432 Class 14 Slides

github.com/THOMASELOVE/2020-432

2020-03-05

Setup

```
library(here); library(magrittr); library(janitor)
library(skimr)
library(rms)
library(aplore3) # for a data set
library(ResourceSelection) # for Hosmer-Lemeshow test
library(broom)
library(tidyverse)

colscr <- read.csv(here("data/screening.csv")) %>% tbl_df
colscr2 <- read.csv(here("data/screening2.csv")) %>% tbl_df
```

Today's Materials

- Logistic Regression
  - on Aggregated Data
  - and describing restricted cubic splines
- Probit Regression: A Useful Alternative Link

Logistic Regression for Aggregated Data

Colorectal Cancer Screening Data

The `screening.csv` data (imported into the R tibble `colscr` are simulated. They mirror a subset of the actual results from the Better Health Partnership's original pilot study of colorectal cancer screening in primary care clinics in Northeast Ohio.

Available to us are the following variables

Variable	Description
`location`	clinic code
`subjects`	number of subjects reported by clinic
`screen_rate`	proportion of `subjects` who were screened
`screened`	number of `subjects` who were screened
`notscreened`	number of `subjects` not screened
`meanage`	mean age of clinic's subjects, years
`female`	% of clinic's subjects who are female
`pct_lowins`	% of clinic's subjects who have Medicaid or are uninsured
`system`	system code

Skim results

```
Skim summary statistics
 n obs: 26
 n variables: 9

Variable type: factor
 variable missing complete  n n_unique                         top_counts ordered
 location       0       26 26       26         A: 1, B: 1, C: 1, D: 1   FALSE
   system       0       26 26        4 Sys: 7, Sys: 7, Sys: 6, Sys: 6   FALSE

Variable type: integer
    variable missing complete  n    mean      sd  p0     p25 median     p75 p100     hist
 notscreened       0       26 26  663.23  271.17 231  508.75   611     791   1356
    screened       0       26 26 2584.04 1765.11 572 1395.25 2169.5 2716    6947
    subjects       0       26 26 3247.27 1945.83 803 1914.75 2765.5 3607.75 7677

Variable type: numeric
    variable missing complete  n  mean     sd    p0    p25 median   p75 p100     hist
      female       0       26 26 58.72  6.29  46.2  55.42  60.05 62.62 70.3
     meanage       0       26 26 60.58  1.93  58    58.82  60.5  61.98 65.9
  pct_lowins       0       26 26 24.47 19.13   0.3   4.8   23.95 44.03 51.3
 screen_rate       0       26 26  0.77  0.072  0.64  0.72   0.76  0.81  0.9
```

Fitting a Logistic Regression Model to Proportion Data

Here, we have a binary outcome (was the subject screened or not?) but we have aggregated results. We can use the counts of the numbers of subjects at each clinic (in `subjects`) and the proportion who were screened (in `screen_rate`) to fit a logistic regression model, as follows:

```
m_screen1 <-  glm(screen_rate ~ meanage + female +
                     pct_lowins + system, family = binomial,
                   weights = subjects, data = colscr)
```

tidy(m_screen1)

```
# A tibble: 7 x 5
  term        estimate std.error statistic  p.value
  <chr>          <dbl>     <dbl>     <dbl>    <dbl>
1 (Intercept)  -1.33    0.553        -2.40  1.64e- 2
2 meanage       0.0680  0.00898       7.57  3.60e-14
3 female       -0.0193  0.00158     -12.2   3.10e-34
4 pct_lowins   -0.0135  0.000859    -15.7   2.36e-55
5 systemSys_2  -0.138   0.0247       -5.61  2.08e- 8
6 systemSys_3  -0.0400  0.0255       -1.57  1.16e- 1
7 systemSys_4   0.0229  0.0294        0.779 4.36e- 1
```

Fitting Counts of Successes and Failures

```
m_screen2 <-  glm(cbind(screened, notscreened) ~
                    meanage + female + pct_lowins + system,
           family = binomial, data = colscr)
```

tidy(m_screen2)

```
# A tibble: 7 x 5
  term        estimate std.error statistic  p.value
  <chr>          <dbl>     <dbl>     <dbl>    <dbl>
1 (Intercept)  -1.33    0.553       -2.40  1.64e- 2
2 meanage       0.0680  0.00898      7.57  3.60e-14
3 female       -0.0193  0.00158    -12.2   3.10e-34
4 pct_lowins   -0.0135  0.000859   -15.7   2.36e-55
5 systemSys_2  -0.138   0.0247      -5.61  2.08e- 8
6 systemSys_3  -0.0400  0.0255      -1.57  1.16e- 1
7 systemSys_4   0.0229  0.0294       0.779 4.36e- 1
```

How does one address this problem in `rms`?

We can use `Glm`.

```
d <- datadist(colscr)
options(datadist = "d")

mod_screen_1 <-  Glm(screen_rate ~ meanage + female +
                         pct_lowins + system,
                     family = binomial, weights = subjects,
                     data = colscr, x = T, y = T)
```

mod_screen_1

```
General Linear Model

 Glm(formula = screen_rate ~ meanage + female + pct_lowins + system,
     family = binomial, data = colscr, weights = subjects, x = T,
     y = T)

                         Model Likelihood
                            Ratio Test
 Obs        26          LR chi2     2008.90
 Residual d.f.19        d.f.              6
 g 0.4614539            Pr(> chi2) <0.0001

              Coef    S.E.    Wald Z Pr(>|Z|)
 Intercept    -1.3270 0.5531  -2.40 0.0164
 meanage       0.0680 0.0090   7.57 <0.0001
 female       -0.0193 0.0016 -12.20 <0.0001
 pct_lowins   -0.0135 0.0009 -15.67 <0.0001
 system=Sys_2 -0.1382 0.0247  -5.61 <0.0001
 system=Sys_3 -0.0400 0.0255  -1.57 0.1159
 system=Sys_4  0.0229 0.0294   0.78 0.4358
```

Using Restricted Cubic Splines

Explaining a Model with a Restricted Cubic Spline

Restricted cubic splines are an easy way to include an explanatory variable in a smooth and non-linear fashion in your model.

- The number of knots, k, are specified in advance, and this is the key issue to determining what the spline will do. We could use AIC to select k, or follow the general idea that for small n, k should be 3, for large n, k should be 5, and so often $k = 4$.
- The location of those knots is not important in most situations, so R places knots by default where the data exist, at fixed quantiles of the predictor's distribution.
- The "restricted" piece means that the tails of the spline (outside the outermost knots) behave in a linear fashion.

The “Formula” from a Model with a Restricted Cubic Spline

- The best way to demonstrate what a spline does is to draw a picture of it. When in doubt, do that: show us how the spline affects the predictions made by the model.
- But you can get a model equation for the spline out of R (heaven only knows what you would do with it.) Use the `latex` function in the `rms` package, for instance.

An Example

```
d <- datadist(iris)
options(datadist = "d")
m1 <- ols(Sepal.Length ~ rcs(Petal.Length, 4) + Petal.Width,
          data = iris, x = TRUE, y = TRUE)
```

m1

```
Linear Regression Model

 ols(formula = Sepal.Length ~ rcs(Petal.Length, 4) + Petal.Width,
     data = iris, x = TRUE, y = TRUE)

                Model Likelihood     Discrimination
                   Ratio Test           Indexes
 Obs       150    LR chi2     253.23    R2       0.815
 sigma0.3609    d.f.             4    R2 adj   0.810
 d.f.      145    Pr(> chi2) 0.0000    g        0.844

 Residuals

       Min         1Q    Median         3Q       Max
 -1.002936 -0.245788  0.009911  0.208848  0.852141


                Coef    S.E.   t     Pr(>|t|)
 Intercept        4.7226 0.1809 26.11 <0.0001
 Petal.Length     0.2434 0.1144  2.13 0.0351
 Petal.Length'    0.5018 0.2921  1.72 0.0880
 Petal.Length'' -0.8730 1.1334 -0.77 0.4424
 Petal.Width    -0.3340 0.1498 -2.23 0.0273
```

Function(m1)

```
Function(m1)
```

```
function (Petal.Length = 4.35, Petal.Width = 1.3)
{
    4.7226352 + 0.24335435 * Petal.Length + 0.021780541 * pmax
        1.3, 0)^3 - 0.037888523 * pmax(Petal.Length - 3.33, 0)
        0.00031123969 * pmax(Petal.Length - 4.8, 0)^3 + 0.0157
        pmax(Petal.Length - 6.1, 0)^3 - 0.33400958 * Petal.Wic
}
<environment: 0x000000001e0fed50>
```

What's in `Function(m1)`?

```
4.72 + 0.243  * Petal.Length
     + 0.022  * pmax( Petal.Length-1.3,  0)^3
     - 0.038  * pmax( Petal.Length-3.33, 0)^3
     + 0.0003 * pmax( Petal.Length-4.8,  0)^3
     + 0.016  * pmax( Petal.Length-6.1,  0)^3
     - 0.334  * Petal.Width
```

where `pmax` is the maximum of the arguments inside its parentheses.

Probit Regression

Colorectal Cancer Screening Data on Individuals

The data in the `colscr2` data frame describe (disguised) data on the status of 172 adults who were eligible for colon cancer screening. The goal is to use the other variables (besides subject ID) to predict whether or not a subject is up to date.

colscr2 contents

Variable	Description
subject	subject ID code
age	subject's age (years)
race	subject's race (White/Black/Other)
hispanic	subject of Hispanic ethnicity (1 = yes / 0 = no)
insurance	Commercial, Medicaid, Medicare, Uninsured
bmi	body mass index at most recent visit
sbp	systolic blood pressure at most recent visit
up_to_date	meets colon cancer screening standards

summary(colscr2)

```
> summary(colscr2)
    subject           age            race       hispanic
 Min.   :101.0   Min.   :51.00   Black:118   Min.   :0.00000
 1st Qu.:143.8   1st Qu.:54.00   Other:  9   1st Qu.:0.00000
 Median :186.5   Median :57.00   White: 45   Median :0.00000
 Mean   :186.5   Mean   :57.80               Mean   :0.06395
 3rd Qu.:229.2   3rd Qu.:61.25               3rd Qu.:0.00000
 Max.   :272.0   Max.   :69.00               Max.   :1.00000
      insurance       bmi             sbp          up_to_date
 Commercial:32   Min.   :17.20   Min.   : 89.0   Min.   :0.0000
 Medicaid  :81   1st Qu.:25.48   1st Qu.:118.0   1st Qu.:0.0000
 Medicare  :46   Median :30.05   Median :127.0   Median :1.0000
 Uninsured :13   Mean   :31.24   Mean   :128.9   Mean   :0.6047
                 3rd Qu.:36.03   3rd Qu.:138.0   3rd Qu.:1.0000
                 Max.   :55.41   Max.   :198.0   Max.   :1.0000
```

A logistic regression model

```
m_scr2_logistic <- glm(up_to_date ~ age + race + hispanic +
                    insurance + bmi + sbp,
                family = binomial, data = colscr2)
```

Results

```
# A tibble: 10 x 5
   term                estimate std.error statistic p.value
   <chr>                  <dbl>     <dbl>     <dbl>   <dbl>
 1 (Intercept)          2.70      2.74     0.986    0.324
 2 age                  0.0205    0.0397   0.516    0.606
 3 raceOther           -1.97      1.00    -1.97     0.0491
 4 raceWhite           -0.321     0.400   -0.802    0.422
 5 hispanic             0.000585  0.795    0.000736 0.999
 6 insuranceMedicaid   -1.02      0.495   -2.05     0.0401
 7 insuranceMedicare   -0.522     0.563   -0.926    0.354
 8 insuranceUninsured   0.110     0.791    0.139    0.889
 9 bmi                  0.0156    0.0214   0.730    0.465
10 sbp                 -0.0242    0.00991 -2.44     0.0147
```

In this model, there appears to be some link between `sbp` and screening, as well as, perhaps, some statistically significant differences between some race groups and some insurance groups.

Predicting status for Harry and Sally

- Harry is age 65, White, non-Hispanic, with Medicare insurance, a BMI of 28 and SBP of 135.
- Sally is age 60, Black, Hispanic, with Medicaid insurance, a BMI of 22 and SBP of 148.

```
newdat_s2 <- tibble(subject = c("Harry", "Sally"),
                     age = c(65, 60),
                     race = c("White", "Black"),
                     hispanic = c(0, 1),
                     insurance = c("Medicare", "Medicaid"),
                     bmi = c(28, 22),
                     sbp = c(135, 148))
```

Predicting Harry and Sally's status

```
predict(m_scr2_logistic, newdata = newdat_s2,
        type = "response")
```

```
        1         2 
0.5904364 0.4215335 
```

The prediction for Harry is 0.59, and for Sally, 0.42, by this logistic regression model.

A probit regression model

Now, consider a probit regression, fit by changing the default link for the `binomial` family as follows:

```
m_scr2_probit <- glm(up_to_date ~ age + race + hispanic +
                         insurance + bmi + sbp,
                     family = binomial(link = "probit"),
                     data = colscr2)
```

tidy(m_scr2_probit)

```
# A tibble: 10 x 5
   term                estimate std.error statistic p.value
   <chr>                  <dbl>     <dbl>     <dbl>   <dbl>
 1 (Intercept)          1.58      1.66       0.955   0.339
 2 age                  0.0135    0.0241     0.558   0.577
 3 raceOther           -1.24      0.587     -2.11    0.0349
 4 raceWhite           -0.199     0.244     -0.818   0.413
 5 hispanic             0.0295    0.485      0.0608  0.952
 6 insuranceMedicaid   -0.619     0.293     -2.11    0.0347
 7 insuranceMedicare   -0.323     0.334     -0.968   0.333
 8 insuranceUninsured   0.0528    0.464      0.114   0.909
 9 bmi                  0.00965   0.0129     0.749   0.454
10 sbp                 -0.0147    0.00594   -2.47    0.0134
```

Interpreting the Probit Model's Coefficients

```
      (Intercept)               age         raceOther
      1.584603569       0.013461338      -1.238445198
        raceWhite          hispanic insuranceMedicaid
     -0.199260184       0.029483051      -0.619276718
insuranceMedicare insuranceUninsured              bmi
     -0.322880519       0.052775722       0.009652339
              sbp
     -0.014695526
```

The probit regression coefficients give the change in the z-score of the outcome of interest (here, `up_to_date`) for a one-unit change in the target predictor, holding all other predictors constant.

- So, for a one-year increase in age, holding all other predictors constant, the z-score for `up_to_date` increases by 0.013
- And for a Medicaid subject as compared to a Commercial subject of the same age, race, ethnicity, bmi and sbp, the z-score for the Medicaid subject is predicted to be -0.619 lower, according to this model.

What about Harry and Sally?

Do the predictions for Harry and Sally change much with this probit model, as compared to the logistic regression?

```
predict(m_scr2_probit, newdata = newdat_s2,
        type = "response")
```

```
        1         2 
0.5885511 0.4364027 
```

Enjoy Your Spring Break!

- Be sure to submit your Project 1 Portfolio and Poster to Canvas by 2 PM on Monday 2020-03-09.