STAT 528 - Advanced Regression Analysis II

Multinomial response regression (part I)

Daniel J. Eck
Department of Statistics
University of Illinois

Last time

- nominal responses
- Multinomial regression via baseline-category logistic model
- data analysis

Learning Objectives Today

- ordinal responses
- proportional-odds model
- data analysis

This slide deck will only contain data analysis. The lecture will be largely on the blackboard.

R Example: Happiness and Traumatic Events

The response variable happiness is an ordinal categorical variable indicating the current happiness level of the individual:

- 1 if very happy
- 2 if pretty happy
- 3 if not too happy

Here trauma is a count of the number of traumatic events that the individual faced in the previous year.

The control variable is a binary categorical variable (race) that was deemed important by the researchers who conducted the study (given two levels: 0 if in A; 1 if in B).

We load in the data

```r
happiness <- read.table("happiness.txt", header=TRUE)
```

and display the first 10 rows:

```r
head(happiness, 10)
```

```
##    control trauma happy
## 1        0      0     1
## 2        0      0     1
## 3        0      0     1
## 4        0      0     1
## 5        0      0     1
## 6        0      0     1
## 7        0      0     1
## 8        0      0     2
## 9        0      0     2
## 10       0      0     2
```

We load in VGAM and fit the proportional-odds model

```r
library(VGAM)
mod <- vglm(happy ~ trauma + control, family=propodds(reverse=FALSE),
            data=happiness)
summary(mod)
```

```
##
## Call:
## vglm(formula = happy ~ trauma + control, family = propodds(reverse = FALSE),
##     data = happiness)
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept):1  -0.5181     0.3382  -1.532  0.12552
## (Intercept):2   3.4006     0.5648   6.021 1.74e-09 ***
## trauma         -0.4056     0.1809  -2.242  0.02493 *
## control        -2.0361     0.6911  -2.946  0.00322 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Names of linear predictors: logitlink(P[Y<=1]), logitlink(P[Y<=2])
##
## Residual deviance: 148.407 on 190 degrees of freedom
##
## Log-likelihood: -74.2035 on 190 degrees of freedom
##
## Number of Fisher scoring iterations: 5
##
## No Hauck-Donner effect found in any of the estimates
##
##
## Exponentiated coefficients:
##    trauma   control
## 0.6665934 0.1305338
```

A LRT suggests that our model fits the data better than a saturated model.

```
pchisq(deviance(mod), df.residual(mod), lower = FALSE)
```

```
## [1] 0.9886371
```

In our notation, the estimates are

$$\begin{array}{c} \hat{\alpha}_1 \approx -0.518 \\ \hat{\alpha}_2 \approx 3.401 \end{array} \qquad \hat{\beta} \approx \begin{pmatrix} -0.406 \\ -2.036 \end{pmatrix}$$

where

$$x = \begin{pmatrix} \text{trauma} \\ \text{control} \end{pmatrix}.$$

Thus happiness is estimated lower (Y is estimated to be larger) as trauma increases.

We can estimate the odds of "very happy'' for control category A (coded 0) relative to control category B (coded 1) with trauma held fixed, and a Wald 95% confidence interval for these estimates:

```
exp(2.036)
```

```
## [1] 7.659908
```

```
exp(2.036 + c(-1,1) * qnorm(0.975) * 0.691)
```

```
## [1]  1.977167 29.675895
```

The control variable (race) has a large effect on happiness

We can do likelihood ratio tests as before

```
modred <- vglm(happy ~ trauma, family=propodds(reverse=FALSE),
               data=happiness)
llrts <- deviance(modred) - deviance(mod)
llrts.df <- df.residual(modred) - df.residual(mod)
llrts
```

```
## [1] 9.231169
```

```
llrts.df
```

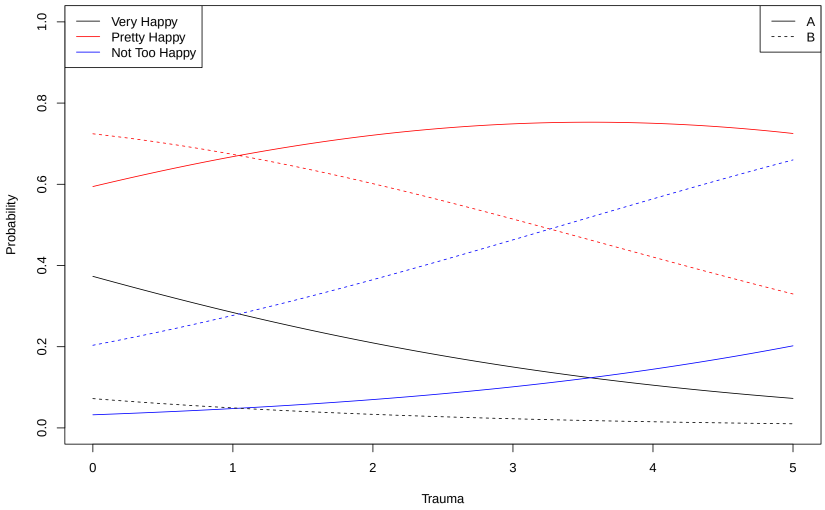
```
## [1] 1
```

```
1 - pchisq(llrts, llrts.df)
```

```
## [1] 0.002379297
```

We can also graph probability curves (versus trauma) by happiness category and control:

```
curve(predict(mod, data.frame(trauma=x,control=0), type="response")[,1],
      xlab="Trauma", ylab="Probability",
      xlim=range(happiness$trauma), ylim=c(0,1))
curve(predict(mod, data.frame(trauma=x,control=0), type="response")[,2],
      add=TRUE, col="red")
curve(predict(mod, data.frame(trauma=x,control=0), type="response")[,3],
      add=TRUE, col="blue")
curve(predict(mod, data.frame(trauma=x,control=1), type="response")[,1],
      add=TRUE, lty=2)
curve(predict(mod, data.frame(trauma=x,control=1), type="response")[,2],
      add=TRUE, col="red", lty=2)
curve(predict(mod, data.frame(trauma=x,control=1), type="response")[,3],
      add=TRUE, col="blue", lty=2)
legend("topright", c("A","B"), lty=1:2)
legend("topleft", c("Very Happy","Pretty Happy","Not Too Happy"), lty=1,
       col=c("black","red","blue"))
```


Very Happy
Pretty Happy
Not Too Happy
A
B
Probability
0.0
0.2
0.4
0.6
0.8
1.0
0
1
2
3
4
5
Trauma

We can check the assumption of proportional odds by comparison with a model that does not assume it:

```
modnotprop <- vglm(happy ~ trauma + control, family=cumulative(parallel=FALSE),
                   data=happiness)
summary(modnotprop)
```

```
##
## Call:
## vglm(formula = happy ~ trauma + control, family = cumulative(parallel = FALSE),
##     data = happiness)
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept):1   -0.5661     0.3662  -1.546   0.1221
## (Intercept):2    3.4837     0.7595   4.587  4.5e-06 ***
## trauma:1        -0.3409     0.2124  -1.605   0.1086
## trauma:2        -0.4836     0.2752  -1.757   0.0789 .
## control:1      -16.8922  1358.1457      NA       NA
## control:2       -1.8467     0.7628  -2.421   0.0155 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Names of linear predictors: logitlink(P[Y<=1]), logitlink(P[Y<=2])
##
## Residual deviance: 146.9951 on 188 degrees of freedom
##
## Log-likelihood: -73.4976 on 188 degrees of freedom
##
## Number of Fisher scoring iterations: 17
##
## Warning: Hauck-Donner effect detected in the following estimate(s):
## '(Intercept):2', 'control:1'
##
##
## Exponentiated coefficients:
```

Now perform the LRT. Keep in mind that forcing proportionality is more restrictive than not enforcing it.

```
llrts <- deviance(mod) - deviance(modnotprop)
llrts.df <- df.residual(mod) - df.residual(modnotprop)
llrts
```

```
## [1] 1.411892
```

```
llrts.df
```

```
## [1] 2
```

```
1 - pchisq(llrts, llrts.df)
```

```
## [1] 0.4936413
```

The proportional-odds model fits this data better.

We can also fit a probit analog to the proportional-odds model

```
mod.probit <- vglm(happy ~ trauma + control,
                   family=cumulative(link="probitlink",parallel=TRUE),
                   data=happiness)
summary(mod.probit)
```

```
##
## Call:
## vglm(formula = happy ~ trauma + control, family = cumulative(link = "probitlink",
##     parallel = TRUE), data = happiness)
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept):1 -0.34808    0.20015  -1.739  0.08201 .
## (Intercept):2  1.91607    0.28287   6.774 1.26e-11 ***
## trauma        -0.22131    0.09897  -2.236  0.02535 *
## control       -1.15712    0.37872  -3.055  0.00225 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Names of linear predictors: probitlink(P[Y<=1]), probitlink(P[Y<=2])
##
## Residual deviance: 148.1066 on 190 degrees of freedom
##
## Log-likelihood: -74.0533 on 190 degrees of freedom
##
## Number of Fisher scoring iterations: 5
##
## No Hauck-Donner effect found in any of the estimates
##
##
## Exponentiated coefficients:
##    trauma   control
## 0.8014668 0.3143908
```

See notes `polr` implementation.