November 9, 2018

An analysis of the happiness data

Dennis D. Cox
Department of Statistics
Rice University

November 9, 2018

1 Example of Ordinal Logistic Regression

We do some analysis of the Happiness data set described in section 8.2.4, page 304 of the text. The response `happy` is ordinal with 3 levels, and there are 2 predictor variables: `race`, which is binary; and `trauma`, which is a count. One thing that is unclear to me is if `trauma` should be transformed or not. Perhaps more traumas leads a person to become less affected by each individual trauma, or it could work the other way.

The data were taken from the book website and put in a file `data.txt`. We did several analyses, and based on the results, it seems trauma needs a quadratic term for trauma, suggesting more traumas makes each individual trauma even worse, although this is would be a highly questionable conclusion. Another possibility is that people who experience more traumas are actually experiencing worse traumas. For example, being injured in an automobile accident would count as one trauma, having a parent die and a sibling murdered would count as two traumas, and I suspect the person suffering those two traumas would score much lower in happiness than the person suffering from the automobile accident.

The transcript of the R-code and the plots are given below.

```
R version 3.3.1 (2016-06-21) -- "Bug in Your Hair"
Copyright (C) 2016 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin13.4.0 (64-bit)
```

```
R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
> happiness = matrix(scan("data.txt"),ncol=3,byrow=T)
Read 291 items
> colnames(happiness) = c("race","trauma","happy")
> happiness[1:4,]
     race trauma happy
[1,]    0      0     1
[2,]    0      0     1
[3,]    0      0     1
[4,]    0      0     1
> apply(happiness,2,table)
$race

 0  1
84 13

$trauma

 0  1  2  3  4  5
25 25 26 14  5  2

$happy

 1  2  3
22 65 10

> race = happiness[,1]
> trauma = happiness[,2]
> happy = happiness[,3]
> # loading VGAM package
Loading required package: stats4
Loading required package: splines
```

```
> fit1 = vglm(happy ~ race + trauma,cumulative(link = logit, parallel = TRUE))
> summary(fit1)

Call:
vglm(formula = happy ~ race + trauma, family = cumulative(link = logit,
    parallel = TRUE))

Pearson residuals:
                    Min      1Q  Median      3Q    Max
logit(P[Y<=1]) -0.7981 -0.6575 -0.4493 -0.1538 2.9096
logit(P[Y<=2]) -5.4716  0.1523  0.2537  0.3040 0.9543

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 ***
race           -2.0361     0.6911  -2.946  0.00322 **
trauma         -0.4056     0.1809  -2.242  0.02493 *
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Number of linear predictors:  2

Names of linear predictors: logit(P[Y<=1]), logit(P[Y<=2])

Dispersion Parameter for cumulative family:   1

Residual deviance: 148.407 on 190 degrees of freedom

Log-likelihood: -74.2035 on 190 degrees of freedom

Number of iterations: 5

Exponentiated coefficients:
     race    trauma
0.1305338 0.6665934
> # trauma is significant.
> ##################################################################
```

```
> # Is there a transformation of trauma which gives a better fit?
> trauma.5 = sqrt(trauma)
> trauma2 = trauma^2
> fit2 = vglm(happy ~ race + trauma + trauma.5 + trauma2,
+ cumulative(link = logit, parallel = TRUE))
> summary(fit2)

Call:
vglm(formula = happy ~ race + trauma + trauma.5 + trauma2, family = cumulative(lin
    parallel = TRUE))

Pearson residuals:
                   Min      1Q  Median      3Q   Max
logit(P[Y<=1]) -0.7194 -0.6540 -0.4625 -0.1056 4.260
logit(P[Y<=2]) -5.5041  0.1408  0.2255  0.2263 0.972

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept):1  -0.9198     0.4319  -2.130  0.03318 *
(Intercept):2   3.2135     0.6366   5.048 4.47e-07 ***
race           -2.3826     0.7443  -3.201  0.00137 **
trauma          0.6951     2.0608   0.337  0.73587
trauma.5       -0.2272     2.2271  -0.102  0.91874
trauma2        -0.2692     0.2599  -1.036  0.30029
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Number of linear predictors:  2

Names of linear predictors: logit(P[Y<=1]), logit(P[Y<=2])

Dispersion Parameter for cumulative family:   1

Residual deviance: 143.9628 on 188 degrees of freedom

Log-likelihood: -71.9814 on 188 degrees of freedom

Number of iterations: 5
```

```
Exponentiated coefficients:
      race     trauma   trauma.5    trauma2
0.09230765 2.00398637 0.79675295 0.76402003
> # that didn't accomplish much
> # none of the trauma coefficients are significant
> # what about the inclusion of all the trauma variables?
> fit0 = vglm(happy ~ race,cumulative(link = logit, parallel = TRUE))
> summary(fit0)

Call:
vglm(formula = happy ~ race, family = cumulative(link = logit,
    parallel = TRUE))

Pearson residuals:
                  Min      1Q  Median      3Q    Max
logit(P[Y<=1]) -0.620 -0.6200 -0.6200 -0.1507 1.6940
logit(P[Y<=2]) -3.795  0.1686  0.2979  0.2979 0.7468

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept):1  -1.0640     0.2479  -4.292 1.77e-05 ***
(Intercept):2   2.6714     0.4261   6.269 3.63e-10 ***
race           -2.0016     0.6817  -2.936  0.00332 **
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Number of linear predictors:  2

Names of linear predictors: logit(P[Y<=1]), logit(P[Y<=2])

Dispersion Parameter for cumulative family:   1

Residual deviance: 153.475 on 191 degrees of freedom

Log-likelihood: -76.7375 on 191 degrees of freedom

Number of iterations: 4
```

```
Exponentiated coefficients:
     race
0.1351195
> # compute p-value for LRT statistic of null hypothesis H0: all 3 trauma variable
> pchisq(2*(76.7375-71.9814),191-188,lower.tail=FALSE)
[1] 0.02320193
> # about the same p-value as just including the (original) trauma variable
> ##################################################################
> # lets try each of them one-at-a-time
> fit3 = vglm(happy ~ race + trauma.5,cumulative(link = logit, parallel = TRUE))
> summary(fit3)

Call:
vglm(formula = happy ~ race + trauma.5, family = cumulative(link = logit,
    parallel = TRUE))

Pearson residuals:
                   Min      1Q  Median      3Q    Max
logit(P[Y<=1]) -0.7736 -0.6110 -0.5152 -0.1508 2.2022
logit(P[Y<=2]) -5.0046  0.1653  0.2889  0.3166 0.8489

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept):1  -0.5901     0.3856  -1.530  0.12592
(Intercept):2   3.2226     0.5669   5.685 1.31e-08 ***
race           -1.9639     0.6844  -2.870  0.00411 **
trauma.5       -0.4979     0.3195  -1.559  0.11910
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Number of linear predictors:  2

Names of linear predictors: logit(P[Y<=1]), logit(P[Y<=2])

Dispersion Parameter for cumulative family:   1

Residual deviance: 151.0682 on 190 degrees of freedom
```

```
Log-likelihood: -75.5341 on 190 degrees of freedom

Number of iterations: 5

Exponentiated coefficients:
     race  trauma.5
0.1403094 0.6078068
> fit4 = vglm(happy ~ race + trauma2,cumulative(link = logit, parallel = TRUE))
> summary(fit4)

Call:
vglm(formula = happy ~ race + trauma2, family = cumulative(link = logit,
    parallel = TRUE))

Pearson residuals:
                   Min      1Q  Median      3Q   Max
logit(P[Y<=1]) -0.7489 -0.7028 -0.4258 -0.1091 3.969
logit(P[Y<=2]) -5.5859  0.1371  0.2173  0.2593 1.016

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept):1 -0.64046    0.28295  -2.263  0.02361 *
(Intercept):2  3.44172    0.56343   6.109 1.01e-09 ***
race          -2.23798    0.71908  -3.112  0.00186 **
trauma2       -0.13243    0.04582  -2.890  0.00385 **
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Number of linear predictors:  2

Names of linear predictors: logit(P[Y<=1]), logit(P[Y<=2])

Dispersion Parameter for cumulative family:   1

Residual deviance: 145.0857 on 190 degrees of freedom

Log-likelihood: -72.5429 on 190 degrees of freedom
```

```
Number of iterations: 5

Exponentiated coefficients:
     race   trauma2
0.1066742 0.8759614
> # this last one seems to be the best fit -
> # judging from p-values, which is not really a great approach
> ########################################################################
> # we could see what gives the best predictive model by cross-validation
> # thats a lot of work
> # lets see what the ace function says
> fit.ace = ace(cbind(race,trauma),happy,mon=c(0,2))
> # this forces the transformations of trauma and happy to be monotone
> plot(trauma,fit.ace$tx[,2])
> # looks very quadratic
> # suggests that we should use a quadratic polynomial
> title(main="ACE estimated transform of trauma")
> plot(happy,fit.ace$ty)
> title(main="ACE estimated transform of happy")
> # looks very linear
> fit.final = vglm(happy ~ race + trauma + trauma2,cumulative(link = logit, parall
> summary(fit.final)

Call:
vglm(formula = happy ~ race + trauma + trauma2, family = cumulative(link = logit,
    parallel = TRUE))

Pearson residuals:
                   Min      1Q  Median      3Q    Max
logit(P[Y<=1]) -0.7302 -0.6492 -0.4574 -0.1048 4.2661
logit(P[Y<=2]) -5.5882  0.1414  0.2272  0.2273 0.9823

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept):1  -0.9353     0.4053  -2.308  0.02100 *
(Intercept):2   3.1973     0.6154   5.196 2.04e-07 ***
race           -2.3800     0.7430  -3.203  0.00136 **
```

```
trauma          0.4908     0.4690   1.046  0.29534
trauma2        -0.2457     0.1184  -2.075  0.03801 *
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Number of linear predictors:  2

Names of linear predictors: logit(P[Y<=1]), logit(P[Y<=2])

Dispersion Parameter for cumulative family:   1

Residual deviance: 143.9732 on 189 degrees of freedom

Log-likelihood: -71.9866 on 189 degrees of freedom

Number of iterations: 5

Exponentiated coefficients:
      race     trauma    trauma2
0.09255419 1.63365113 0.78216602
> # I think this is the best model, although it was discovered with some data snoo
```

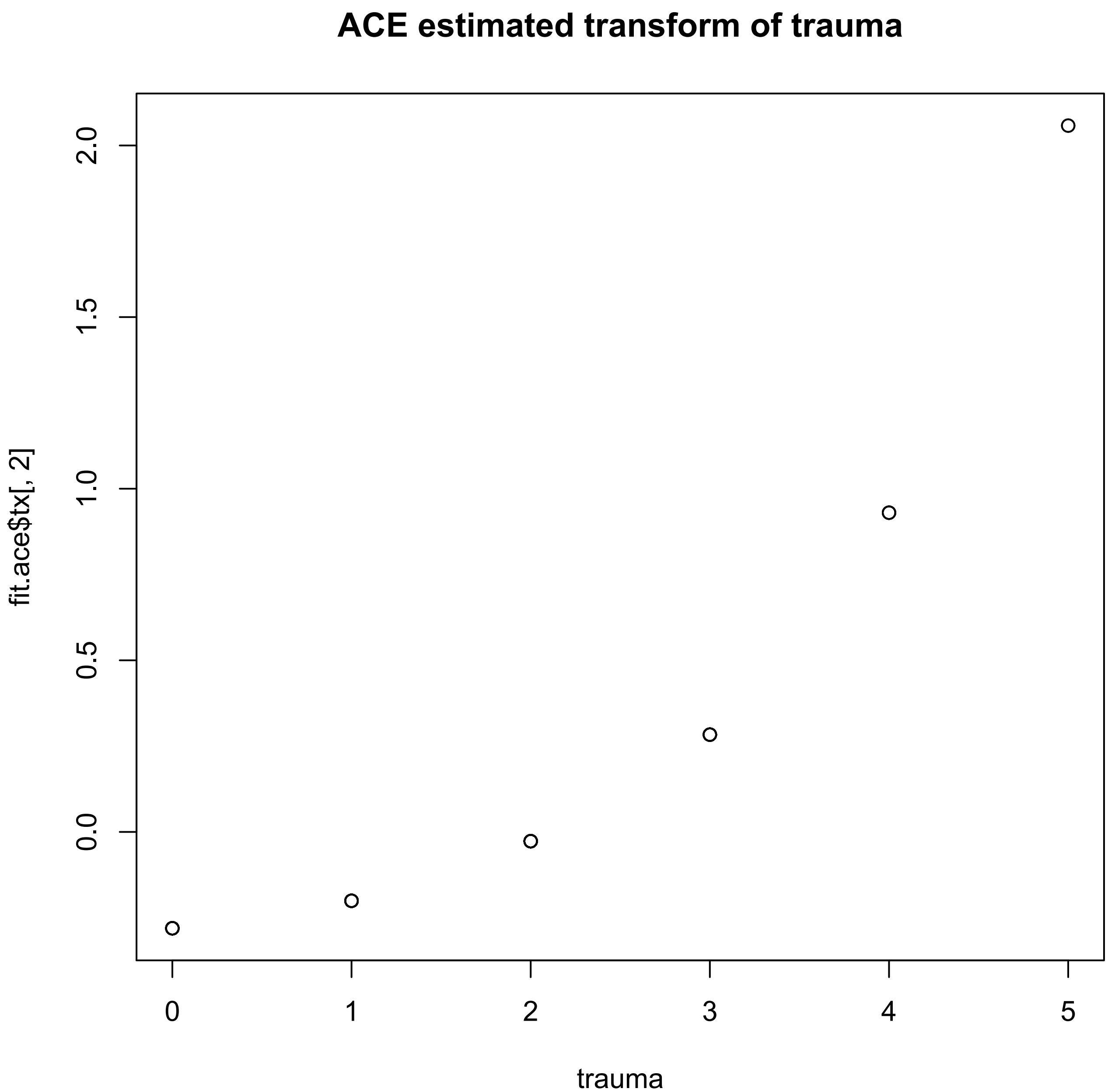

Figure 1: Plot of ACE transformation for trauma variable.

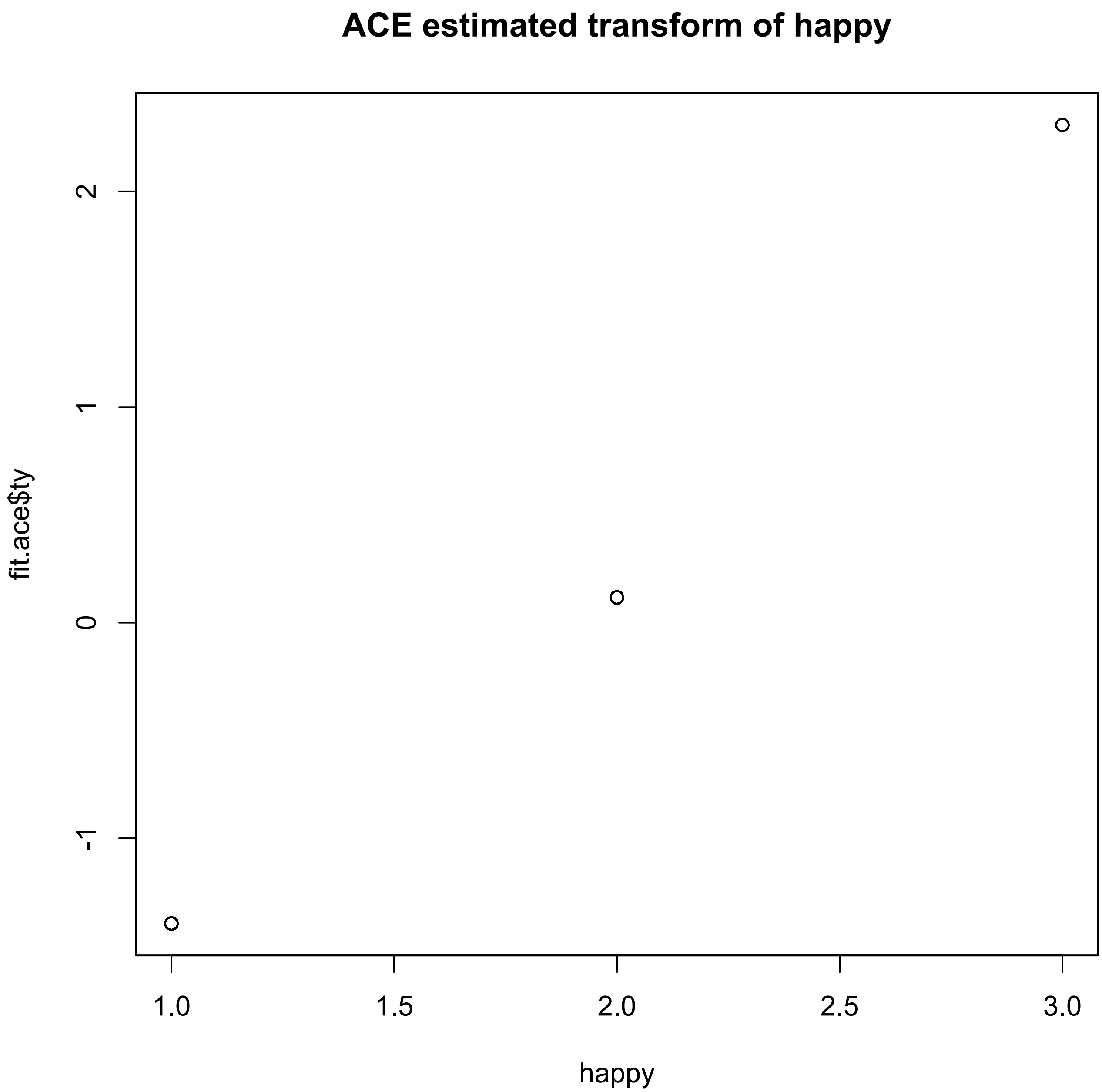

Figure 2: Plot of ACE transformation for happy.