Chapter 14 Logistic regression

Adapted from Timothy Hanson

Department of Statistics, University of South Carolina

Stat 705: Data Analysis II

Generalized linear models

- Generalize regular regression to non-normal data $\{(Y_i, \mathbf{x}_i)\}_{i=1}^n$, most often Bernoulli or Poisson Y_i.
- The general theory of GLMs has been developed to outcomes in the exponential family (normal, gamma, Poisson, binomial, negative binomial, ordinal/nominal multinomial).
- The ith *mean* is $\mu_i = E(Y_i)$
- The ith *linear predictor is* $\eta_i = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_k x_{ik} = \mathbf{x}_i' \boldsymbol{\beta}$.
- A GLM relates the mean to the linear predictor through a *link function* $g(\cdot)$:

$$g(\mu_i) = \eta_i = \mathbf{x}_i' \boldsymbol{\beta}.$$

14.1, 14.2 Binary response regression

Let $Y_i \sim \text{Bern}(\pi_i)$. Y_i might indicate the presence/absence of a disease, whether O-rings on the Challenger will fail, etc. (pp. 555-556).

We wish to relate the probability of "success" π_i to explanatory covariates $\mathbf{x}_i = (1, x_{i1}, \ldots, x_{ik})$.

$$Y_i \sim \text{Bern}(\pi_i),$$

implying $E(Y_i) = \pi_i$ and $\text{var}(Y_i) = \pi_i(1 - \pi_i)$.

Identity link $g(\mu) = \mu$

The **identity link** gives $\pi_i = \boldsymbol{\beta}'\mathbf{x}_i$. When $\mathbf{x}_i = (1, x_i)'$, this reduces to

$$Y_i \sim \text{Bern}(\beta_0 + \beta_1 x_i).$$

- When x_i is large or small, π_i can be less than zero or greater than one.
- The identity like is appropriate for a restricted range of x_i values.
- It can of course be extended to $\pi_i = \boldsymbol{\beta}'\mathbf{x}_i$ where $\mathbf{x}_i = (1, x_{i1}, \ldots, x_{ik})$.
- This model can be fit in SAS `proc genmod`.

Individual Bernoulli vs. aggregated binomial

Data can be stored in one of two ways:

- If each subject has their own individual binary outcome Y_i, we can write `model y=x1 x2` in `proc genmod` or `proc logistic`.
- If data are grouped, so that there are $Y_{\cdot j}$ successes out of n_j with covariate $\mathbf{x}_j$, $j = 1, \ldots, c$, then write `model y/n=x1 x2`. This method is sometimes used to reduce a very large number of individuals n to a small number of distinct covariates c; it is essentially a product binomial model.

Association between snoring and heart disease

From Agresti (2013).

Let s be someone's snoring score, $s \in \{0, 2, 4, 5\}$.

		Heart disease		Proportion
Snoring	s	yes	no	yes
Never	0	24	1355	0.017
Occasionally	2	35	603	0.055
Nearly every night	4	21	192	0.099
Every night	5	30	224	0.118

This is fit in `proc genmod`:

```
data glm;
 input snoring disease total @@;
 datalines;
 0 24 1379 2 35 638 4 21 213 5 30 254
 ;
proc genmod data=glm; model disease/total = snoring / dist=bin link=identity;
run;
```

Extracting useful inferences

The fitted model is

$$\hat{\pi}(s) = 0.0172 + 0.0198s.$$

For every unit increase in snoring score s, the probability of heart disease increases by about 2%.

The p-values test $H_0 : \beta_0 = 0$ and $H_0 : \beta_1 = 0$. The latter is more interesting and we reject at the $\alpha = 0.001$ level. The probability of heart disease is strongly, *linearly* related to the snoring score.

We'll denote the maximum likelihood estimates by $\hat{\boldsymbol{\beta}}$ instead of $\mathbf{b}$ in this chapter. Both PROC LOGISTIC and PROC GENMOD give MLEs.

14.2, 14.3 Logistic regression

Often a fixed change in x has less impact when $\pi(x)$ is near zero or one.

Example: Let $\pi(x)$ be probability of getting an A in a statistics class and x is the number of hours a week you work on homework. When $x = 0$, increasing x by 1 will change your (very small) probability of an A very little. When $x = 4$, adding an hour will change your probability quite a bit. When $x = 20$, that additional hour probably won't improve your chances of getting an A much. You were at essentially $\pi(x) \approx 1$ at $x = 10$.

Of course, this is a *mean* model. Individuals will vary.

Logit link and logistic regression

The most widely used nonlinear function to model probabilities is the logit link:

$$\text{logit}(\pi_i) = \log\left(\frac{\pi_i}{1-\pi_i}\right) = \beta_0 + \beta_1 x_i.$$

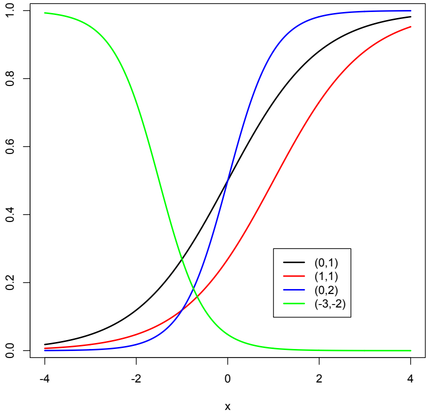
Solving for π_i and then dropping the subscripts we get the probability of success ($Y = 1$) as a function of x:

$$\pi(x) = \frac{\exp(\beta_0 + \beta_1 x)}{1 + \exp(\beta_0 + \beta_1 x)} = \left[1 + \exp(-\beta_0 - \beta_1 x)\right]^{-1}.$$

When $\beta_1 > 0$ the function increases from 0 to 1; when $\beta_1 < 0$ it decreases. When $\beta = 0$ the function is constant for all values of x and Y is unrelated to x.

The logistic function is $\text{logit}^{-1}(x) = e^x/(1+e^x)$.

Logistic functions for various β_0 and β_1



Logistic curves $\pi(x) = e^{\beta_0+\beta_1 x}/(1+e^{\beta_0+\beta_1 x})$ with $(\beta_0, \beta_1) = (0,1)$, $(1,1)$, $(0,2)$, $(-3,-2)$. What about $(\beta_0, \beta_1) = (\log 2, 0)$?

Logistic regression on snoring data

To fit the snoring data to the logistic regression model we use the same SAS code as before (`proc genmod`), except we specify LINK=LOGIT (or drop the LINK option, since LOGIT is the default) and obtain $b_0 = -3.87$ and $b_1 = 0.40$ as maximum likelihood estimates.

```
proc genmod data=glm;
*We dropped DIST=BIN too, though it’s better practice to include it;
model disease/total = snoring;
run;
```

Logistic output

You can also use `proc logistic` to fit binary regression models.

```
proc logistic data=glm;
model disease/total = snoring;
run;
```

The fitted model is $\hat{\pi}(x) = \frac{\exp(-3.87+0.40x)}{1+\exp(-3.87+0.40x)}$. As before, we reject $H_0 : \beta_1 = 0$; there is a strong, positive association between snoring score and developing heart disease.

Crab mating (Agresti, 2013)

Data on $n = 173$ female horseshoe crabs.

- C = color (1,2,3,4=light medium, medium, dark medium, dark).
- S = posterior(?) spine condition (1,2,3=both good, one worn or broken, both worn or broken). Males attach to posterior spines when mating.
- W = carapace width (cm).
- Wt = weight (kg).
- Sa = number of satellites (additional male crabs besides her nest-mate husband) nearby. Satellite males fertilize some of the female's eggs.

We are initially interested in the probability that a female horseshoe crab has one or more satellites ($Y_i = 1$) as a function of carapace width.

Horseshoe Crab facts

- Horseshoe crabs aren't that closely related to crabs.
- Their mass spawning events (e.g., at Delaware Bay in DE and NJ) attract thousands of shorebirds, including the threatened Red Knot
- These events also attract(ed) commercial fishermen (eel and conch fisheries), fertilizer companies (no longer), and the biomedical industry (unique antibacterial properties of their blue blood)
- Exploitation of horseshoe crabs has greatly affected migrating shorebirds as well (see Red Knot above).

Horseshoe Crab spawning

: Horseshoe Crab spawning event

: Female Horseshoe Crab with mate and satellite males

Shorebirds Feeding

: Shore birds feeding on horseshoe crab eggs

: Red Knot with tag B95–the so-called Moon Bird–has migrated over a quarter-million miles since first tagged in 1995

Crab data in SAS

```
data crabs;
weight=weight/1000; color=color-1;
*Convert satellite to a binary variable rather than a count;
y=0; if satell>0 then y=1; id=_n_; run;
proc logistic data=crabs;
model y(event=’1’)=width / link=logit; run;
```

`event=’1’` tells SAS to model $\pi_i = P(Y_i = 1)$ rather than $\pi_i = P(Y_i = 0)$. The default link is logit (giving logistic regression) – I specify it here anyway for transparency.

14.3 Model interpretation

For simple logistic regression

$$Y_i \sim \text{Bern}\left(\frac{e^{\beta_0+\beta_1 x_i}}{1+e^{\beta_0+\beta_1 x_i}}\right).$$

An odds ratio: let's look at how the odds of success changes when we increase x by one unit:

$$\begin{aligned}\frac{\pi(x+1)/[1-\pi(x+1)]}{\pi(x)/[1-\pi(x)]} &= \frac{\left[\frac{e^{\beta_0+\beta_1 x+\beta_1}}{1+e^{\beta_0+\beta_1 x+\beta_1}}\right] \Big/ \left[\frac{1}{1+e^{\beta_0+\beta_1 x+\beta_1}}\right]}{\left[\frac{e^{\beta_0+\beta_1 x}}{1+e^{\beta_0+\beta_1 x}}\right] \Big/ \left[\frac{1}{1+e^{\beta_0+\beta_1 x}}\right]} \\ &= \frac{e^{\beta_0+\beta_1 x+\beta_1}}{e^{\beta_0+\beta_1 x}} = e^{\beta_1}.\end{aligned}$$

When we increase x by one unit, the odds of an event occurring increases by a factor of e^{β_1}, *regardless of the value of x*.

Horseshoe crab data

Let's look at $Y_i = 1$ if a female crab has one or more satellites, and $Y_i = 0$ if not. So

$$\pi(x) = \frac{e^{\beta_0+\beta_1 x}}{1 + e^{\beta_0+\beta_1 x}},$$

is the probability of a female having more than her nest-mate around as a function of her width x.

From SAS's output we obtain a table with estimates $\hat{\beta}_0$ and $\hat{\beta}_1$ as well as standard errors, χ^2 test stattistics, and p-values that $H_0 : \beta_0 = 0$ and $H_0 : \beta_1 = 0$. We also obtain an estimate of the odds ratio e^{b_1} and a 95% CI for e^{β_1}.

SAS output

```
                                   Standard          Wald
Parameter     DF     Estimate         Error    Chi-Square    Pr > ChiSq
Intercept      1     -12.3508        2.6287       22.0749        <.0001
width          1       0.4972        0.1017       23.8872        <.0001

                        Odds Ratio Estimates

                          Point          95% Wald
             Effect     Estimate      Confidence Limits
             width         1.644       1.347       2.007
```

We estimate the probability of a satellite as

$$\hat{\pi}(x) = \frac{e^{-12.35+0.50x}}{1+e^{-12.35+0.50x}}.$$

The odds of having a satellite increases by a factor between 1.3 and 2.0 times for every *cm* increase in carapace width.

The coefficient table houses estimates $\hat{\beta}_j$, $\text{se}(\hat{\beta}_j)$, and the Wald statistic $z_j^2 = \{\hat{\beta}_j/\text{se}(\hat{\beta}_j)\}^2$ and p-value for testing $H_0 : \beta_j = 0$. What do we conclude here?

14.4 Multiple predictors

Now we have $k = p - 1$ predictors $\mathbf{x}_i = (1, x_{i1}, \ldots, x_{i,p-1})$ and fit

$$Y_i \sim \text{bin}\left(n_i, \frac{\exp(\beta_0 + \beta_1 x_{i1} + \cdots + \beta_{p-1} x_{i,p-1})}{1 + \exp(\beta_0 + \beta_1 x_{i1} + \cdots + \beta_{p-1} x_{i,p-1})}\right).$$

- Many of these predictors may be sets of dummy variables associated with categorical predictors.
- e^{β_j} is now termed the *adjusted* odds ratio. This is how the odds of the event occurring changes when x_j increases by one unit *keeping the remaining predictors constant*.
- This interpretation may not make sense if two predictors are highly related. Examples?

Categorical predictors

We have predictor X, a categorical variable that takes on values $x \in \{1, 2, \ldots, I\}$. We need to allow each level of $X = x$ to affect $\pi(x)$ differently. This is accomplished by the use of dummy variables. The usual zero/one dummies $z_1, z_2, \ldots, z_{I-1}$ for X are defined:

$$z_j = \left\{ \begin{array}{cc} 1 & X = j \\ 0 & X \neq j \end{array} \right\}.$$

This is the default in PROC GENMOD with a CLASS X statement; it can be obtained in PROC LOGISTIC with the PARAM=REF option.

Example with $I = 3$

Say $I = 3$, then a simple logistic regression in X is

$$\text{logit } \pi(x) = \beta_0 + \beta_1 z_1 + \beta_2 z_2.$$

which gives

$$\begin{aligned} \text{logit } \pi(x) &= \beta_0 + \beta_1 && \text{when} && X = 1 \\ \text{logit } \pi(x) &= \beta_0 + \beta_2 && \text{when} && X = 2 \\ \text{logit } \pi(x) &= \beta_0 && \text{when} && X = 3 \end{aligned}$$

This sets class $X = I$ as the baseline. The first category can be set to baseline with REF=FIRST next to the variable name in the CLASS statement.

14.5 Tests of regression effects

An overall test of H_0 : logit $\pi(\mathbf{x}) = \beta_0$ versus H_1 : logit $\pi(\mathbf{x}) = \mathbf{x}'\boldsymbol{\beta}$ is generated in PROC LOGISTIC three different ways: LRT, score, and Wald versions. This checks whether some subset of variables in the model is important.

Recall the crab data covariates:

- C = color (1,2,3,4=light medium, medium, dark medium, dark).
- S = spine condition (1,2,3=both good, one worn or broken, both worn or broken).
- W = carapace width (cm).
- Wt = weight (kg).

We'll take $C = 4$ and $S = 3$ as baseline categories.

Crab data

There are two categorical predictors, C and S, and two continuous predictors W and Wt. Let $Y = 1$ if a randomly drawn crab has one or more satellites and $\mathbf{x} = (C, S, W, Wt)$ be her covariates. An *additive* model including all four covariates would look like

$$\begin{aligned} \text{logit } \pi(\mathbf{x}) &= \beta_0 + \beta_1 I\{C=1\} + \beta_2 I\{C=2\} + \beta_3 I\{C=3\} \\ &\quad + \beta_4 I\{S=1\} + \beta_5 I\{S=2\} + \beta_6 W + \beta_7 Wt \end{aligned}$$

This model is fit via

```
proc logistic data=crabs descending;
 class color spine / param=ref;
 model y = color spine width weight / lackfit;
```

The H-L GOF statistic yields $p = 0.88$ so there's no evidence of gross lack of fit.

Parameter estimates & overall tests

```
                                    Standard          Wald
Parameter      DF     Estimate         Error    Chi-Square    Pr > ChiSq
Intercept       1      -9.2734        3.8378        5.8386        0.0157
color     1     1       1.6087        0.9355        2.9567        0.0855
color     2     1       1.5058        0.5667        7.0607        0.0079
color     3     1       1.1198        0.5933        3.5624        0.0591
spine     1     1      -0.4003        0.5027        0.6340        0.4259
spine     2     1      -0.4963        0.6292        0.6222        0.4302
width           1       0.2631        0.1953        1.8152        0.1779
weight          1       0.8258        0.7038        1.3765        0.2407
```

Color seems to be important. Plugging in $\hat{\boldsymbol{\beta}}$ for $\boldsymbol{\beta}$,

$$
\begin{aligned}
\operatorname{logit} \hat{\pi}(\mathbf{x}) \quad = \quad & -9.27 + 1.61 I\{C = 1\} + 1.51 I\{C = 2\} + 1.11 I\{C = 3\} \\
& -0.40 I\{S = 1\} - 0.50 I\{S = 2\} + 0.26 W + 0.83 Wt
\end{aligned}
$$

Overall checks that one or more predictors are important:

```
              Testing Global Null Hypothesis: BETA=0

Test                 Chi-Square       DF     Pr > ChiSq
Likelihood Ratio        40.5565        7         <.0001
Score                   36.3068        7         <.0001
Wald                    29.4763        7         0.0001
```

Type III tests for dropping effects

Type III tests are (1) $H_0 : \beta_1 = \beta_2 = \beta_3 = 0$, color not needed to explain whether a female has satellite(s), (2) $H_0 : \beta_4 = \beta_5 = 0$, spine not needed, (3) $H_0 : \beta_6 = 0$, width not needed, and (4) $H_0 : \beta_7 = 0$, weight not needed:

```
                   Type 3 Analysis of Effects

                                      Wald
         Effect         DF     Chi-Square     Pr > ChiSq
         color           3         7.1610         0.0669
         spine           2         1.0105         0.6034
         width           1         1.8152         0.1779
         weight          1         1.3765         0.2407
```

The largest *p*-value is 0.6 for dropping spine; when refitting the model without spine, we still strongly reject $H_0 : \beta_1 = \beta_2 = \beta_3 = \beta_4 = \beta_5 = \beta_6 = 0$, and the H-L shows no evidence of lack of fit. We have:

```
                   Type 3 Analysis of Effects

                                      Wald
         Effect         DF     Chi-Square     Pr > ChiSq
         color           3         6.3143         0.0973
         width           1         2.3355         0.1265
         weight          1         1.2263         0.2681
```

Continuing...

We do not reject that we can drop weight from the model, and so we do (What happened to width?!):

```
               Testing Global Null Hypothesis: BETA=0

       Test                 Chi-Square       DF     Pr > ChiSq
       Likelihood Ratio        38.3015        4         <.0001
       Score                   34.3384        4         <.0001
       Wald                    27.6788        4         <.0001

                      Type 3 Analysis of Effects

                                      Wald
              Effect      DF    Chi-Square    Pr > ChiSq
              color        3        6.6246        0.0849
              width        1       19.6573        <.0001

              Analysis of Maximum Likelihood Estimates

                                 Standard          Wald
Parameter      DF    Estimate       Error    Chi-Square    Pr > ChiSq
Intercept       1    -12.7151      2.7618       21.1965        <.0001
color     1     1      1.3299      0.8525        2.4335        0.1188
color     2     1      1.4023      0.5484        6.5380        0.0106
color     3     1      1.1061      0.5921        3.4901        0.0617
width           1      0.4680      0.1055       19.6573        <.0001
```

Final model

The new model is

$$\text{logit } \pi(\mathbf{x}) = \beta_0 + \beta_1 I\{C = 1\} + \beta_2 I\{C = 2\} \beta_3 I\{C = 3\} + \beta_4 W.$$

We *do not* reject that color can be dropped from the model $H_0 : \beta_1 = \beta_2 = \beta_3$, but we do reject that the dummy for $C = 2$ can be dropped, $H_0 : \beta_2 = 0$.

Maybe unnecessary levels in color are clouding its importance. It's possible to test whether we can combine levels of C using `contrast` statements. When I tried this, I was able to combine colors 1, 2, and 3 into one "light" category vs. color 4 "dark."

Comments

- The odds of having satellite(s) significantly increases by $e^{1.4023} \approx 4$ for medium vs. dark crabs.
- The odds of having satellite(s) significantly increases by a factor of $e^{0.4680} \approx 1.6$ for every *cm* increase in carapace width when fixing color.
- Lighter, wider crabs tend to have satellite(s) more often.
- The H-L GOF test shows no gross LOF.
- We didn't check for interactions. If an interaction between color and width existed, then the odds ratio of satellite(s) for different colored crabs would change with how wide she is.

Interactions

An additive model is easily interpreted because an odds ratio from changing values of one predictor does not change with levels of another predictor. However, often this is incorrect and we may introduce additional terms into the model such as interactions.

An interaction between two predictors allows the odds ratio for increasing one predictor to change with levels of another. For example, in the last model fit, the odds of having satellite(s) increases by a factor of 4 for medium crabs vs. dark *regardless of carapace width*.

Interactions, continued...

A two-way interaction is defined by multiplying the variables together; if one or both variables are categorical then all possible pairings of dummy variables are considered.

In PROC GENMOD and PROC LOGISTIC, categorical variables are defined through the CLASS statement and all dummy variables are created and handled internally. The Type III table provides a test that the interaction can be dropped; the table of regression coefficients tells you whether individual dummies can be dropped.

Quadratic effects (pp. 575–577)

For a categorical predictor X with I levels, adding $I-1$ dummy variables allows for a different event probability at each level of X.

For a continuous predictor Z, the model assumes that the log-odds of the event increases *linearly* with Z. This may or may not be a reasonable assumption, but can be checked by adding nonlinear terms, the simplest being Z^2.

Consider a simple model with continuous Z:

$$\text{logit } \pi(Z) = \beta_0 + \beta_1 Z.$$

LOF from this model can manifest itself in rejecting a GOF test (Pearson, deviance, or H-L) or a residual plot that shows curvature.

Quadratic and higher order effects

Adding a quadratic term

$$\text{logit } \pi(Z) = \beta_0 + \beta_1 Z + \beta_2 Z^2,$$

may improve fit and allows testing the adequacy of the simpler model via $H_0 : \beta_2 = 0$. Cubic and higher order powers can be added, but the model can become unstable with, say, higher than cubic powers. A better approach might be to fit a *generalized additive model* (GAM):

$$\text{logit } \pi(Z) = f(Z),$$

where $f(\cdot)$ is estimated from the data, often using splines.

Adding a simple quadratic term can be done, e.g.,

```
proc logistic; model y/n = z z*z;
```

14.6 Model selection

Two competing goals:

- The model should fit the data well.
- The model should be simple to interpret (smooth rather than overfit – principle of parsimony).

Often hypotheses on how the outcome is related to specific predictors will help guide the model building process.

Agresti (2013) suggests a rule of thumb: at least 10 events and 10 non-events should occur for each predictor in the model (including dummies). So if $\sum_{i=1}^{N} y_i = 40$ and $\sum_{i=1}^{N} n_i = 830$, you should have no more than $40/10 = 4$ predictors in the model.

Horseshoe crab data

Recall that in all models fit we strongly rejected $H_0 : \text{logit } \pi(\mathbf{x}) = \beta_0$ in favor of $H_1 : \text{logit } \pi(\mathbf{x}) = \mathbf{x}'\boldsymbol{\beta}$:

```
                  Testing Global Null Hypothesis: BETA=0

          Test                 Chi-Square       DF     Pr > ChiSq
          Likelihood Ratio        40.5565        7         <.0001
          Score                   36.3068        7         <.0001
          Wald                    29.4763        7         0.0001
```

However, it was not until we carved superfluous predictors from the model that we showed significance for the included model effects.

This is an indication that several covariates may be highly related, or correlated. Often variables are *highly* correlated and therefore one or more are redundant. We need to get rid of some!

Automated model selection

Although not ideal, automated model selection is necessary with large numbers of predictors. With $p-1=10$ predictors, there are $2^{10}=1024$ possible models; with $p-1=20$ there are $1,048,576$ to consider.

Backwards elimination starts with a large pool of potential predictors and step-by-step eliminates those with (Wald) p-values larger than a cutoff (the default is 0.05 in SAS PROC LOGISTIC).

We performed backwards elimination by hand for the crab mating data.

Backwards elimination in SAS, default cutoff

```
proc logistic data=crabs1 descending;
 class color spine / param=ref;
 model y = color spine width weight color*spine color*width color*weight
  spine*width spine*weight width*weight / selection=backward;
```

When starting from all main effects and two-way interactions, the default *p*-value cutoff 0.05 yields only the model with width as a predictor

```
                    Summary of Backward Elimination

           Effect                   Number        Wald
   Step    Removed          DF          In    Chi-Square    Pr > ChiSq
      1    color*spine       6           9        0.0837        1.0000
      2    width*color       3           8        0.8594        0.8352
      3    width*spine       2           7        1.4906        0.4746
      4    weight*spine      2           6        3.7334        0.1546
      5    spine             2           5        2.0716        0.3549
      6    width*weight      1           4        2.2391        0.1346
      7    weight*color      3           3        5.3070        0.1507
      8    weight            1           2        1.2263        0.2681
      9    color             3           1        6.6246        0.0849

                Analysis of Maximum Likelihood Estimates

                                     Standard          Wald
  Parameter      DF    Estimate        Error    Chi-Square    Pr > ChiSq
  Intercept       1    -12.3508       2.6287       22.0749        <.0001
  width           1      0.4972       0.1017       23.8872        <.0001
```

Backwards elimination in SAS, cutoff 0.15

Let's change the criteria for removing a predictor to p-value ≥ 0.15.

```
model y = color spine width weight color*spine color*width color*weight
 spine*width spine*weight width*weight / selection=backward slstay=0.15;
```

Yielding a more complicated model:

```
                      Summary of Backward Elimination

             Effect                        Number        Wald
     Step    Removed           DF              In    Chi-Square    Pr > ChiSq
        1    color*spine        6               9        0.0837        1.0000
        2    width*color        3               8        0.8594        0.8352
        3    width*spine        2               7        1.4906        0.4746
        4    weight*spine       2               6        3.7334        0.1546
        5    spine              2               5        2.0716        0.3549
```

AIC & model selection

“All models are wrong; some models are useful.” – George Box*.

It is often of interest to examine several competing models. In light of underlying biology or science, one or more models may have relevant interpretations within the context of why data were collected in the first place.

In the absence of scientific input, a widely-used model selection tool is the Akaike information criterion (AIC),

$$\text{AIC} = -2[L(\hat{\beta};\mathbf{y}) - p].$$

The term $L(\hat{\beta};\mathbf{y})$ represents model fit. If you add a parameter to a model, $L(\hat{\beta};\mathbf{y})$ has to increase. If we only used $L(\hat{\beta};\mathbf{y})$ as a criterion, we'd keep adding predictors until we ran out. The p penalizes for the number of the predictors in the model.

Crab data

The AIC has very nice properties in large samples in terms of prediction. The smaller the AIC is, the better the model fit (asymptotically).

Model	AIC
W	198.8
$C + W$	197.5
$C + W + Wt + W * C + W * Wt$	196.8

The best model is the most complicated one, according to AIC. One might choose the slightly "worse" model $C + W$ for its enhanced interpretability.

14.7 Goodness of fit and grouping

The deviance GOF statistic is defined to be

$$G^2 = 2\sum_{j=1}^{c}\left\{Y_{\cdot j}\log\left(\frac{Y_{\cdot j}}{n_j\hat{\pi}_j}\right) + (n_j - Y_{\cdot j})\log\left(\frac{1 - Y_{\cdot j}/n_j}{1-\hat{\pi}_j}\right)\right\},$$

where $\hat{\pi}_j = \frac{e^{\mathbf{x}_j'\mathbf{b}}}{1+e^{\mathbf{x}_j'\mathbf{b}}}$ are fitted values.

Pearson's GOF statistic is

$$X^2 = \sum_{j=1}^{c}\frac{(Y_{\cdot j} - n_j\hat{\pi}_j)^2}{n_j\hat{\pi}_j(1-\hat{\pi}_j)}.$$

Both statistics are approximately χ^2_{c-p} in large samples assuming that the number of *trials* $n = \sum_{j=1}^{c} n_j$ increases in such a way that each n_j increases. These are the same type of GOF test requiring replicates in Sections 3.7 & 6.8.

Aggregating over distinct covariates

Binomial data is often recorded as individual (Bernoulli) records:

i	y_i	n_i	x_i
1	0	1	9
2	0	1	14
3	1	1	14
4	0	1	17
5	1	1	17
6	1	1	17
7	1	1	20

Grouping the data yields an identical model:

i	y_i	n_i	x_i
1	0	1	9
2	1	2	14
3	2	3	17
4	1	1	20

- $\hat{\boldsymbol{\beta}}$, se$(\hat{\beta}_j)$, and $L(\hat{\boldsymbol{\beta}})$ don't care if data are grouped.
- The quality of residuals and GOF statistics *depend on how data are grouped*. D and Pearson's X^2 will change! (Bottom, p. 590).

Comments on grouping

- In PROC LOGISTIC, type AGGREGATE and SCALE=NONE after the MODEL statement to get D and X^2 based on grouped data. This option *does not* compute residuals based on the grouped data. You can aggregate over all variables or a subset, e.g. AGGREGATE=(width).
- The Hosmer and Lemeshow test statistic orders observations $(\mathbf{x}_i, Y_i)$ by fitted probabilities $\hat{\pi}(\mathbf{x}_i)$ from smallest to largest and divides them into (typically) $g = 10$ groups of roughly the same size. A Pearson test statistic is computed from these g groups; this statistic is *approximately* χ^2_{g-2}. Termed a "near-replicate GOF test." The LACKFIT option in PROC LOGISTIC gives this statistic.

Comments on grouping

- Pearson, Deviance, and Hosmer & Lemeshow all provide a p-value for the null H_0 : the model fits based on χ^2_{c-p} where c is the distinct number of covariate levels (using AGGREGATE). The alternative model for the first two is the *saturated model* where every μ_i is simply replaced by y_i.
- We can also test $\text{logit}\{\pi(x)\} = \beta_0 + \beta_1 x$ versus the more general model $\text{logit}\{\pi(x)\} = \beta_0 + \beta_1 x + \beta_2 x^2$ via $H_0 : \beta_2 = 0$.

Crab data GOF tests, only width as predictor

Raw (Bernoulli) data with `aggregate scale=none lackfit;`

```
            Deviance and Pearson Goodness-of-Fit Statistics

   Criterion          Value       DF      Value/DF      Pr > ChiSq
   Deviance         69.7260       64        1.0895          0.2911
   Pearson          55.1779       64        0.8622          0.7761

                     Number of unique profiles: 66

               Partition for the Hosmer and Lemeshow Test

                                    y = 1                  y = 0
   Group         Total     Observed     Expected     Observed    Expected
       1            19            5         5.39           14       13.61
       2            18            8         7.62           10       10.38
       3            17           11         8.62            6        8.38
       4            17            8         9.92            9        7.08
       5            16           11        10.10            5        5.90
       6            18           11        12.30            7        5.70
       7            16           12        12.06            4        3.94
       8            16           12        12.90            4        3.10
       9            16           13        13.69            3        2.31
      10            20           20        18.41            0        1.59

                Hosmer and Lemeshow Goodness-of-Fit Test

                      Chi-Square       DF      Pr > ChiSq
                          5.2465        8          0.7309
```

Comments

- There are $c = 66$ distinct widths $\{\mathbf{x}_i\}$ out of $n = 173$ crabs. For χ^2_{66-2} to hold, we must keep sampling crabs that only have one of the 66 *fixed number of widths*! Does that make sense here?
- The Hosmer and Lemeshow test gives a p-value of 0.73 based on $g = 10$ groups.
- The raw statistics do not tell you *where* lack of fit occurs. Deviance and Pearson residuals do tell you this (later). Also, the table provided by the H-L tells you which groups are ill-fit should you reject H_0 : logistic model holds.
- GOF tests are meant to detect *gross* deviations from model assumptions. No model ever truly fits data except hypothetically.

14.8 Diagnostics

GOF tests are *global* checks for model adequacy. Residuals and influential measures can refine a model inadequacy diagnosis.

The data are $(\mathbf{x}_j, Y_{\cdot j})$ for $j = 1, \ldots, c$. The j^{th} *fitted value* is an estimate of $\mu_j = E(Y_{\cdot j})$, namely $\widehat{E(Y_{\cdot j})} = \hat{\mu}_j = n_j \hat{\pi}_j$ where $\pi_j = \frac{e^{\boldsymbol{\beta}' \mathbf{x}_j}}{1+e^{\boldsymbol{\beta}' \mathbf{x}_j}}$ and $\hat{\pi}_j = \frac{e^{\hat{\boldsymbol{\beta}}' \mathbf{x}_j}}{1+e^{\hat{\boldsymbol{\beta}}' \mathbf{x}_j}}$. The raw residual e_j is what we see $(Y_{\cdot j})$ minus what we predict $(n_j \hat{\pi}_j)$. The Pearson residual divides this by an estimate of $\sqrt{\text{var}(Y_{\cdot j})}$:

$$r_{P_j} = \frac{y_{\cdot j} - n_j \hat{\pi}_j}{\sqrt{n_j \hat{\pi}_j (1 - \hat{\pi}_j)}}.$$

The Pearson GOF statistic is

$$X^2 = \sum_{j=1}^{c} r_{P_j}^2.$$

Diagnostics

The standardized Pearson residual is given by

$$r_{SP_j} = \frac{y_{\cdot j} - n_j \hat{\pi}_j}{\sqrt{n_j \hat{\pi}_j (1 - \hat{\pi}_j)(1 - \hat{h}_j)}},$$

where $\hat{h}_j$ is the j^{th} diagonal element of the *hat* matrix $\hat{\mathbf{H}} = \hat{\mathbf{W}}^{1/2} \mathbf{X} (\mathbf{X}' \hat{\mathbf{W}} \mathbf{X})^{-1} \mathbf{X}' \hat{\mathbf{W}}^{1/2}$ where $\mathbf{X}$ is the design matrix

$$\mathbf{X} = \begin{bmatrix} 1 & x_{11} & \cdots & x_{1,p-1} \\ 1 & x_{21} & \cdots & x_{2,p-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{c1} & \cdots & x_{c,p-1} \end{bmatrix},$$

and

$$\hat{\mathbf{W}} = \begin{bmatrix} n_1 \hat{\pi}_1 (1 - \hat{\pi}_1) & 0 & \cdots & 0 \\ 0 & n_2 \hat{\pi}_2 (1 - \hat{\pi}_2) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & n_c \hat{\pi}_c (1 - \hat{\pi}_c) \end{bmatrix}.$$

Alternatively, p. 592 defines a deviance residual.

Diagnostics

- Your book suggests lowess smooths of residual plots (pp. 594–595), based on the identity $E(Y_i - \hat{\pi}_i) = E(e_i) = 0$ for Bernoulli data. You are looking for a line that is *approximately* zero, not perfectly zero. The line will have a natural increase/decrease at either end if there are lots of zeros or ones – e.g. last two plots on p. 595.
- Residual plots for individual predictors might show curvature; adding quadratic terms or interactions can improve fit.
- An overall plot is a smoothed r_{SP_j} versus the linear predictor $\hat{\eta}_j = \hat{\boldsymbol{\beta}}'\mathbf{x}_j$. This plot will tell you if the model tends to over or underpredict the observed data for ranges of the linear predictor.
- You can look at individual r_{SP_j} to determine model fit. For the crab data, this might flag some individual crabs as ill-fit or unusual relative to the model. I usually flag $|r_{SP_j}| > 3$ as being ill-fit by the model.

Influential observations

Unlike linear regression, the leverage $\hat{h}_j$ in logistic regression depends on the model fit $\hat{\boldsymbol{\beta}}$ as well as the covariates $\mathbf{x}_j$. Points that have extreme predictor values $\mathbf{x}_j$ may not have high leverage $\hat{h}_j$ if $\hat{\pi}_j$ is close to 0 or 1. Here are the influence diagnostics available in PROC LOGISTIC:

- Leverage $\hat{h}_j$. Still may be useful for detecting "extreme" predictor values $\mathbf{x}_j$.
- $c_j = r_{SP_j}^2 \hat{h}_j / [p(1 - \hat{h}_j)^2]$ measures the change in the joint confidence region for $\boldsymbol{\beta}$ when j is left out (Cook's distance).
- DFBETA$_{js}$ is the standardized change in $\hat{\beta}_s$ when observation j is left out.
- The change in the X^2 GOF statistic when obs. j is left out is DIFCHISQ$_j = r_{SP_j}^2/(1 - \hat{h}_j)$. ($\Delta X_j^2$ in your book)

I suggest simply looking at plots of c_j vs. j.

Diagnostics/influence in crab data

```
proc logistic data=crabs descending; class color / param=ref;
model y(event='1')=color width;
output out=diag1 stdreschi=r xbeta=eta p=p c=c;

proc sgscatter data=diag1;
title "Crab data diagnostic plots";
plot r*(width eta) r*color / loess;

proc sgscatter data=diag1;
title "Crab data diagnostic plots";
plot (r c)*id;

proc sort; by color width;
proc sgplot data=diag1;
title1 "Predicted probabilities";
series x=width y=p / group=color;
yaxis min=0 max=1;

proc print data=diag1(where=(c>0.3 or r>3 or r<-3));
var y width color c r;
```

Case study, mosquito infection (pp. 573–575)

Study of a disease outbreak spread by mosquitoes. Y_i is whether the ith person got the disease. x_{i1} is the person's age, x_{i2} is socioeconomic status (1=upper, 2=middle, 3=lower), and x_{i3} is sector (0=sector 1 and 1=sector 2).

```
proc logistic data=disease;
 class ses sector / param=ref;
 model disease(event='1')=age ses sector / lackfit;
```

Note smoothed residual plot vs. age! Try backwards elimination from full interaction model.

Prediction Rules

It seems intuitive to base prediction of an outcome given $\mathbf{x}$ upon the following rule:

$$If\, \hat{\pi}_{\mathbf{x}} > 0.5, \text{then } \hat{Y}_{\mathbf{x}} = 1; \text{ else } \hat{Y}_{\mathbf{x}} = 0.$$

We can create a 2-way table of $\hat{Y}_{\mathbf{x}}$ vs. $Y_{\mathbf{x}}$ for any given threshold value of $\hat{\pi}_{\mathbf{x}}$ and readily visualize two types of classification errors: $\hat{Y}_{\mathbf{x}} = 1$ when $Y_{\mathbf{x}} = 0$, and $\hat{Y}_{\mathbf{x}} = 0$ when $Y_{\mathbf{x}} = 1$. A best classification rule would minimize the sum of these classification errors.

Prediction rules–Example

Snoring Data (Agresti 2013)

Snoring (x)	$Y_x = 0$	$Y_x = 1$	$\hat{\pi}_x$
0	1355	24	0.0205
2	603	35	0.0443
4	192	21	0.0931
5	224	30	0.1324

Prediction rules–Example

Assume our threshold is 0.0205. Then if $\hat{\pi}_x > 0.0205$, $\hat{Y}_x = 1$; else $\hat{Y}_x = 0$.

	$Y_x = 0$	$Y_x = 1$	
$\hat{Y}_x = 0$	1355	24	1379
$\hat{Y}_x = 1$	603+192+224=1019	35+21+30=86	1105
	2374	110	

From the table, we can compute

$$\hat{P}(\hat{Y}_x = 0 | Y_x = 1) = \frac{24}{110} = 0.218 = 1 - \hat{P}(\hat{Y}_x = 1 | Y_x = 1)$$

$$\hat{P}(\hat{Y}_x = 1 | Y_x = 0) = \frac{1019}{2374} = 0.429 = 1 - \hat{P}(\hat{Y}_x = 0 | Y_x = 0)$$

ROC Curve

ROC (Receiver Operating Characteristic) curves are created by plotting the *sensitivity* ($P(\hat{Y}_{\mathbf{x}} = 1 | Y_{\mathbf{x}} = 1)$) versus 1-*specificity* ($1 - P(\hat{Y}_{\mathbf{x}} = 0 | Y_{\mathbf{x}} = 0)$) over ordered unique values of $\hat{\pi}_{\mathbf{x}}$.

- Often, an optimal choice of x for classifying responses is found by observing the first point on the ROC curve that touches a line with slope 1 as the line's y-intercept decreases from 1.
- The area under the ROC curve is a measure of the model's predictive power.
- In our example, only one classifier has good sensitivity, but its specificity is poor. All the other classifiers have good specificity, but poor sensitivity.

Fitting logistic regression models (pp. 564–565)

The data are $(\mathbf{x}_j, Y_{\cdot j})$ for $j = 1, \ldots, c$.
The model is

$$Y_{\cdot j} \sim \text{bin}\left(n_j, \frac{e^{\boldsymbol{\beta}'\mathbf{x}_j}}{1 + e^{\boldsymbol{\beta}'\mathbf{x}_j}}\right).$$

The pmf of $Y_{\cdot j}$ in terms of $\boldsymbol{\beta}$ is

$$p(y_j; \boldsymbol{\beta}) = \begin{pmatrix} n_j \\ y_{\cdot j} \end{pmatrix} \left[\frac{e^{\boldsymbol{\beta}'\mathbf{x}_j}}{1 + e^{\boldsymbol{\beta}'\mathbf{x}_j}}\right]^{y_{\cdot j}} \left[1 - \frac{e^{\boldsymbol{\beta}'\mathbf{x}_j}}{1 + e^{\boldsymbol{\beta}'\mathbf{x}_j}}\right]^{n_j - y_{\cdot j}}.$$

The likelihood is the product of all N of these and the log-likelihood simplifies to

$$L(\boldsymbol{\beta}) = \sum_{k=1}^{p} \beta_k \sum_{j=1}^{c} y_{\cdot j} x_{jk} - \sum_{j=1}^{c} \log\left[1 + \exp\left(\sum_{k=1}^{p} \beta_k x_{jk}\right)\right] + \text{constant}.$$

Inference

The likelihood (or score) equations are obtained by taking partial derivatives of $L(\boldsymbol{\beta})$ with respect to elements of $\boldsymbol{\beta}$ and setting equal to zero. Newton-Raphson is used to get $\hat{\boldsymbol{\beta}}$, see the following optional slides if interested.

The inverse of the covariance of $\hat{\boldsymbol{\beta}}$ has ij^{th} element

$$-\frac{\partial^2 L(\boldsymbol{\beta})}{\partial \beta_i \partial \beta_j} = \sum_{s=1}^{N} x_{si} x_{sj} n_s \pi_s (1 - \pi_s),$$

where $\pi_s = \frac{e^{\boldsymbol{\beta}' \mathbf{x}_s}}{1+e^{\boldsymbol{\beta}' \mathbf{x}_s}}$. The *estimated* covariance matrix $\widehat{\text{cov}}(\hat{\boldsymbol{\beta}})$ is obtained by replacing $\boldsymbol{\beta}$ with $\hat{\boldsymbol{\beta}}$. This can be rewritten

$$\widehat{\text{cov}}(\hat{\boldsymbol{\beta}}) = \{\mathbf{X}' \text{diag}[n_j \hat{\pi}_j (1 - \hat{\pi}_j)] \mathbf{X}\}^{-1}.$$

How to get the estimates? (Optional...)

Newton-Raphson in one dimension: Say we want to find where $f(x) = 0$ for differentiable $f(x)$. Let x_0 be such that $f(x_0) = 0$. Taylor's theorem tells us

$$f(x_0) \approx f(x) + f'(x)(x_0 - x).$$

Plugging in $f(x_0) = 0$ and solving for x_0 we get $\hat{x}_0 = x - \frac{f(x)}{f'(x)}$. Starting at an x near x_0, $\hat{x}_0$ should be closer to x_0 than x was. Let's iterate this idea t times:

$$x^{(t+1)} = x^{(t)} - \frac{f(x^{(t)})}{f'(x^{(t)})}.$$

Eventually, if things go right, $x^{(t)}$ should be close to x_0.

Higher dimensions

If $\mathbf{f}(\mathbf{x}) : \mathbb{R}^p \to \mathbb{R}^p$, the idea works the same, but in vector/matrix terms. Start with an initial guess $\mathbf{x}^{(0)}$ and iterate

$$\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)} - [D\mathbf{f}(\mathbf{x}^{(t)})]^{-1}\mathbf{f}(\mathbf{x}^{(t)}).$$

If things are "done right," then this should converge to $\mathbf{x}_0$ such that $\mathbf{f}(\mathbf{x}_0) = \mathbf{0}$.
We are interested in solving $DL(\boldsymbol{\beta}) = \mathbf{0}$ (the score, or likelihood equations!) where

$$DL(\boldsymbol{\beta}) = \begin{bmatrix} \frac{\partial L(\boldsymbol{\beta})}{\partial \beta_1} \\ \vdots \\ \frac{\partial L(\boldsymbol{\beta})}{\partial \beta_p} \end{bmatrix} \text{ and } D^2 L(\boldsymbol{\beta}) = \begin{bmatrix} \frac{\partial L(\boldsymbol{\beta})}{\partial \beta_1^2} & \cdots & \frac{\partial L(\boldsymbol{\beta})}{\partial \beta_1 \partial \beta_p} \\ \vdots & \ddots & \vdots \\ \frac{\partial L(\boldsymbol{\beta})}{\partial \beta_p \partial \beta_1} & \cdots & \frac{\partial L(\boldsymbol{\beta})}{\partial \beta_p^2} \end{bmatrix}.$$

Newton-Raphson

So for us, we start with $\boldsymbol{\beta}^{(0)}$ (maybe through a MOM or least squares estimate) and iterate

$$\boldsymbol{\beta}^{(t+1)} = \boldsymbol{\beta}^{(t)} - [D^2 L(\boldsymbol{\beta})(\boldsymbol{\beta}^{(t)})]^{-1} DL(\boldsymbol{\beta}^{(t)}).$$

The process is typically stopped when $|\boldsymbol{\beta}^{(t+1)} - \boldsymbol{\beta}^{(t)}| < \epsilon$.

- Newton-Raphson uses $D^2 L(\boldsymbol{\beta})$ as is, with the **y** plugged in.
- Fisher scoring instead uses $E\{D^2 L(\boldsymbol{\beta})\}$, with expectation taken over **Y**, which is *not* a function of the observed **y**, but harder to get.
- The latter approach is harder to implement, but conveniently yields $\widehat{\text{cov}}(\hat{\boldsymbol{\beta}}) \approx [-E\{D^2 L(\boldsymbol{\beta})\}]^{-1}$ evaluated at $\hat{\boldsymbol{\beta}}$ when the process is done.