes, No)

Data
```

```
##    Dept Gender Yes  No
## 1     1   Male 512 313
## 2     1 Female  89  19
## 3     2   Male 353 207
## 4     2 Female  17   8
## 5     3   Male 120 205
## 6     3 Female 202 391
## 7     4   Male 138 279
## 8     4 Female 131 244
## 9     5   Male  53 138
## 10    5 Female  94 299
## 11    6   Male  22 351
## 12    6 Female  24 317
```

```r
# Dept effect only, no Gender
m1 <- glm(cbind(Yes,No) ~ factor(Dept), data=Data, family=binomial)
summary(m1)
```

```
##
## Call:
## glm(formula = cbind(Yes, No) ~ factor(Dept), family = binomial,
##     data = Data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.4064  -0.4550   0.1456   0.5471   4.1323
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)    0.59346    0.06838   8.679   <2e-16 ***
## factor(Dept)2 -0.05059    0.10968  -0.461    0.645
## factor(Dept)3 -1.20915    0.09726 -12.432   <2e-16 ***
## factor(Dept)4 -1.25833    0.10152 -12.395   <2e-16 ***
## factor(Dept)5 -1.68296    0.11733 -14.343   <2e-16 ***
## factor(Dept)6 -3.26911    0.16707 -19.567   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 877.056  on 11  degrees of freedom
## Residual deviance:  21.736  on  6  degrees of freedom
## AIC: 102.68
##
## Number of Fisher Scoring iterations: 4
```

3a:

```r
sat <- update(m1, ~ factor(Dept)*Gender)
anova(m1, sat, test="Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: cbind(Yes, No) ~ factor(Dept)
## Model 2: cbind(Yes, No) ~ factor(Dept) + Gender + factor(Dept):Gender
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1         6     21.735
## 2         0      0.000  6   21.735 0.001352 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The goodness of fit test returned a p value of 0.001352, indicating that the interaction model fits the data beter than the simple model, with statistical significance.

3b:

```
Data$y.hat <- (Data$Yes + Data$No) * fitted(m1)
Data$resid <- rstandard(m1, type="pearson")
Data
```

```
##    Dept Gender Yes  No     y.hat      resid
## 1     1   Male 512 313 531.43087 -4.1530728
## 2     1 Female  89  19  69.56913  4.1530728
## 3     2   Male 353 207 354.18803 -0.5037077
## 4     2 Female  17   8  15.81197  0.5037077
## 5     3   Male 120 205 113.99782  0.8680662
## 6     3 Female 202 391 208.00218 -0.8680662
## 7     4   Male 138 279 141.63258 -0.5458732
## 8     4 Female 131 244 127.36742  0.5458732
## 9     5   Male  53 138  48.07705  1.0005342
## 10    5 Female  94 299  98.92295 -1.0005342
## 11    6   Male  22 351  24.03081 -0.6197526
## 12    6 Female  24 317  21.96919  0.6197526
```

The residuals for those applicants to department 1 were relatively large compared to the other departments about 4.15, however the model was able to predict with greater accuracy an applicant's likelihood of gaining admission for the other departments, with residuals of between about 0.55-1.0.

3c:

```
m2 <- update(m1, ~ . + Gender)
exp(coef(m2))
```

```
##   (Intercept) factor(Dept)2 factor(Dept)3 factor(Dept)4 factor(Dept)5
##    1.97767415    0.95753028    0.28291804    0.27400567    0.17564230
## factor(Dept)6    GenderMale
##    0.03664494    0.90495497
```

As shown in the output above, the conditional odds ratio between admissions and gender is about 0.90.

3d

```
Count <- c(sum(Data$Yes[Data$Gender=="Male"]),sum(Data$Yes[Data$Gender=="Female"]),sum(Data$No[ Data$Ge

Table <- matrix(Count, 2, 2)

rownames(Table) <- c("Male", "Female")
colnames(Table) <- c("Yes", "No")
Table
```

```
##         Yes   No
## Male   1198 1493
## Female  557 1278
```

```
Table[1,1] * Table[2,2] / ( Table[1,2] * Table[2,1] )
```

```
## [1] 1.84108
```

```r
# Also
m1a <- update(m1, ~ Gender)

exp(coef(m1a))
```

```
## (Intercept)  GenderMale
##   0.4358372   1.8410800
```

As demonstrated in the output above, the marginal table, collapsed over department, has odds ratio 1.84.

3e

```r
admit_rates <- data.frame(

  Dept <- c(1,2,3,4,5,6),
  Male <- c(0.620606061,
0.630357143,
0.369230769,
0.330935252,
0.277486911,
0.058981233
),
  Female <- c(
    0.824074074,
0.68,
0.340640809,
0.349333333,
0.239185751,
0.070381232)
)

names(admit_rates) <- c("Dept", "Male", "female")

admit_rates
```

```
##   Dept       Male     female
## 1    1 0.62060606 0.82407407
## 2    2 0.63035714 0.68000000
## 3    3 0.36923077 0.34064081
## 4    4 0.33093525 0.34933333
## 5    5 0.27748691 0.23918575
## 6    6 0.05898123 0.07038123
```

As we can see from the table above, there does not appear to be a large disparity in the admission rates to the departments between genders, they are all relatively similar for each department (except department 1). However, there is a big difference in the admission rates between departments, which explains the difference in the Odds Ratios.

Additionally, this is an example of simpson's paradox (like the baseball problem). When divided along different categories, different conclusions can be made as to what improves or supresses one's chances of admission. However, associations are not always causal, which explains the disparities between the associations. Maybe males tend to apply more often to departments that have higher acceptance rates, which does not necessarily mean that the school is discriminating against males.

**Question 4**

```
filename <- "http://users.stat.ufl.edu/~aa/cat/data/Alligators2.dat"

Data <- read.table(file=filename, header=T)

names(Data) <- c("lake","size","F","I","R","B","O")

Sizes <- c(" < 2.3", " > 2.3")

Data$Size <- factor(rep(Sizes, 4), levels=Sizes[c(2,1)])

Lakes <- c("George", "Hancock", "Oklawaha", "Trafford")

Data$Lake <- factor(rep(Lakes[c(2,3,4,1)], rep(2,4)), levels=Lakes)

library(VGAM)
```

```
## Loading required package: stats4

## Loading required package: splines

##
## Attaching package: 'VGAM'

## The following object is masked from 'package:lmtest':
##
##     lrtest
```

4a:

```
fit <- vglm(cbind(I,R,B,O,F) ~ Size + Lake, data=Data, family=multinomial)

coefs <- round(coef(fit), 2)

coefs <- matrix(coefs, 4, 5)

rownames(coefs) <- paste("log(pi[", c("I","R","B","O"), "]/pi[F])",sep="")

colnames(coefs) <- c("Intercept", "Length<2.3","Hancock", "Oklawaha", "Trafford")

summary(fit)
```

```
##
## Call:
## vglm(formula = cbind(I, R, B, O, F) ~ Size + Lake, family = multinomial,
##     data = Data)
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept):1  -1.549019   0.424922  -3.645 0.000267 ***
## (Intercept):2  -3.314533   1.053084      NA       NA
## (Intercept):3  -2.093077   0.662236  -3.161 0.001574 **
```

```
## (Intercept):4  -1.904272   0.525825  -3.621 0.000293 ***
## Size < 2.3:1    1.458205   0.395945   3.683 0.000231 ***
## Size < 2.3:2   -0.351263   0.580033  -0.606 0.544786
## Size < 2.3:3   -0.630660   0.642480  -0.982 0.326296
## Size < 2.3:4    0.331550   0.448247   0.740 0.459506
## LakeHancock:1  -1.658359   0.612877  -2.706 0.006813 **
## LakeHancock:2   1.242777   1.185432   1.048 0.294466
## LakeHancock:3   0.695118   0.781263   0.890 0.373608
## LakeHancock:4   0.826196   0.557540   1.482 0.138378
## LakeOklawaha:1  0.937219   0.471906   1.986 0.047030 *
## LakeOklawaha:2  2.458872   1.118128   2.199 0.027871 *
## LakeOklawaha:3 -0.653208   1.202098  -0.543 0.586861
## LakeOklawaha:4  0.005653   0.776513   0.007 0.994191
## LakeTrafford:1  1.121985   0.490513   2.287 0.022174 *
## LakeTrafford:2  2.935253   1.116409   2.629 0.008559 **
## LakeTrafford:3  1.087767   0.841669   1.292 0.196221
## LakeTrafford:4  1.516369   0.621435   2.440 0.014683 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Names of linear predictors: log(mu[,1]/mu[,5]), log(mu[,2]/mu[,5]),
## log(mu[,3]/mu[,5]), log(mu[,4]/mu[,5])
##
## Residual deviance: 17.0798 on 12 degrees of freedom
##
## Log-likelihood: -47.5138 on 12 degrees of freedom
##
## Number of Fisher scoring iterations: 5
##
## Warning: Hauck-Donner effect detected in the following estimate(s):
## '(Intercept):2'
##
##
## Reference group is level  5  of the response
```

Invertebrates:

$$log() = -1.55 + 1.46(size) - 1.66(hancock) + 0.94(Oklawaha) + 1.12(Trafford)$$

Reptiles:

$$log() = -3.31 + -0.35(size) + 1.24(hancock) + 2.46(Oklawaha) + 2.94(Trafford)$$

Birds:

$$log() = -2.09 + -0.63(size) + 0.7(hancock) + -0.65(Oklawaha) + 1.09(Trafford)$$

Others:

$$log() = -1.9 + 0.33(size) + 0.83(hancock) + 0.006(Oklawaha) + 1.52(Trafford)$$

4b:

The coefficient on length for invertebrates is about 1.46. This means that the log odds of having Invertebrates instead of fish as the primary food choice increases if the alligator's length is less than 2.3 meters. For a given lake, the log odds that the food choice is an invertebrate increases by about 4.29 for a given value.

4c:

```
a <- data.frame(
  "lake" = c("H > 2.3", "H < 2.3", "O > 2.3", "O < 2.3", "T > 2.3", "T < 2.3", "G > 2.3", "G < 2.3"),
  "Probability" <- fitted(fit)[,5]

  )

names(a) <- c("lake", "Probability")

a
```

```
##      lake Probability
## 1 H > 2.3   0.5353035
## 2 H < 2.3   0.5701978
## 3 O > 2.3   0.2581861
## 4 O < 2.3   0.4584385
## 5 T > 2.3   0.1842990
## 6 T < 2.3   0.2957525
## 7 G > 2.3   0.4521040
## 8 G < 2.3   0.6574425
```

The above output lists the probability that the primary food choice is fish for each length. For length $> 2.3$ meters in lake Okalawaha, the probability that the primary food choice is fish is 0.2581861, for length $< 2.3$ meters, the probability is 0.4584385.

**Question 5**

5a:

prediction equation:

$$log(\hat{\pi}_r/\hat{\pi}_D) = -2.3 + 0.5x$$

5b:

$$0 = log(1/1) = -2.3 + 0.5x$$

$$x = 4.6$$

Thus, pi_hat_R > pi_hat_D when annual income exceeds $46,000.

5c:

$$\hat{\pi}_I(x) = 0 + 0x$$

$$\hat{\pi}_I = \frac{e^0}{e^{1+0.3x} + e^{3.3-0.2x} + e^0}$$

$$\hat{\pi}_I = \frac{1}{e^{1+0.3x} + e^{3.3-0.2x} + 1}$$

**Question 6**

```
Not <- c(6,6,6);
Pretty <- c(43,113,57);
Very <- c(75,178,117);
Income <- c("Below", "Average", "Above")
scores = c(1,2,3)
data.frame(Income, Not, Pretty, Very)
```

```
##    Income Not Pretty Very
## 1   Below   6     43   75
## 2 Average   6    113  178
## 3   Above   6     57  117
```

```
happy <- vglm(cbind(Pretty, Very, Not) ~ scores, family=multinomial)
```

6a:

```
coef(happy)
```

```
## (Intercept):1 (Intercept):2      scores:1      scores:2
##     2.2038939     2.5551795     0.1313533     0.2275057
```

$$log(\hat{\pi}_1/\hat{\pi}_3) = 2.2038939 + 0.1313533x$$

$$log(\hat{\pi}_2/\hat{\pi}_3) = 2.5551795 + 0.2275057x$$

6b:

Since the beta coefficient for income is positive, the odds of being in the higher category increase as income increases (from not happy > pretty happy in the 1st equation and not happy > very happy in the second equation).

6c:

```
lrtest(happy)
```

```
## Likelihood ratio test
##
## Model 1: cbind(Pretty, Very, Not) ~ scores
## Model 2: cbind(Pretty, Very, Not) ~ 1
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   2 -15.386
## 2   4 -15.858  2 0.9439     0.6238
```

```
anova.vglm(happy)
```

```
## Analysis of Deviance Table (Type II tests)
##
## Model: 'multinomial', 'VGAMcategorical'
##
```

```
## Link: 'multilogitlink'
##
## Response: cbind(Pretty, Very, Not)
##
##        Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## scores  2  0.94387         4     4.1348   0.6238
```

The likelihood ratio test returned a chi square statistic of 0.94 on 2 degrees of freedom, which corresponds to a p value of 0.62. We thus do not have evidence that our model fits the data better than the simple model, and thus we do not have evidence that happiness is independent of income.

6d:

```
summary(happy)
```

```
##
## Call:
## vglm(formula = cbind(Pretty, Very, Not) ~ scores, family = multinomial)
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept):1   2.2039     0.7376   2.988 0.002807 **
## (Intercept):2   2.5552     0.7256   3.521 0.000429 ***
## scores:1        0.1314     0.3468   0.379 0.704906
## scores:2        0.2275     0.3412   0.667 0.504907
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Names of linear predictors: log(mu[,1]/mu[,3]), log(mu[,2]/mu[,3])
##
## Residual deviance: 3.1909 on 2 degrees of freedom
##
## Log-likelihood: -15.3864 on 2 degrees of freedom
##
## Number of Fisher scoring iterations: 4
##
## No Hauck-Donner effect found in any of the estimates
##
##
## Reference group is level  3  of the response
```

```
1-pchisq(3.1909, 2)
```

```
## [1] 0.2028172
```

The deviance goodness of fit test returned a p value of 0.2028172. Thus, we can safely say that our model fits the data well.

6e:

```
predict(happy,data.frame(score=2), type="response")
```

```
##      Pretty      Very        Not
## 1 0.3757865 0.5878425 0.03637108
## 2 0.3562457 0.6135187 0.03023560
## 3 0.3366529 0.6382915 0.02505563
```

The probability that someone with average family income reports a very happy marriage is 0.6135187.