Numerical Issues in Statistical Computing for the Social Scientist

Micah Altman

Jeff Gill

Michael McDonald

A Wiley-Interscience Publication

JOHN WILEY & SONS, INC.

New York / Chichester / Weinheim / Brisbane / Singapore / Toronto

9

Convergence Problems in Logistic Regression

Paul Allison

9.1 INTRODUCTION

Anyone who has much practical experience with logistic regression will have occasionally encountered problems with convergence. Such problems are usually both puzzling and exasperating. Most researchers haven't a clue as to why certain models and certain data sets lead to convergence difficulties. And for those who do understand the causes of the problem, it's usually unclear whether and how the problem can be fixed.

In this chapter, I explain why numerical algorithms for maximum likelihood estimation of the logistic regression model sometimes fail to converge. And I will also consider a number possible solutions. Along the way, I will look at the performance of several popular computing packages when they encounter convergence problems of varying kinds.

9.2 OVERVIEW OF LOGISTIC MAXIMUM LIKELIHOOD ESTIMATION

I begin with a review of the logistic regression model and maximum likelihood estimation of the parameters of that model. For a sample of n cases ($i = 1, \ldots, n$), there are data on a dummy dependent variable y_i (with values of 1 and 0) and a vector of explanatory variables $\mathbf{x}_i$ (including a 1 for the intercept term). The logistic regression

model states that:

$$\Pr(y_i = 1|x_i) = \frac{1}{1 + \exp(-\boldsymbol{\beta}\mathbf{x}_i)} \tag{9.1}$$

where $\boldsymbol{\beta}$ is a vector of coefficients. Equivalently, we may write the model in "logit" form:

$$\ln\left[\frac{\Pr(y_i = 1|x_i)}{\Pr(y_i = 0|x_i)}\right] = \boldsymbol{\beta}\mathbf{x}_i \tag{9.2}$$

Assuming that the n cases are independent, the log-likelihood function for this model is:

$$\ell(\boldsymbol{\beta}) = \sum_i \boldsymbol{\beta}\mathbf{x}_i y_i - \sum_i \ln[1 + \exp(\boldsymbol{\beta}\mathbf{x}_i)] \tag{9.3}$$

The goal of maximum likelihood estimation is to find a set of values for $\boldsymbol{\beta}$ that maximize this function. One well-known approach to maximizing a function like this is to differentiate it with respect to $\boldsymbol{\beta}$, set the derivative equal to 0, and then solve the resulting set of equations. The first derivative of the log-likelihood is:

$$\frac{\partial \ell(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} = \sum_i x_i y_i - \sum_i x_i \hat{y}_i, \tag{9.4}$$

where $\hat{y}_i$ is the predicted value of y_i:

$$\hat{y}_i = \frac{1}{1 + \exp(-\boldsymbol{\beta}\mathbf{x}_i)}. \tag{9.5}$$

The next step is to set the derivative equal to 0 and solve for $\boldsymbol{\beta}$:

$$\sum_i \mathbf{x}_i y_i - \sum_i \mathbf{x}_i \hat{y}_i. \tag{9.6}$$

Because $\boldsymbol{\beta}$ is a vector, (9.6) is actually a set of equations, one for each of the parameters to be estimated. These equations are identical to the "normal" equations for least-squares linear regression, except that by (9.4), $\mathbf{y}$ is a non-linear function of the $\mathbf{x}_i$'s rather than a linear function.

For some models and data (e.g., "saturated" models), the equations in (9.6) can be explicitly solved for the ML estimator $\boldsymbol{\beta}$. For example, suppose there is a single dichotomous $\mathbf{x}$ variable, so that the data can be arrayed in a 2×2 table, with observed cell frequencies f_{11}, f_{12}, f_{21}, and f_{22}. Then the ML estimator of the coefficient of $\mathbf{x}$ is given by the logarithm of the "cross-product ratio":

$$\hat{\beta} = \log\left[\frac{f_{11} f_{22}}{f_1 2 f_{21}}\right]. \tag{9.7}$$

For most data and models, however, the equations in (9.6) have no explicit solution. In such cases, the equations must be solved by numerical methods, of which there are many. The most popular numerical method is the Newton-Raphson algorithm. Let

$\mathbf{U}(\boldsymbol{\beta})$ be the vector of first derivatives of the log-likelihood with respect to $\boldsymbol{\beta}$ and let $\mathbf{I}(\boldsymbol{\beta})$ be the matrix of second derivatives. That is,

$$\mathbf{U}(\boldsymbol{\beta}) = \frac{\partial \ell(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} = \sum_i \mathbf{x}_i y_i - \sum_i \mathbf{x}_i \hat{y}_i$$

$$\mathbf{I}(\boldsymbol{\beta}) = \frac{\partial^2 \ell(\boldsymbol{\beta})}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}'} = \sum_i \mathbf{x}_i \mathbf{x}_i' \hat{y}_i (1 - \hat{y}_i) \tag{9.8}$$

The vector of first derivatives $U(\boldsymbol{\beta})$ is sometimes called the *gradient* or *score* while the matrix of second derivatives $I(\boldsymbol{\beta})$ is called the *Hessian*. The Newton-Raphson algorithm is then

$$\boldsymbol{\beta}_{j+1} = \boldsymbol{\beta}_j - \mathbf{I}^{-1}(\boldsymbol{\beta}_j)\mathbf{U}(\boldsymbol{\beta}_j) \tag{9.9}$$

where $\mathbf{I}^{-1}$ is the inverse of $\mathbf{I}$. Chapter 4 of this volume showed what can go wrong with this process as well as some remedies.

To operationalize this algorithm, a set of starting values $\boldsymbol{\beta}_0$ is required. Choice of starting values is not critical; usually, setting $\boldsymbol{\beta}_0 = \mathbf{0}$ works fine. The starting values are substituted into the right-hand side of (9.9), which yields the result for the first iteration, $\boldsymbol{\beta}_1$, These values are then substituted back into the right hand side, the first and second derivatives are recomputed, and the result is $\boldsymbol{\beta}_2$. The process is repeated until the maximum change in each parameter estimate from one iteration to the next is less than some criterion, at which point we say that the algorithm has converged. Once we have the results of the final iteration, $\hat{\boldsymbol{\beta}}$, a byproduct of the Newton-Raphson algorithm is an estimate of the covariance matrix of the coefficients, which is just $-\mathbf{I}^{-1}(\boldsymbol{\beta})$. Estimates of the standard errors of the coefficients are obtained by taking the square roots of the main diagonal elements of this matrix.

9.3 WHAT CAN GO WRONG?

A problem that often occurs in trying to maximize a function is that the function may have local maxima, that is, points that are larger than any nearby point but not as large as some more distant point. In those cases, setting the first derivative equal to zero will yield equations that have more than one solution. And if the starting values for the Newton-Raphson algorithm are close to a local maximum, the algorithm will most likely iterate to that point rather than the global maximum. Fortunately, problems with multiple maxima cannot occur with logistic regression because the log-likelihood is globally concave, which means that the function can have at most one maximum.

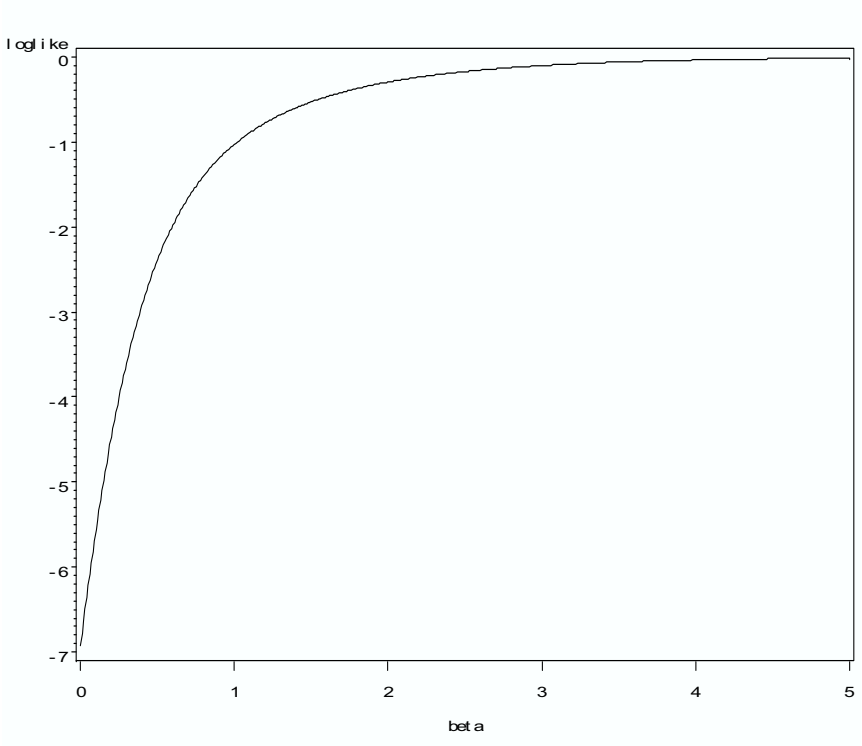
Unfortunately, there are many situations in which the likelihood function has *no* maximum, in which case we say that the maximum likelihood estimate does not exist. Consider the set of data on 10 observations in Table 9.1.

For these data, it can be shown that the ML estimate of the intercept is 0. Figure 9.1 shows a graph of the log-likelihood as a function of the slope "beta". It is apparent that, although the log-likelihood is bounded above by 0, it does not reach a maximum as beta increases. We can make the log-likelihood as close to 0 as we choose by making beta sufficiently large. Hence, there is no maximum likelihood estimate.

Table 9.1 Data Exhibiting Complete Separation

x	y	x	y
-5	0	1	1
-4	0	2	1
-3	0	3	1
-2	0	2	1
-1	0	5	1

Fig. 9.1 Log Likelihood Versus Beta



This is an example of a problem known as *complete separation* (Albert and Anderson 1984), which occurs whenever there exists some vector of coefficients $\mathbf{b}$ such that $\mathbf{y}_i = 1$ whenever $\mathbf{bx}_i > 0$ and $y_i = 0$ whenever $\mathbf{bx}_i < 0$. In other words, complete separation occurs whenever a linear function of $\mathbf{x}$ can generate perfect predictions of $\mathbf{y}$. For our hypothetical data set, a simple linear function that satisfies this property is $0 + 1(\mathbf{x})$. That is, when $\mathbf{x}$ is greater than 0, $\mathbf{y} = 1$, and when $\mathbf{x}$ is less than 0, $\mathbf{y} = 0$.

A related problem is known as *quasi-complete separation*. This occurs when (a) there exists some coefficient vector $\mathbf{b}$ such that $\mathbf{bx}_i \geq 0$ whenever $\mathbf{y}_i = 1$, and $\mathbf{bx}_i \leq 0$ whenever $\mathbf{y}_i = 0$, and equality holds for at least one case in each category of the dependent variable. Table9.2 displays a data set that satisfies this condition.

Table 9.2 Data Exhibiting Quasi-Complete Separation

x	y	x	y
-5	0	1	1
-4	0	2	1
-3	0	3	1
-2	0	2	1
-1	0	5	1
0	0	0	1

The difference between this data set and the previous one is that there are two more observations, each with *x* values of 0 but having different values of *y*.

The log-likelihood function for these data, shown in Figure 9.2, is similar in shape to that in Figure 9.1. However, the asymptote for the curve is not 0, but a number that is approximately -1.39. In general, the log-likelihood function for quasi-complete separation will not approach 0, but some number lower than that. In any case, the curve has no maximum so, again, the maximum likelihood estimate does not exist.

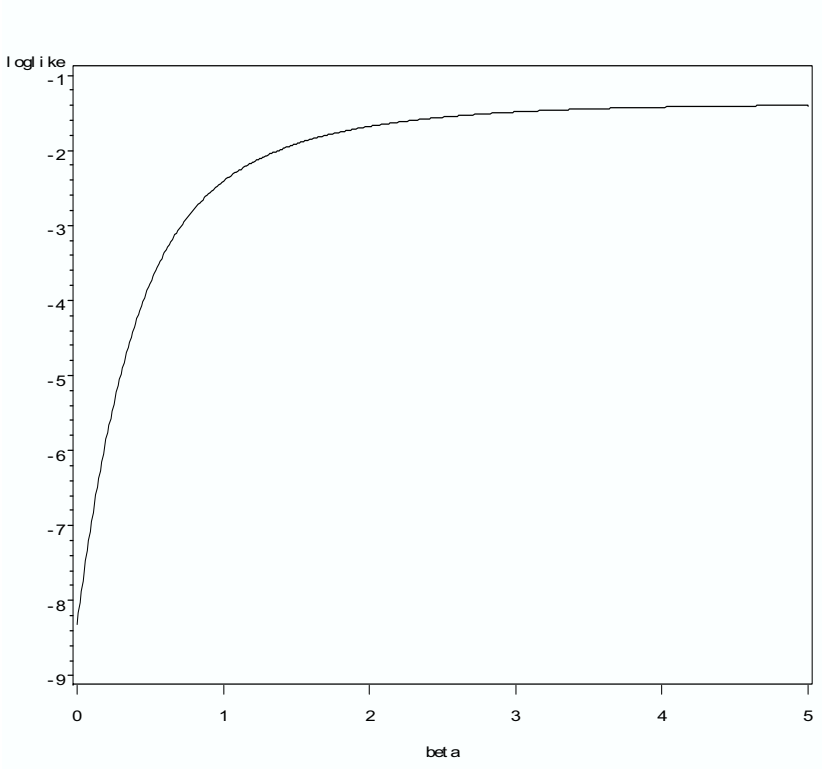
Of the two conditions, complete and quasi-complete separation, the latter is far more common. It most often occurs when an explanatory variable *x* is a dummy variable and, for one value of *x*, either every case has the event *y*=1 or every case has the event *y*=0. Consider the following 2×2 table:

		y	
		1	0
x	1	5	0
	0	15	10

If we form the linear combination $c = 0 + (1)\,x$, we have $c \geq 0$ when $y = 1$ and $c \leq 0$ when $y = 0$. Further, for all the cases in the second row, $c = 0$ for both values of y. So the conditions of quasi-complete separation are satisfied.

To get some intuitive sense of why this leads to non-existence of the maximum likelihood estimator, consider equation 9.7 which gives the maximum likelihood estimator of the slope coefficient for a 2×2 table. For our quasi-complete table, this would be undefined because there is a zero in the denominator. The same problem would occur if there were a zero in the numerator because the logarithm of zero is also undefined. If the table is altered to read:

Fig. 9.2 Log Likelihood as a Function of the Slope, Quasi-Complete Separation



		y	
		1	0
x	1	5	0
	0	0	10

then there is *complete* separation with zeros in both the numerator and the denominator. So the general principle is evident: Whenever there is a zero in any cell of a 2×2 table, the maximum likelihood estimate of the logistic slope coefficient does not exist. This principle also extends to multiple independent variables:

For any dichotomous independent variable in a logistic regression, if there is a zero in the 2×2 *table formed by that variable and the dependent variable, the ML estimate for the regression coefficient will not exist.*

This is by far the most common cause of convergence failure in logistic regression. Obviously, it is more likely to occur when the sample size is small. Even in large samples, it will frequently occur when there are extreme splits on the frequency distribution of either the dependent or independent variables. Consider, for example, a logistic regression predicting the occurrence of a rare disease. Suppose further,

that the explanatory variables include a set of seven dummy variables representing different age levels. It would not be terribly surprising if no one had the disease for at least one of the age levels, but this would produce quasi-complete separation.

9.4 BEHAVIOR OF THE NEWTON-RAPHSON ALGORITHM UNDER SEPARATION

We just saw that when there are explicit formulas for the maximum likelihood estimate and there is either complete or quasi-complete separation, the occurrence of zeros in the formulas prevents computation. What happens when the Newton-Raphson algorithm is applied to data exhibiting either kind of separation? That depends on the particular implementation of the algorithm. The classic behavior is this: at each iteration, the parameter estimate for the variable (or variables) with separation gets larger in magnitude. Iterations continue until the fixed iteration limit is exceeded. At whatever limit is reached, the parameter estimate is large and the estimated standard error is extremely large. If separation is complete, the log-likelihood will be reported as zero.

What actually happens depends greatly on the software implementation of the algorithm. One issue is the criterion used for determining convergence. Some older/cruder logistic regression programs determine convergence by examining the change in the log-likelihood from one iteration to the next. If that change is very small, convergence is declared and the iterations cease. Unfortunately, as seen in Figures 9.1 and 9.2, with complete or quasi-complete separation the log-likelihood may change imperceptibly from one iteration to the next even though the parameter estimate is rapidly increasing in magnitude. So, to avoid the false appearance of convergence, it is essential that convergence be evaluated by looking at changes in the parameter estimate across iterations, rather than changes in the log-likelihood.

At the other extreme, some logistic regression programs have algorithms that attempt to detect complete or quasi-complete separation, and then issue appropriate warnings to the user. Albert and Anderson (1984) proposed one "empirical" method that has been implemented in `PROC LOGISTIC` in SAS software (SAS Institute 1999). It has the following steps:

1. If the convergence criterion is satisfied within eight iterations, conclude that there is no problem.

2. For all iterations after the eighth, compute the predicted probability of the observed response for each observation, which is given by:

$$\hat{y}_i = \frac{1}{1 + \exp[(2y_i - 1)\hat{\boldsymbol{\beta}}\mathbf{x}_i]}$$

   If the predicted probability is one for all observations, conclude that there is complete separation and stop the iterations.

3. If the probability of the observed response is large (≥ 0.95) for some observations (but not all), examine the estimated standard errors for that iteration. If they exceed some criterion, conclude that there is quasi-complete separation and stop the iteration.

The check for complete separation is very reliable, but the check for quasi-complete separation is less so. For more reliable checks of quasi-complete separation, methods based on linear programming algorithms have been proposed by Albert and Anderson (1984) and Santner and Duffy (1986).

To determine how available software handles complete and quasi-complete separation, I tried estimating logistic regression models for the data sets in Table 9.1 and 9.2 using several popular statistical packages. For some packages (SAS, Stata), more than one command or procedure was evaluated. Keep in mind that these tests were run in September 2002 using software versions that were available to me at the time. Results are summarized in Table 9.3. Several criteria were used in the evaluation:

9.4.1 Warning Messages

Ideally, the program should detect the separation and issue a clear warning message to the user. The program that came closest to this ideal was the SAS procedure LOGISTIC. For complete separation, it printed the message:

```
                Complete separation of data points detected.

WARNING: The maximum likelihood estimate does not exist.
WARNING: The LOGISTIC procedure continues in spite of the above warning.
         Results shown are based on the last maximum likelihood iteration.
         Validity of the model fit is questionable.
```

For quasi-complete separation, the message was

```
                Quasicomplete separation of data points detected.

WARNING: The maximum likelihood estimate may not exist.
WARNING: The LOGISTIC procedure continues in spite of the above warning.
         Results shown are based on the last maximum likelihood iteration.
          Validity of the model fit is questionable.
```

SPSS did a good job for complete separation, presenting the warning:

```
Estimation terminated at iteration number 21 because a perfect
fit is detected. This solution is not unique. Warning \# 18582
Covariance matrix cannot be computed. Remaining statistics will
be omitted.
```

But SPSS gave no warning message for quasi-complete separation. For complete separation, the Stata LOGIT command gave the message:

```
outcome = x\texttt{>}-1 predicts data perfectly
```

For quasi-complete separation, the message was

```
outcome = x\texttt{>}0 predicts data perfectly except for x==0 subsample:
```

```
x dropped and 10 obs not used
```

For complete separation, Systat produced the message

```
Failure to fit model or perfect fit
```

but said nothing about the quasi-complete case.

Several other programs produced warning messages that were ambiguous or cryptic. SAS CATMOD marked certain coefficients estimates with # and said that they were "regarded to be infinite." JMP identified some coefficient estimates as "unstable." R said that some of the fitted probabilities were numerically 0 or 1. For quasi-complete separation, the GENMOD procedure in SAS reported that the negative of the Hessian matrix was not positive definite. And finally, several programs gave no warning messages whatsoever (GLIM, LIMDEP, Stata MLOGIT, Minitab).

9.5 FALSE CONVERGENCE

Strictly speaking, the Newton-Raphson algorithm should not converge under either complete or quasi-complete separation. Nevertheless, the only program that exhibited this classic behavior was Minitab. No matter how high I raised the maximum number of iterations, the program would not converge. With the exception of SAS LOGISTIC and Stata LOGIT (which stopped the iterations once separation had been detected), the remaining programs all reported that the convergence criterion had been met. In some cases (GLIM, R, SPSS, and Systat), it was necessary to increase the maximum iterations beyond the default in order to achieve this apparent convergence.

Lacking information on the convergence criterion for most of these programs, I don't have an explanation for the false convergence. One possibility is that the convergence criterion is based on the log-likelihood rather than the parameter estimates. But in the case of SAS GENMOD, SAS documentation is quite explicit that the criterion is based on parameter estimates. In any case, the combination of apparent convergence and lack of clear warning messages in many programs means that some users are likely to be misled about the viability of their parameter estimates.

9.6 REPORTING OF PARAMETER ESTIMATES AND STANDARD ERRORS

Some programs that do a good job of detecting and warning about complete separation then fail to report any parameter estimates or standard errors (SPSS, STATA LOGIT). This might seem sensible since non-convergent estimates are essentially worthless as parameter estimates. However, they may still serve a useful diagnostic purpose in determining which variables have complete or quasi-complete separation.

Table 9.3 Performance of Packages under Complete and Quasi-Complete Separation

	Warning Messages		False Convergence		Report Estimates		LR Statistics	
	Comp	Quasi	Comp	Quasi	Comp	Quasi	Comp	Quasi
GLIM			*	*	*	*		
JMP	A	A	*	*	*	*	*	*
LIMDEP			*	*	*	*		
Minitab					*	*		
R	A	A	*	*	*	*	*	*
SAS GENMOD		A	*		*	*	*	*
SAS LOGISTIC	C	C			*	*		
SAS CATMOD	A	A	*	*	*	*		
SPSS	C			*		*		
Stata LOGIT	C	C				*		
Stata MLOGIT			*	*	*	*		
Systat	C			*	*	*		

Note: C=clear warning, A=ambiguous warning.

9.6.1 Likelihood Ratio Statistics

Some programs (SAS GENMOD, JMP), can report optional likelihood-ratio chi-square tests for each of the coefficients in the model. Unlike Wald chi-squares, which are essentially useless under complete or quasi-complete separation, the likelihood ratio test is still a valid test of the null hypothesis that a coefficient is equal to 0. Thus, even if a certain parameter can't be estimated, we can still judge whether or not it is significantly different from 0.

9.6.1.1 Diagnosis of Separation Problems We are now in a position to make some recommendations about how the statistical analyst should detect problems of complete or quasi-complete separation. If you're using software that gives clear diagnostic messages (SAS LOGISTIC, STATA LOGIT), then half the battle is won. But there is still a need to determine which variables are causing the problem, and to get a better sense of the nature of the problem.

The second step (or the first step with programs that do not give good warning messages) is to carefully examine the estimated coefficients and their standard errors. Variables with non-existent coefficients will invariably have large parameter estimates, typically greater than 5.0, and huge standard errors, producing Wald chi-square statistics that are near 0. If any of these variables is a dummy (indicator) variable, the next step is to compute the 2×2 table for each dummy variable with the dependent variable. A frequency of zero in any single cell of the table means quasi-

complete separation. Less commonly, if there are two zeroes (diagonally opposed) in the table, the condition is complete separation.

Once you have determined which variables are causing separation problems, it's time to consider possible solutions. The potential solutions are somewhat different for complete and quasi-complete separation, so I will treat them separately. I begin with the more common problem of quasi-complete separation.

9.6.2 Solutions for Quasi-complete Separation

9.6.2.1 Deletion of Problem Variables In practice, the most widely used method for dealing with quasi-complete separation is simply to delete from the model any variables whose coefficients did not converge. *I do not recommend this method*. If a variable has quasi-complete separation with the dependent variable, it is natural to suspect that variable has a strong (albeit, non-infinite) effect on the dependent variable. Deleting variables with strong effects will certainly obscure the effects of those variables, and is also likely to bias the coefficients for other variables in the model.

9.6.2.2 Combining Categories As noted earlier, the most common cause of quasi-complete separation is a dummy predictor variable such that, for one level of the variable, either every observation has the event or no observation has the event. For those cases in which the problem variable is one of set of variables representing a single categorical variable, the problem can often be easily solved by combining categories. For example, suppose that marital status has five categories: never married, currently married, divorced, separated, and widowed. This variable could be represented by four dummy variables, with currently married as the reference category. Suppose, further, that the sample contains 50 persons who are divorced but only 10 who are separated. If the dependent variable is 1 for employed and 0 for unemployed, it's quite plausible that all 10 of the separated persons would be employed, leading to quasi-complete separation. A natural and simple solution is to combine the divorced and separated categories, turning two dummy variables into a single dummy variable.

Similar problems often arise when a quantitative variable, like age, is chopped into a set of categories, with dummy variables for all but one of the categories. Although this can be a useful device for representing non-linear effects, it can easily lead to quasi-complete separation if the number of categories is large and the number of cases within some categories is small. The solution is to use a smaller number of categories, or perhaps revert to the original quantitative representation of the variable.

If the dummy variable represents an irreducible dichotomy, like sex, then this solution is clearly not feasible. However, there is another simple method that often provides a very satisfactory solution.

9.6.2.3 Do Nothing and Report Likelihood Ratio Chi-Squares Just because maximum likelihood estimates don't exist for some coefficients because of quasi-complete separation, that doesn't mean they don't exist for other variables in the logistic regression model. In fact, if one leaves the offending variables in the

model, the coefficients, standard errors, and test statistics for the remaining variables are still valid maximum likelihood estimates. So one attractive strategy is just to leave the problem variables in the model. The coefficients for those variables could be reported as $+\infty$ or $-\infty$. The standard errors and Wald statistics for the problem variables will certainly be incorrect but, as noted above, likelihood ratio tests for the null hypothesis that the coefficient is zero are still valid. If these statistics are not available as options in the computer program, they can be easily obtained by fitting the model with and without each problem variable, then taking twice the positive difference in the log-likelihoods.

If the problem variable is a dummy variable, then the estimates you get for the non-problem variables have a special interpretation. They are the ML estimates for the subsample of cases who fall into the category of the dummy variable in which observations differ on the dependent variable. For example, suppose that the dependent variable is whether or not a person smokes cigars. A dummy variable for sex is included in the model, but none of the women smoke cigars, producing quasi-complete separation. If sex is left in the model, the coefficients for the remaining variables (e.g., age, income, education) represent the effects of those variables among men only. (This can easily be verified by actually running the model for men only). The advantage of doing it in the full sample with sex as a covariate is that one also gets a test of the sex effect (using the likelihood ratio chi-square) while controlling for the other predictor variables.

9.6.2.4 Exact Inference As previously mentioned, problems of separation are most likely to occur in small samples and/or when there is an extreme split on the dependent variable. Of course, even without separation problems, maximum likelihood estimates may not have good properties in small samples. One possible solution is to abandon maximum likelihood entirely and do exact logistic regression. This method was originally proposed by Cox (1970), but was not computationally feasible until the advent the `LogXact` program and, more recently, the introduction of exact methods to the `LOGISTIC` procedure in SAS.

Exact logistic regression is designed to produce exact *p*-values for the null hypothesis that each predictor variable has a coefficient of 0, conditional on all the other predictors. These *p*-values, based on permutations of the data rather than on large-sample chi-square approximations, are essentially unaffected by complete or quasi-complete separation. The coefficient estimates reported with this method are usually conditional maximum likelihood estimates, and these can break down when there is separation. In that event, both `LogXact` and `PROC LOGISTIC` report median unbiased estimates for the problem coefficients. If the true value is β, a median unbiased estimator β_u has the property

$$\Pr(\beta_u \leq \beta) \geq 1/2, \qquad \Pr(\beta_u \geq \beta) \geq 1/2 \tag{9.10}$$

Hirji *et al.* (1989) demonstrated that the median unbiased estimator is generally more accurate than the maximum likelihood estimator for small sample sizes.

I used `PROC LOGISTIC` to do exact estimation for the data in Tables 9.1 and 9.2. For the completely separated data in Table 9.1, the *p*-value for the coefficient of

x was 0.0079. The median unbiased estimate was 0.7007. For the quasi-completely separated data in Table9.2, the *p*-value was 0.0043 with a median unbiased estimate 0.9878.

Despite the attractiveness of exact logistic regression, it's essential to emphasize that it is computationally feasible only for quite small samples. For example, I recently tried to estimate a model for 150 cases with a 2 to 1 split on the dependent variable and five independent variables. The standard version of `LogXact` was unable to handle the problem. An experimental version of `LogXact` using a Markov Chain Monte Carlo method took three months of computation to produce the *p*-value for just one of the five independent variables.

9.6.2.5 Bayesian Estimation In those situations where none of the preceding solutions is appropriate, a natural approach is to do Bayesian estimation with a prior distribution on the regression coefficients (Hsu and Leonard 1997, Kahn and Raftery 1996). This should be easily accomplished with widely-available software like `BUGS` , but I have yet to try this approach. However, my very limited experience with this approach suggests that the results obtained are extremely sensitive to the choice of prior distribution.

9.6.3 Solutions for complete separation

It is fortunate that complete separation is less common than quasi-complete separation because, when it occurs, it is considerably more difficult to deal with. For example, it's not feasible to leave the problem variable in the model because that makes it impossible to get maximum likelihood estimates for any other variables. And combining categories for dummy variables will not solve the problem either. Exact logistic regression may work for small sample problems, but even then complete separation can lead to breakdowns in both conditional maximum likelihood and median unbiased estimation. Bayesian estimation may be a feasible solution, but it does require an informative prior distribution on the problem parameters and the results may be sensitive to the choice of that distribution.

With conventional logistic regression, about the only practical approach to complete separation is to delete the problem variable from the model. That allows one to get estimates for the remaining variables but, as noted earlier, the exclusion of the problem variable could lead to biased estimates for the other variables. If one chooses to go this route, it is also essential to compute and report likelihood ratio chi-squares or exact *p*-values (where feasible) for the problem variable so that the statistical significance of this variable can be assessed.

9.6.4 Extensions

In this chapter, I have focused entirely on problems of non-convergence with binary logistic regression. But it's important to stress that complete and quasi-complete separation also lead to non-existence of maximum likelihood estimates under other

"link" functions for binary dependent variables, including the probit model and the complementary log-log model. For the most part, software treatment of data with separation is the same with these link functions as with the logit link. The possible solutions I described for the logistic model should also work for these alternative link functions, with one exception: the computation of exact p-values is only available for the logit link function.

Data separation can also occur for the unordered multinomial logit model; in fact, complete and quasi-complete separation were first defined in this more general setting (Albert and Anderson 1984). Separation problems can also occur for the ordered (cumulative) logit model although, to my knowledge, separation has not been rigorously defined for this model. Table 9.4 displays data with complete separation for a three-valued, ordered dependent variable

Table 9.4 Ordered Data Exhibiting Quasi-Complete Separation

x	1	2	3	4	5	6	7	8	9	10	11	12
y	1	1	1	1	2	2	2	2	3	3	3	3

These data could be modified to produce quasi-complete separation by adding a new observation with $x = 5$ and $y = 1$. `PROC LOGISTIC` in SAS correctly identifies both of these conditions and issues the same warning messages we saw earlier.

REFERENCES

Albert, A. , and J. A. Anderson. (1984). On the Existence of Maximum Likelihood Estimates in Logistic Regression Models. *Biometrika* **71**, 1-10.

Amemiya, Takeshi. (1985). *Advanced Econometrics*. Cambridge, MA: Harvard University Press.

Cox, D. R. (1970). The Continuity Correction. *Biometrika* **57**, 217-219.

Hirji, Karim F., Anastasios A. Tsiatis, and Cyrus R. Mehta. (1989). Median Unbiased Estimation for Binary Data. *The American Statistician* **43**, 7-11.

Hsu, John S. J., and Tom Leonard. (1997). Hierarchical Bayesian Semiparametric Procedures for Logistic Regression. *Biometrika* **84**, 85-93.

Kahn, Michael J., and Adrian E. Raftery. (1996). Discharge Rates of Medicare Stroke Patients to Skilled Nursing Facilities: Bayesian Logistic Regression with Unobserved Heterogeneity. *Journal of the American Statistical Association* **91**, 29-41.

Santner, Thomas J., and Diane E. Duffy. (1986). A Note on A. Albert and J. A. Anderson"s Conditions for the Existence of Maximum Likelihood Estimates in Logistic Regression Models. *Biometrika* **73**, 755-758.