WFC198\_Bolas\_Hwk3\_logistic\_and\_poisson

Ellie Bolas

April 23, 2018

## Logitistic Regressions Homework

Remember: Logistic Regression: relationship with the predictor variable (x) is modeled on the success probability (expected value) of the Bernoulli random variable (y).

logit(pi) = beta0 + beta1xi Y ~ Binomial(pi) #bernoulli is a special case of binomial. A binomial with one attempt = bernoulli. Y is predicted by the binomial distribution of pi pi = *expected value* of a Bernoulli random variable/occurence probability Y = actual observations, response, 0 or 1 xi = *predictor* beta0 + beta1xi = *linear predictor* p is between 0 or 1, we need to transform it to infinite so we can model predicors, use logit link

logistic regression will return the maximum likelihood parameter estimates for beta0 and beta1 on a logit scale, returns estimates of how x affects logit(p). Then, back-transform

# back-transform: pi = (exp(logit(pi)))/(exp(logit(pi))+1)

## Task 1: Simulating logistic regression data: Simulate data for a logistic regression with one predictor variable

1. The intercept of the linear predictor is -0.2 (that is, minus 0.2); the slope (describing the effect of covariate X) is 1.3. The sample size (number of data points) is 30. (1 Pt)

beta0 <- (-0.2) #intercept, logi(p) when x = 0  
beta1 <- 1.3 #slope, effect of x on logit(p)  
n <- 30 #sample size, number of observations

1. The predictor variable, X, comes from a uniform distribution with lower limit -1 (that is, minus 1) and upper limit 1. (1 Pt)

#runif generates uniform distribution.uniform distribution means any variable between -1 and 1. need to tell is n=number of observations and the upper and lower limits of the distribution  
  
x <- runif(n, -1, 1) #put in paramters of min=-1, max=1, this is some kind of continuous variable, take 30 samples from this distribution

1. Calculate the expected values both on the logit and on the response (i.e., probability) scale (2 Pts)

#to calculate expected values on logit scale, l.p=logit(p)  
  
l.p <- beta0 + beta1\*x #bc x is a vector, then l.p returns a string for each possible x  
  
#p=probability, applying the inverse of the logit here to get rid of the log to put it on a probability scale, y depends on p  
p <- exp(l.p)/(exp(l.p) + 1)  
p #just checking that this is between 0 and 1

## [1] 0.3387012 0.6917888 0.5397297 0.2411923 0.6293545 0.2784554 0.1968797  
## [8] 0.4036637 0.3471610 0.2117725 0.7123715 0.3820065 0.6200270 0.6250079  
## [15] 0.3463688 0.3824616 0.1910443 0.2235264 0.3757532 0.2626538 0.2959942  
## [22] 0.3354794 0.7379764 0.2259358 0.1867894 0.2252693 0.5346248 0.3779282  
## [29] 0.7272512 0.3467773

1. Use the rbinom() function to generate the response variable Y. Note that you need to specify three arguments in rbinom(): how many observations to generate (n), the number of Bernoulli trials for each of these observations (which is 1, because each data point in a logistic regression is Binary), and the probability (ie, expected value on the natural scale). Use ?rbinom for more help. (2 Pts).

#generate Bernoulli random variables for y  
#need to find the probability of success for each trial, use n=number of trials, p=vector of probabilities (p), size=number of trials per observation  
  
y <- rbinom(n, 1, p) #will create a vector of y for each value of p, vector is all 0s and 1s

## Task 2: Analyzing data from Task 1

1. As we have done in lab, analyze the data you generated under Q1 using logistic regression. Include the covariate in your model. Make sure your “knitted” report shows the summary output from your model\* (2 Pts)

#make a model  
lrmod <-glm(y~x, family = "binomial")  
summary(lrmod)

##   
## Call:  
## glm(formula = y ~ x, family = "binomial")  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.5728 -0.8673 -0.5812 0.8028 1.9238   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -0.1242 0.4381 -0.283 0.7768   
## x 1.8147 0.7683 2.362 0.0182 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 40.381 on 29 degrees of freedom  
## Residual deviance: 33.375 on 28 degrees of freedom  
## AIC: 37.375  
##   
## Number of Fisher Scoring iterations: 4

#logit(pi) = -0.5615 + 0.9287x

1. Calculate the differences between the parameter estimates and the true input parameters, and save these differences in a numerical vector. Make sure your “knitted” report shows the values in the numerical vector\*. TIP: you can use the coefficients() function to extract parameter estimates from a model object (2 Pts)

predict.par <- coef(lrmod) #pulled the coefficients from the model  
predict.par

## (Intercept) x   
## -0.1241894 1.8146570

true.par <- c(beta0, beta1) #true parameters  
diff <- (predict.par-true.par)  
diff #difference between extimates and true parameters

## (Intercept) x   
## 0.07581056 0.51465695

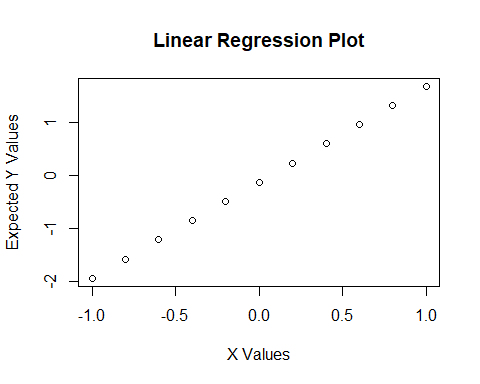
1. Generate expected values of Y on the link scale (ie., the logit scale) based on the model fit under a) for a sequence of values of X ranging from -1 to 1 at intervals of 0.2. (2 Pts)

newx<- data.frame(x = seq(-1,1, 0.2)) #making new data with new x values  
  
#generate expected y's  
exp.y <- predict.glm(object = lrmod, newdata =newx, type = "link", se.fit = TRUE ) #response returns data on the scale of the response variable, so a default binmoial model returns probabilities on the logit sclae, "response" returns predicted probabilities (automatically back transforms)  
  
exp.y # "fit" is the name of the expected y vectors, "se.fit"" are the standard errors

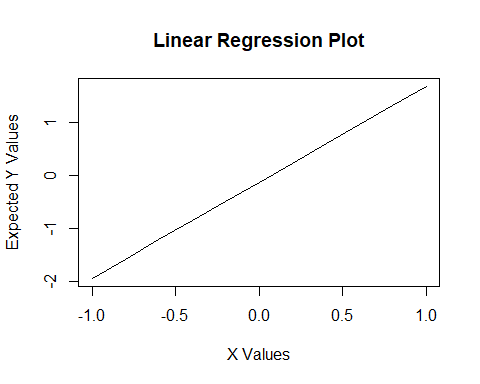
## $fit  
## 1 2 3 4 5 6   
## -1.9388464 -1.5759150 -1.2129836 -0.8500522 -0.4871208 -0.1241894   
## 7 8 9 10 11   
## 0.2387419 0.6016733 0.9646047 1.3275361 1.6904675   
##   
## $se.fit  
## 1 2 3 4 5 6 7   
## 0.7788772 0.6551885 0.5469111 0.4649397 0.4247865 0.4381045 0.5006446   
## 8 9 10 11   
## 0.5971365 0.7139442 0.8426613 0.9785996   
##   
## $residual.scale  
## [1] 1

1. Plot these expected values against the sequence of values of X generated in c). Appropriately label your axes. (2 Pts)

plot(newx$x, exp.y$fit, main = "Linear Regression Plot", xlab = "X Values", ylab = "Expected Y Values") #pulls the x values out of the df, and the expected y values



#not sure if the plot should be a line plot:  
plot(newx$x, exp.y$fit, main = "Linear Regression Plot", xlab = "X Values", ylab = "Expected Y Values", type = "l")



## Task 3: Analyzing Poisson regression data

The data file “HW3data.csv” contains two columns, “Species”, showing the number of bird species detected at a given site (response variable, count), and “Quality”, and index of habitat quality at that site (predictor variable), where higher numbers indicate higher quality.

1. Read the data file into R and perform a Poisson regression (with the covariate) on the data (2 Pts)

getwd()

## [1] "C:/Users/ebola/Google Drive/Git/WFC198\_Git/WFC198/Lab3\_Hwk3"

birddat <- read.csv("HW3data.csv")  
head(birddat)

## Species Quality  
## 1 6 1.920514  
## 2 5 2.150402  
## 3 7 1.505326  
## 4 5 1.933705  
## 5 5 1.652852  
## 6 5 1.851024

poismod <- glm(Species~Quality, data= birddat, family = "poisson")  
summary(poismod)

##   
## Call:  
## glm(formula = Species ~ Quality, family = "poisson", data = birddat)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.70274 -0.33175 -0.04382 0.39331 1.29872   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 1.1422 0.1267 9.017 < 2e-16 \*\*\*  
## Quality 0.2672 0.0840 3.181 0.00147 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for poisson family taken to be 1)  
##   
## Null deviance: 70.158 on 96 degrees of freedom  
## Residual deviance: 59.439 on 95 degrees of freedom  
## AIC: 377.1  
##   
## Number of Fisher Scoring iterations: 5

#intercept: 1.1422, slope: 0.2672

1. Generate expected values of Y on the response scale based on the model fit under a) for a sequence of habitat quality values, ranging from the minimum observed quality to the maximum observed quality, at intervals of 0.1 (2 Pts)

#need to generate a new set of x based on the min and max of the original data  
  
minbird <- min(birddat$Quality)  
maxbird <- max(birddat$Quality)  
#newx2 <- data.frame(z = seq(minbird, maxbird, 0.1)) here, bc I re-named the column z, predict.glm got confused and couldn't find the Quality variable. Use the same variable name throughout  
newx2 <- data.frame(Quality = seq(minbird, maxbird, 0.1))  
newx2 #this will give us the expected values for the full length of x, gives a more complete sequence

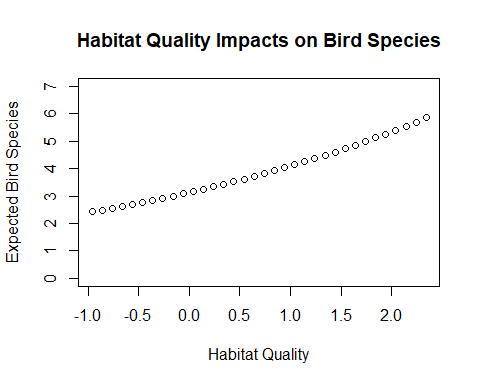
## Quality  
## 1 -0.95971684  
## 2 -0.85971684  
## 3 -0.75971684  
## 4 -0.65971684  
## 5 -0.55971684  
## 6 -0.45971684  
## 7 -0.35971684  
## 8 -0.25971684  
## 9 -0.15971684  
## 10 -0.05971684  
## 11 0.04028316  
## 12 0.14028316  
## 13 0.24028316  
## 14 0.34028316  
## 15 0.44028316  
## 16 0.54028316  
## 17 0.64028316  
## 18 0.74028316  
## 19 0.84028316  
## 20 0.94028316  
## 21 1.04028316  
## 22 1.14028316  
## 23 1.24028316  
## 24 1.34028316  
## 25 1.44028316  
## 26 1.54028316  
## 27 1.64028316  
## 28 1.74028316  
## 29 1.84028316  
## 30 1.94028316  
## 31 2.04028316  
## 32 2.14028316  
## 33 2.24028316  
## 34 2.34028316

#now calculate expected y on response scale  
exp.y2 <- predict.glm(object=poismod, newdata= newx2, type = "response", se.fit =T)  
exp.y2

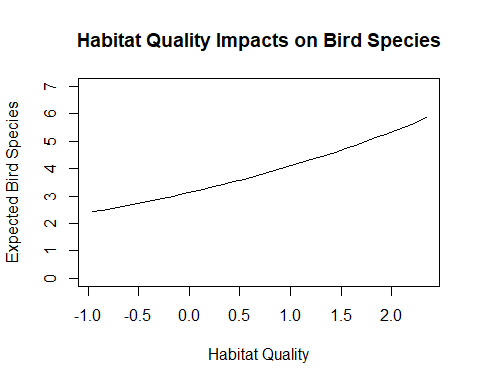
## $fit  
## 1 2 3 4 5 6 7 8   
## 2.425000 2.490660 2.558096 2.627359 2.698498 2.771562 2.846605 2.923679   
## 9 10 11 12 13 14 15 16   
## 3.002840 3.084145 3.167651 3.253418 3.341508 3.431982 3.524906 3.620347   
## 17 18 19 20 21 22 23 24   
## 3.718371 3.819049 3.922454 4.028658 4.137738 4.249771 4.364837 4.483019   
## 25 26 27 28 29 30 31 32   
## 4.604401 4.729070 4.857114 4.988625 5.123696 5.262425 5.404910 5.551253   
## 33 34   
## 5.701559 5.855934   
##   
## $se.fit  
## 1 2 3 4 5 6 7   
## 0.4937330 0.4867922 0.4791690 0.4708410 0.4617868 0.4519868 0.4414240   
## 8 9 10 11 12 13 14   
## 0.4300851 0.4179616 0.4050520 0.3913640 0.3769179 0.3617519 0.3459281   
## 15 16 17 18 19 20 21   
## 0.3295419 0.3127357 0.2957168 0.2787831 0.2623575 0.2470291 0.2335954   
## 22 23 24 25 26 27 28   
## 0.2230835 0.2167091 0.2157262 0.2211702 0.2336055 0.2530401 0.2790567   
## 29 30 31 32 33 34   
## 0.3110421 0.3483775 0.3905381 0.4371222 0.4878440 0.5425136   
##   
## $residual.scale  
## [1] 1

1. Plot these expected values against the range of habitat quality values generated in b). Adjust the y axis to range from 0 to 7 using the ylim argument\*\*. Label your axes appropriately. (2 Pts).

#remember that "fit" is the expected y values  
plot(newx2$Quality, exp.y2$fit, main = "Habitat Quality Impacts on Bird Species", xlab = "Habitat Quality", ylab = "Expected Bird Species", ylim = c(0,7))

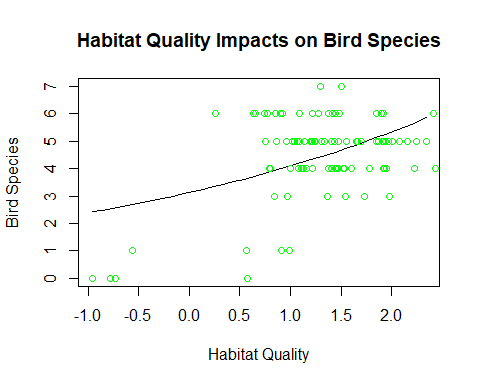


#should it be a line?  
plot(newx2$Quality, exp.y2$fit, main = "Habitat Quality Impacts on Bird Species", xlab = "Habitat Quality", ylab = "Expected Bird Species", ylim = c(0,7), type = "l")



1. Add the observations to the plot generated in c). (1 Pt)

plot(newx2$Quality, exp.y2$fit, main = "Habitat Quality Impacts on Bird Species", xlab = "Habitat Quality", ylab = "Bird Species", ylim = c(0,7), type = "l") #plotted model as a line, data as points  
points(birddat$Quality, birddat$Species, col = "green")



## Question 5 (not an R question): (4 Pts)

Interpret the results from the Poisson regression in Q3. In 2-3 sentences, quantitatively\*\*\* describe the effect of the predictor variable on the response variable (both direction and magnitude; use units); what the null hypothesis is in this context, which decision you make about the null hypothesis, based on what part of the results. Type your answer into your R script; remember to comment it out, since it is not code! Please use line breaks, rather than writing everything into one very long line. \*\*By that I mean, it is not enough to say the effect is positive or negative, strong or weak, you need to use the parameter estimate in your description, and explain what that number means in the context of this example.

## Answer:

The null hypotheses is that there is no relationship between habitat quality and number of bird species present. However, data indicate that habitat quality has a positive relationship (0.2672) with number of bird species present that is highly significant (p = 0.00147). This means that the null hypothesis can be be rejected, and we can conclude that increased habitat quality yields increased abundance of bird species.