36-720: Logistic Regression and Logit Models

Brian Junker

October 1, 2007

- The Logistic Regression Model
- Example: Logistic Regression
- Interpreting the Coefficients
- Example: Variable Selection

[Christensen, Ch 4]

The Logistic Regression Model

Unlike the log-linear modeling problems we have considered so far in the course there are many problems in which one variable is clearly a "response" variable, and the others are "predictor" variables.

We will concentrate on the case where the response Y is a *binary* outcome, $y = 0$ or $y = 1$, and we have discrete or continuous predictor variables $x_1, x_2, \ldots, x_q$.

We begin again by analogy with the normal-distribution linear model:

$$y_i \overset{indep}{\sim} N(m_i, \sigma^2)\,, \quad \text{where } m_i = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_q x_{iq} \qquad (i = 1, \ldots, I).$$

Recall that for *contingency tables*, the y_i were non-negative counts (we called them n_i) and we considered alternative models such as

$$\begin{aligned} (y_1, \ldots, y_I) &\sim Poiss(m_1) \times \cdots \times Poiss(m_I)\,, \\ (y_1, \ldots, y_I) &\sim Multinom(N, (p_1, \ldots, p_I)) \\ &etc. \end{aligned}$$

where $N = \sum_i y_i$, $m_i = N p_i$, and m_i followed a log-linear model

$$\log m_i = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_q x_{iq}$$

When the y's are *binary* a different transformation is needed, to account for range restrictions, heterogeneous error variances, and other problems that would be encountered if we tried ordinary regression when $y_i = 0$ or 1.

Again we model as

$$y_i \sim \text{some distribution depending on the mean } p_i = E[Y_i]$$

and since $p_i \in [0, 1]$, we often use an s-shaped function to stretch p_i out to the whole real line (so unrestricted linear modeling is possible). Some common choices are:

- *Tangent* transformation: $\theta_i = \tan(\pi \cdot (p_i - \frac{1}{2}))$
- *Probit* transformation: $\theta_i = \Phi^{-1}(p_i)$, where $\Phi(z)$ is the $N(0, 1)$ cdf (a.k.a. "Normal ogive").
- *Logit* transformation: $\theta_i = \log \frac{p_i}{1-p_i}$ = log-odds.

We will concentrate on the logit form $\theta_i = \log \frac{p_i}{1-p_i}$. It is also known by the inverse transformation, the *logistic* transformation, $p_i = \frac{\exp \theta_i}{1+\exp \theta_i}$.

Since $\log p/(1 - p)$ is the natural parameter (think *exponential families*!) for the bernoulli and binomial distributions, then our model for binary response variables y_i will be

$$y_i \overset{indep}{\sim} Bernoulli(p_i)$$

where $p_i = E[y_i]$ is modeled via *logistic regression*:

$$\log \frac{p_i}{1 - p_i} = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_q x_{iq}$$

If there are n_i cases with the same covariate values $x_{i1}, \ldots, x_{iq}$, then we can also build a Binomial model

$$y_{i+} \overset{indep}{\sim} Binom(n_i, p_i)$$

(where y_{i+} is the number of 1's for that set of fixed covariates), and a similar logistic regression model.

Note that the Bernoulli model above is a special case.

Logistic regression models are again generalized linear models (see Ch's 9 and 11 of Christensen) and so can be fitted with `glm(..., family=binomial)` in R.

Many of the model-selection strategies that we talked about before work in this case too.

It is still meaningful t o look at Cook's distances, which are calculated using the same sorts of approximations that we talked about for log-linear Cook's distances.

Residual analysis is a little troublesome. For example, one version of the standardized residuals for the "Bernoulli" version of the model look like this

$$r_i = \frac{y_i - \hat{p}_i}{\sqrt{\hat{p}_i(1-\hat{p}_i)}}$$

Both the numerator and the denominator can cause problems:

- If $|y_i - \hat{p}_i| \approx 1$, then the residual will be enormous (the *denominator* is close to zero!).
- Even if p_i is not close to 0 or 1, the fact that y_i can only be zero or one in the *numerator* usually causes lots of noticable "streaking" in residual plots, even if the model fits well.

Similar problems exist with deviance residuals, etc. These problems also cause the residuals to not look very good in normal probability plots.

Example: Logistic Regression

Mosteller and Tukey (1977) collected data on average verbal test scores for 6th graders at 20 mid-Atlantic schools taken from *The Coleman Report*:

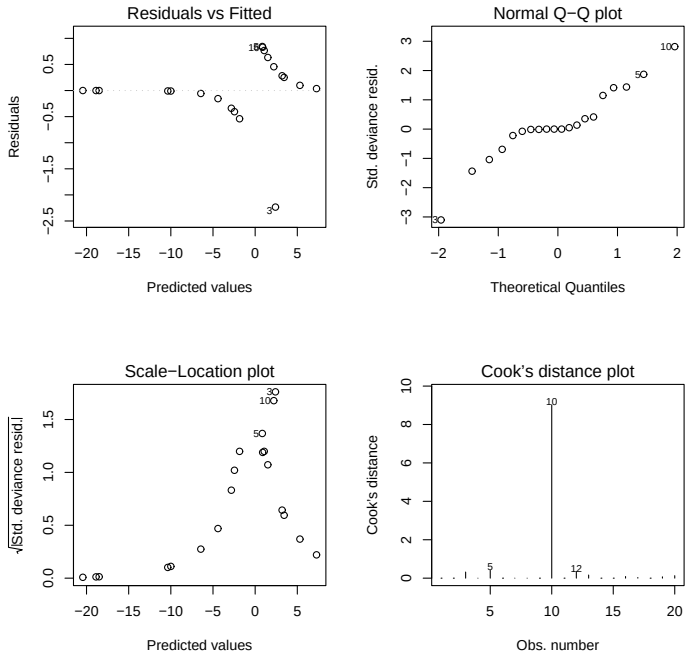
	X1	X2	X3	X4	X5	Y	Z
1	3.83	28.87	7.20	26.60	6.19	1	37.01
2	2.89	20.10	-11.71	24.40	5.17	0	26.51
3	2.86	69.05	12.32	25.70	7.04	0	36.51
4	2.92	65.40	14.28	25.70	7.10	1	40.70
5	3.06	29.59	6.31	25.40	6.15	1	37.10
6	2.07	44.82	6.16	21.60	6.41	0	33.90
7	2.52	77.37	12.70	24.90	6.86	1	41.80
8	2.45	24.67	-0.17	25.01	5.78	0	33.40
9	3.13	65.01	9.85	26.60	6.51	1	41.01
10	2.44	9.99	-0.05	28.01	5.57	1	37.20
11	2.09	12.20	-12.86	23.51	5.62	0	23.30
12	2.52	22.55	0.92	23.60	5.34	0	35.20
13	2.22	14.30	4.77	24.51	5.80	0	34.90
14	2.67	31.79	-0.96	25.80	6.19	0	33.10
15	2.71	11.60	-16.04	25.20	5.62	0	22.70
16	3.14	68.47	10.62	25.01	6.94	1	39.70
17	3.54	42.64	2.66	25.01	6.33	0	31.80
18	2.52	16.70	-10.99	24.80	6.01	0	31.70
19	2.68	86.27	15.03	25.51	7.51	1	43.10
20	2.37	76.73	12.77	24.51	6.96	1	41.01

Here, X1 = staff salaries per pupil; X2 = percent of fathers in white collar jobs, X3 = socioeconomic status, X4 = average verbal test scores for *teachers* at each school, X5 = (mothers' years of schooling)/2, Z = mean verbat test scores for *students* at each school; and Y = 1 if Z > 37 and Y = 0 if not (so Y is a cutoff that might be used to evaluate school performance).

Let us begin by fitting an additive (main effects only) logistic regression model to y in the above data.

```
> schools <- read.table("mosteller-tukey.txt")
> summary(fit0 <- glm(y ~ x1 + x2 + x3 +x4 +x5,data=schools,family=binomial))

Call:
glm(formula = y ~ x1 + x2 + x3 + x4 + x5, family = binomial,
    data = schools)

Deviance Residuals:
       Min          1Q      Median          3Q         Max
-2.234e+00  -8.112e-02  -8.213e-05   3.263e-01   8.441e-01

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -4.5635    33.1771  -0.138    0.891
x1            2.1346     3.3235   0.642    0.521
x2            0.1135     0.1592   0.713    0.476
x3            0.9789     0.8487   1.153    0.249
x4            2.0242     1.3251   1.528    0.127
x5          -10.0928     9.7992  -1.030    0.303

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 27.526  on 19  degrees of freedom
Residual deviance:  8.343  on 14  degrees of freedom
```

As the next slide shows, the residuals look terrible, but the fit is apparently already pretty good: deviance=8.3 on 14 df.



Interpreting the Coefficients

For *log-linear models*, main-effects only corresponded to the model of independence, since the model for m_i's (or p_i's) would then be multiplicative:

$$\begin{aligned} m_i &= \exp\left\{\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots \beta_q x_{iq}\right\} \\ &= a_0 \cdot a_1^{x_{i1}} \cdot a_2^{x_{i2}} \cdots a_q^{x_{iq}} \end{aligned}$$

For *logistic regression models*, the relationship isn't so straightforward

$$p_i = \frac{\exp\left\{\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots \beta_q x_{iq}\right\}}{1 + \exp\left\{\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots \beta_q x_{iq}\right\}}$$

but there is a nice interpretation in terms of odds ratios (see next slide)...

Let $p(x_1, \ldots, x_j, \ldots, x_q)$ be the success probability corresponding to covariate values $(x_1, \ldots, x_j, \ldots, x_q)$. Then

$$\log \frac{p(x_1, \ldots, x_j, \ldots, x_q)}{1 - p(x_1, \ldots, x_j, \ldots, x_q)} = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_j x_j + \cdots + \beta_q x_{iq}$$

and in particular if we increase the j^{th} covariate by one unit, then

$$\begin{aligned} &logOR[p(x_1, \ldots, (x_j + 1), \ldots, x_q), p(x_1, \ldots, x_j, \ldots, x_q)] \\ &= (\beta_0 - \beta_0) + \cdots + \beta_j((x_j + 1) - x_j) + \cdots + \beta_q(x_q - x_q) \\ &= \beta_j \end{aligned}$$

so that β_j is the log-odds-ratio for a successful outcome, if we increase the covariate x_j by 1 unit.

Equivalently β_j is the increase in odds of a successful outcome for an increase of 1 unit in x_j (holding the other x's fixed).

Reading off the coefficients table in the example,

- If we increase staff salaries per pupil by 1 unit, the model predicts an increase in log-odds of a successful school increase by 2.13;
- If we increase percent of fathers in white collar jobs, the model predicts an increase in odds of a successful school increase by 0.11; etc.

Since β_j is a log-odds-ratio between x_j and the outcome y, when β_j is (insignificantly different from) zero, we can infer that y and x_j are independent, conditional on the other x's in the model.

In the case of the model in this example, *none* of the coefficients are significantly different from zero! The problems here are the same as you might encounter in a conventional regression analysis

- Small sample size—only 20 observations
- Collinearity in the x's—indeed

  ```
  > X <- model.matrix(fit0)
  > round(eigen(t(X)%*%X)$values,2)
  [1] 57561.25  3701.05   465.95     3.12     1.82     0.02
  ```

  so we expect to see at most two or three significant predictors after variable selection.

Example: Variable Selection

We might try to seek a better model using stepwise variable selection or some other model selection heuristic.

```
> library(MASS)
> basemodel <- glm(y ~ x1 + x2 + x3 +x4 + x5 ,data=schools,family=binomial)
> fit1 <- eval(stepAIC(basemodel,
          scope=list(lower=.~ 1,upper=.~x1 + x2 + x3 +x4 + x5,k=2))$call)

> anova(fit1,fit0,test="Chisq")
Analysis of Deviance Table

Model 1: y ~ x3 + x4
Model 2: y ~ x1 + x2 + x3 + x4 + x5
  Resid. Df Resid. Dev Df Deviance P(>|Chi|)
1        17    10.1414
2        14     8.3429  3   1.7984    0.6153
```

So it looks like the model involving only X3=SES and X4=teachers' verbal scores does about as well as the model involving all five main effects. Examining the coefficient

```
> summary(fit1)$coefficients
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -41.8188    24.5239  -1.705   0.0882 .
x3            0.3646     0.1799   2.027   0.0426 *
x4            1.5615     0.9428   1.656   0.0977 .
```

we see that both SES and teachers' verbal scores have positive effects on the log-odds of a successful school ($y = 1$).

If we try to expand the model search to consider interactions of all orders, something interesting happens:

```
> fit2 <- eval(stepAIC(basemodel,
          scope=list(lower=.~ 1,upper=.~(x1 + x2 + x3 +x4 + x5)^5,k=2))$call)
[...]
  y ~ x3 + x4 + x5 + x4:x5
[...]
  There were 50 or more warnings (use warnings() to see the first 50)

> warnings()
Warning messages:
1: algorithm did not converge in:
   glm.fit(X, y, wt, offset = object$offset, family = object$family,   ...
2: fitted probabilities numerically 0 or 1 occurred in:
   glm.fit(X, y, wt, offset = object$offset, family = object$family,   ...
[etc.]
```

Comparing `fitted(fit2)` to the actual y's you will see that they agree to 8 or more decimal places—in other words, the model `y ~ x3 + x4*x5` is essentially already the *saturated model*, with $|y_i - \hat{p}_i| \approx 0$.

But with $\hat{p}_i$ so close to zero or one, the left hand side of

$$\log \frac{\hat{p}_i}{1 - \hat{p}_i} = \hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \cdots + \hat{\beta}_q x_{iq}$$

is essentially infinite, and no MLE's $\hat{\beta}_j$ can be found (hence "`algorithm did not converge`" above).

This is a standard problem with saturated logistic regression models.

Finally we recall that y is a dichotomized version of z: $y = 1$ if $z > 37$ and $y = 0$ otherwise. What happens if we try variable selection on the usual normal-errors linear regression model for z?

```
> basemodel <- lm(z ~ x1 + x2 + x3 +x4 + x5 ,data=schools)
> norm1 <- eval(stepAIC(basemodel,
           scope=list(lower=.~ 1,upper=.~(x1 + x2 + x3 +x4 + x5)^5),
           k=2)$call) # k=2 for AIC
> norm1$call
lm(formula = z ~ x1 + x3 + x4, data = schools)
> norm2 <- eval(stepAIC(basemodel,
           scope=list(lower=.~ 1,upper=.~(x1 + x2 + x3 +x4 + x5)^5),
           k=log(20))$call) # k=log(sample size) for BIC
> norm2$call
lm(formula = z ~ x3 + x4, data = schools)
```

Even though the stepwise procedure had access to interactions of all orders, the interaction `x4*x5` was not in the final model for `z`.

This suggests that the `x4*x5` interaction was useful for predicting the simpler response surface of `y` (dichotomized `z`) than for predicting the more complex response surface of `z` itself.

We should dichotomize with care, and then only if the substantive question requires it.

- Dichotomization always changes the information in the data.
- If you must dichotomize, I'd suggest doing a sensitivity analysis (try different dichotomizations and see how that affects the results).