R Notebook: Logistic Regression Examples

Riley Smith

14 November 2016

The Following analyses are based on the R data analysis examples provided by UCLA's Institute for Digital Research and Education.[1]

[1] UCLA: Statistical Consulting Group, "R Data Analysis Examples."

> "Logistic regression, also called a logit model, is used to model dichotomous outcome variables. In the logit model the log odds of the outcome is modeled as a linear combination of the predictor variables."
>
> -[2]

[2] UCLA: Statistical Consulting Group, "R Data Analysis Examples."

The Logistic Equation

$$\ln\left(\frac{\pi}{1-\pi}\right) = \alpha + \beta X$$

$$\pi = \frac{\epsilon^{\alpha+\beta X}}{1+\epsilon^{\alpha+\beta X}}$$

π: $p(Y = 1)$
$1 - \pi$: $p(Y = 0)$

Logistic Equation Intercept

$$\pi = \frac{\epsilon^{\alpha}}{1+\epsilon^{\alpha}}$$

The Logistic Intercept: Estimate of $Y = 1$ when $X = 0$

Example Data Description

> "[Suppose] a researcher is interested in how variables, such as GRE (Graduate Record Exam scores), GPA (grade point average) and prestige of the undergraduate institution, effect admission into graduate school. The response variable, admit/don't admit, is a binary variable.
>
> This dataset has a binary response (outcome, dependent) variable called admit. There are three predictor variables: gre, gpa and rank. We will treat the variables gre and gpa as continuous. The variable rank takes on the values 1 through 4. Institutions with a rank of 1 have the highest prestige, while those with a rank of 4 have the lowest."
>
> -[3]

[3] UCLA: Statistical Consulting Group, "R Data Analysis Examples."

```r
dat <- read.csv("http://www.ats.ucla.edu/stat/data/binary.csv")
R.msmm(dat) ## Summary of numeric variables ##
```

	M	SD	Min	Max	NAs
admit	0.32	0.47	0	1	0
gre	587.7	115.5	220	800	0
gpa	3.39	0.38	2.26	4	0
rank	2.48	0.94	1	4	0

```r
xtabs(~ admit + rank, data = dat) ## cross-taulation ##
```

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

Example Data Analysis: The Logit Model

```r
dat$rank <- factor(dat$rank)
lrm <- glm(admit ~ gre + gpa + rank, data = dat, family = "binomial")
```

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-3.9900	1.1400	-3.50	0.0005
gre	0.0023	0.0011	2.07	0.0385
gpa	0.8040	0.3318	2.42	0.0154
rank2	-0.6754	0.3165	-2.13	0.0328
rank3	-1.3402	0.3453	-3.88	0.0001
rank4	-1.5515	0.4178	-3.71	0.0002

Table 3: Logistic Regression Model Fit Statistics

	Estimate	Degrees of Freedom
Null Deviance	499.98	399
Residual Deviance	458.52	394
AIC	470.52	

Table 4: Logistic Regression: Deviance Residuals Summary [1]

M	-0.11
SD	1.07
Min	-1.63
Max	2.08

Note:
[1] ***M*** *= Mean,* ***SD*** *= Standard Deviation,* ***Min*** *= Minimum, &* ***Max*** *= Maximimum*

Deviance residuals: A measure of model fit.

The summary tables above provide the *model coefficients* with corresponding *standard errors*, *(Wald) z-statistics*, and *p-values*; as well as summary statistics for the distribution of *deviance residuals* for individual cases used in computing the logistic regression model.[4]

[4] UCLA: Statistical Consulting Group, "R Data Analysis Examples."

Logistic regression coefficients ($\beta's$): Level of change in the log odds of the outcome per one unit increase in the predictor.

Logit Model: Confidence Intervals (CI) & Odds Ratios (OR)

Confidence Intervals

```
## CIs using profiled log-likelihood ##
CI <- confint(lrm)
```

```r
## CIs using standard errors ##
CI.se <- confint.default(lrm)
```

Odds Ratio

```r
library(aod) ## "wald.test()" ##
wald.test(b = coef(lrm), Sigma = vcov(lrm), Terms = 4:6)
```

Wald test:

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

```r
l <- cbind(0,0,0,1,-1,0)
wald.test(b = coef(lrm), Sigma = vcov(lrm), L = l)
```

Wald test:

Chi-squared test:
X2 = 5.5, df = 1, P(> X2) = 0.019

```r
## odds ratios##
OR <- exp(coef(lrm))
```

Table 5: Logistic Regression Odds Ratios (Φ) & Confidence Intervals (*CI*) [1]

		CI	
	Φ	2.5 %	97.5 %
(Intercept)	0.0185	0.0019	0.1665
gre	1.0023	1.0001	1.0044
gpa	2.2345	1.1739	4.3238
rank2	0.5089	0.2723	0.9448
rank3	0.2618	0.1316	0.5115
rank4	0.2119	0.0907	0.4707

Note:
[1] Confidence intervals are based on the logistic regression
model's profiled log-likelihood function,
rather than the standard errors

```
dat.p1 <- with(dat,
  data.frame(gre = mean(gre), gpa = mean(gpa), rank = factor(1:4)))
dat.p1
```

gre	gpa	rank
587.7	3.39	1
587.7	3.39	2
587.7	3.39	3
587.7	3.39	4

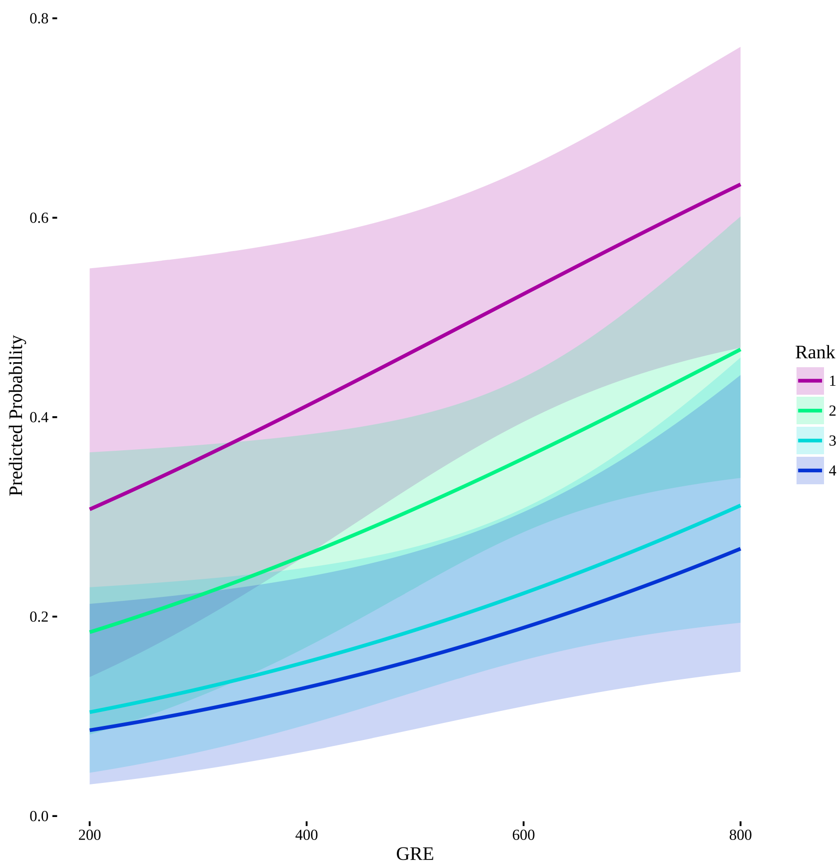
```
dat.p1$rankP <- predict(lrm, newdata = dat.p1, type = "response")
dat.p1
```

gre	gpa	rank	rankP
587.7	3.39	1	0.5166
587.7	3.39	2	0.3523
587.7	3.39	3	0.2186
587.7	3.39	4	0.1847

```
dat.p2 <- with(dat,
  data.frame(gre = rep(seq(from = 200, to = 800, length.out = 100), 4),
  gpa = mean(gpa), rank = factor(rep(1:4, each = 100))))

dat.p3 <- cbind(dat.p2, predict(lrm, newdata = dat.p2, type = "link", se = TRUE))

dat.p3 <- within(dat.p3, {
  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(dat.p3)
```

gre	gpa	rank	fit	se.fit	residual.scale	UL	LL	PredictedProb
200.0	3.39	1	-0.8115	0.5148	1	0.5492	0.1394	0.3076
206.1	3.39	1	-0.7978	0.5091	1	0.5499	0.1424	0.3105
212.1	3.39	1	-0.7840	0.5034	1	0.5505	0.1454	0.3134
218.2	3.39	1	-0.7703	0.4978	1	0.5512	0.1485	0.3164
224.2	3.39	1	-0.7566	0.4922	1	0.5519	0.1517	0.3194
230.3	3.39	1	-0.7429	0.4866	1	0.5525	0.1549	0.3224

```
ggplot(dat.p3, aes(x = gre, y = PredictedProb)) + theme_tufte() +
    geom_ribbon(aes(ymin = LL, ymax = UL, fill = rank), alpha = .2) +
    geom_line(aes(colour = rank), size = 1) +
    scale_colour_manual(values = cols2(4)) +
    scale_fill_manual(values = cols2(4)) +
    labs(x = "GRE", y = "Predicted Probability", colour = "Rank", fill = "Rank")
```



References

UCLA: Statistical Consulting Group. "R Data Analysis Examples: Logit Regression," 2016. http://www.ats.ucla.edu/stat/r/dae/logit.htm.