BinaryLogisticRegression

SumbarajuAditya

2/7/2021

**Binary logistic regression** Logistic regression is a classification algorithm and it it also called as logic regression. Binary logistic regression models the relationship between a set of predictors and a binary response variable. (Frost, 2021) A binary response has only two possible values, such as Yes (1) or No (0). A binary regression model helps us to understand how changes in the predictor values are associated with changes in the probability of an event occurring. (Bruin 2011)

**regression equation**

A regression equation is a statistical model that determined the specific relationship between the predictor variable and the outcome variable. A model regression equation allows you to predict the outcome with a relatively small amount of error. The mathematical formula of the linear regression can be written as

b0 and b1 are known as the regression beta coefficients or parameters: b0 is the intercept of the regression line; that is the predicted value when x = 0. b1 is the slope of the regression line. e is the error term (also known as the residual errors), the part of y that can be explained by the regression model.

**Replicate the logistic regression example from this link**

<https://stats.idre.ucla.edu/r/dae/logit-regression/> (Data: <https://stats.idre.ucla.edu/stat/data/binary.csv>)

binary\_df <- read.csv("C:/BU/DSC520/assignment\_repo/dsc520/data/binary.csv")  
head(binary\_df)

## admit gre gpa rank  
## 1 0 380 3.61 3  
## 2 1 660 3.67 3  
## 3 1 800 4.00 1  
## 4 1 640 3.19 4  
## 5 0 520 2.93 4  
## 6 1 760 3.00 2

**Dataset Observations**

1. admit - The response variable, admit/don’t admit, it is a binary variable dependent of predictor variable gre,gpa,rank

2. gre- predictor variable

3. gpa - predictor variable

4. rank - predictor variable

5. Rank with 1 have highest prestige and 4 stands lowest and is calculated based on the gpa scores

# get the descriptive of dataset  
summary (binary\_df)

## admit gre gpa rank   
## Min. :0.0000 Min. :220.0 Min. :2.260 Min. :1.000   
## 1st Qu.:0.0000 1st Qu.:520.0 1st Qu.:3.130 1st Qu.:2.000   
## Median :0.0000 Median :580.0 Median :3.395 Median :2.000   
## Mean :0.3175 Mean :587.7 Mean :3.390 Mean :2.485   
## 3rd Qu.:1.0000 3rd Qu.:660.0 3rd Qu.:3.670 3rd Qu.:3.000   
## Max. :1.0000 Max. :800.0 Max. :4.000 Max. :4.000

# Standard deviations  
sapply (binary\_df, sd)

## admit gre gpa rank   
## 0.4660867 115.5165364 0.3805668 0.9444602

## two-way contingency table of categorical outcome and predictors  
xtabs(~admit + rank, data = binary\_df)

## rank  
## admit 1 2 3 4  
## 0 28 97 93 55  
## 1 33 54 28 12

**Logit Model**

1. To estimate the logistic regression model we would be using glm (generalized linear model) function.

2. indicating rank as categorical value

binary\_df$rank <- factor(binary\_df$rank)  
logit\_glm <- glm(admit ~ gre + gpa + rank, data = binary\_df, family = "binomial")  
summary(logit\_glm)

##   
## Call:  
## glm(formula = admit ~ gre + gpa + rank, family = "binomial",   
## data = binary\_df)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.6268 -0.8662 -0.6388 1.1490 2.0790   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -3.989979 1.139951 -3.500 0.000465 \*\*\*  
## gre 0.002264 0.001094 2.070 0.038465 \*   
## gpa 0.804038 0.331819 2.423 0.015388 \*   
## rank2 -0.675443 0.316490 -2.134 0.032829 \*   
## rank3 -1.340204 0.345306 -3.881 0.000104 \*\*\*  
## rank4 -1.551464 0.417832 -3.713 0.000205 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 499.98 on 399 degrees of freedom  
## Residual deviance: 458.52 on 394 degrees of freedom  
## AIC: 470.52  
##   
## Number of Fisher Scoring iterations: 4

From the coefficients it is evident that

1. A unit increase in gre increases the log odds of admissions by 0.002264

2. A unit increase in gpa increases the log odds of being admitted by 0.804038

**confidence intervals**

confint(logit\_glm)

## Waiting for profiling to be done...

## 2.5 % 97.5 %  
## (Intercept) -6.2716202334 -1.792547080  
## gre 0.0001375921 0.004435874  
## gpa 0.1602959439 1.464142727  
## rank2 -1.3008888002 -0.056745722  
## rank3 -2.0276713127 -0.670372346  
## rank4 -2.4000265384 -0.753542605

test for an overall effect of rank using the wald.test function

library(aod)  
wald.test(b = coef(logit\_glm), Sigma = vcov(logit\_glm), Terms = 4:6)

## Wald test:  
## ----------  
##   
## Chi-squared test:  
## X2 = 20.9, df = 3, P(> X2) = 0.00011

The chi-squared test statistic of 20.9, with three degrees of freedom is associated with a p-value of 0.00011 indicating that the overall effect of rank is statistically significant.(Bruin 2011)

**Predicted probabilities** Predicted probabilities can be computed for both categorical and continuous predictor variables. To start with Predicted probabilities we need to create a new dataframe with the values we want the independent variables to take on to create our predictions.

binary\_df1 <- with(binary\_df, data.frame(gre = mean(gre), gpa = mean(gpa), rank = factor(1:4)))  
binary\_df1

## gre gpa rank  
## 1 587.7 3.3899 1  
## 2 587.7 3.3899 2  
## 3 587.7 3.3899 3  
## 4 587.7 3.3899 4

Let’s create a new variable to predict the rank variable using predicted probability (type=“response”)

binary\_df1$rankP <- predict(logit\_glm, newdata = binary\_df1, type = "response")  
binary\_df1

## gre gpa rank rankP  
## 1 587.7 3.3899 1 0.5166016  
## 2 587.7 3.3899 2 0.3522846  
## 3 587.7 3.3899 3 0.2186120  
## 4 587.7 3.3899 4 0.1846684

From the above output we see that

1. The predicted probability of being accepted into a graduate program is 0.5166016 for students from the highest prestige (rank=1) undergraduate institutions

2. 0.1846684 for students from the lowest ranked institutions (rank=4), holding gre and gpa at their means.

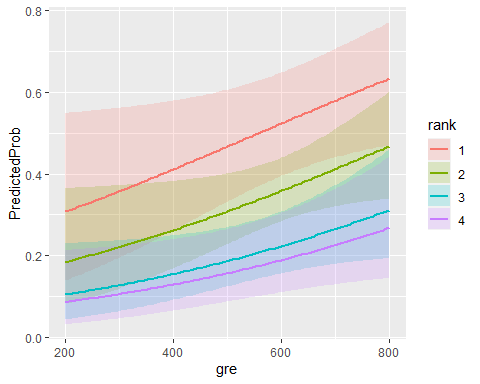
**predicted probabilities varying the value of gre and rank** Let’s try to plot 100 values of gre between 200 and 800, at each value of rank (rank=1, 2, 3, and 4).

binary\_df2 <- with(binary\_df, data.frame(gre = rep(seq(from = 200, to = 800, length.out = 100),  
 4), gpa = mean(gpa), rank = factor(rep(1:4, each = 100))))  
  
binary\_df3 <- cbind(binary\_df2, predict(logit\_glm, newdata = binary\_df2, type = "link",  
 se = TRUE))  
binary\_df3 <- within(binary\_df3, {  
 PredictedProb <- plogis(fit)  
 LL <- plogis(fit - (1.96 \* se.fit))  
 UL <- plogis(fit + (1.96 \* se.fit))  
})  
  
## view first few rows of final dataset  
head(binary\_df3)

## gre gpa rank fit se.fit residual.scale UL LL  
## 1 200.0000 3.3899 1 -0.8114870 0.5147714 1 0.5492064 0.1393812  
## 2 206.0606 3.3899 1 -0.7977632 0.5090986 1 0.5498513 0.1423880  
## 3 212.1212 3.3899 1 -0.7840394 0.5034491 1 0.5505074 0.1454429  
## 4 218.1818 3.3899 1 -0.7703156 0.4978239 1 0.5511750 0.1485460  
## 5 224.2424 3.3899 1 -0.7565919 0.4922237 1 0.5518545 0.1516973  
## 6 230.3030 3.3899 1 -0.7428681 0.4866494 1 0.5525464 0.1548966  
## PredictedProb  
## 1 0.3075737  
## 2 0.3105042  
## 3 0.3134499  
## 4 0.3164108  
## 5 0.3193867  
## 6 0.3223773

**ggplot2**

library(ggplot2)  
ggplot(binary\_df3, aes(x = gre, y = PredictedProb)) + geom\_ribbon(aes(ymin = LL,  
 ymax = UL, fill = rank), alpha = 0.2) + geom\_line(aes(colour = rank),  
 size = 1)



Is our model with predictors fits significantly better than a model with just an intercept? Lets test it.

with(logit\_glm, null.deviance - deviance)

## [1] 41.45903

with(logit\_glm, df.null - df.residual)

## [1] 5

#Calculate P-Value

options(scipen=10000)  
with(logit\_glm, pchisq(null.deviance - deviance, df.null - df.residual, lower.tail = FALSE))

## [1] 0.00000007578194

The chi-square of 41.45903 with 5 degrees of freedom and an associated p-value of less than 0.001 tells us that our model as a whole fits significantly better than an null model. (Bruin 2011)

**References:** \* Introduction to SAS. UCLA: Statistical Consulting Group (Bruin 2011)

Bruin, J. 2011. “Newtest: Command to Compute New Test @ONLINE.” February 2011. <https://stats.idre.ucla.edu/stata/ado/analysis/>.

\* Frost, J. (2021). Binary logistic regression. Retrieved February 08, 2021, from <https://statisticsbyjim.com/glossary/binary-logistic-regression/>