Logistic Regression Models
Joseph M Hilbe

ERRATA and COMMENTS
4th Printing (Printed Sept, 2010)
(updated to: 27 October, 2011)

The 4th printing enhances Stata code to use version 11 rather than version 9-10 code. The book was completed before Stata version 11 was published. For example, when constructing synthetic data, the book now uses the new Stata pseudo-random number generators rather than the ones I created back in 1995 – the suite of *rnd** commands -- or Roberto Gutierrez's unpublished *genbinomial* command.

No more corrections to the text are planned for future printings. A second edition is planned to be published in 2013 and will include nested logistic regression, and chapters on latent class models and on Bayesian logistic models. Both single and multilevel models will be examined. Certain areas of the present edition will be re-written to assist in clarity Any suggestions you have, or typos/errors you discover in the present printing of the first edition will be most appreciated.

Instructors may request a gratis 187-page *Solutions Manual for Logistic Regression Models*, Chapman & Hall/CRC, ISBM: 978-1-4398-2066-7. Contact author for details (hilbe@asu.edu or jhilbe@aol.com). It is available from publisher, but I will need to give you added information.

NOTE: 4th Printing is found on page opposite the table of contents. The numbers on the line under "Printed in the United States of America..." end with the number 4 -- the last number is the printing.

Thanks to Zhehui Luo of the Michigan State Dept of Epidemiology and students in my courses on Logistic Regression and Advanced Logistic Regression for identifying remaining typos & errors. I have added comments and additions to the actual errata. The Comments section follows the Errata, beginning with page 3.

Page xvii: Final full paragraph at the bottom of the page. The books web site should now read:
http://works.bepress.com/joseph_hilbe/

Page 1(bottom) page 2 (top): Starting from the sentence beginning with "First, the error term..." on the bottom line of Page 1, amend to read:
"First, the error terms are non-normally distributed. Second, the..."

Page 18: The terms A*D/B*C near the bottom of the page: Change to small letters to read as:
"The odds ratio is calculated by (a*d)/(b*c) or (a/c)/(b/d)."

Page 19: Near top of page: add "nolog" to first Stata command line under "MAXIMUM LIKELIHOOD LOGISTIC COMMAND". Read as:
". logistic death anterior, nolog"

Page 30,31: The comments to the right of the calculations of probabilities for each of the three non-reference Killip level. Delete the ending phrase for each, "with respect to KK1".

Page 39: The top Stata output in mid page: the term "ons" should read "_cons"

Page 110 Eq 5.19 Close parentheses for both numerator an denominator.

Page 118: ">" sign between RR and left side equation should be "="

Pages 120 and 128: The "///" symbols should be "//".

Page 130 third word, "percent", of the first full paragraph is misspelled. The sentence should read as: "The 95 percent confidence interval of the attributable risk is given as"

Page 132 third/fourth line under equation 5.40. Change sentence beginning with "Scaling replaces" to read as:
"... Scaling replaces *W* by the product of the model standard error and square root of the Pearson dispersion statistic."
Thus, scaled SE = se(βs) = se(β)*sqrt(Pdispersion).

Page 133: Close space between *rbinomial* and *(d,exb)*. Read as `gen by = rbinomial(d, exb)`

Page 172: R code: 9th line from top of page. Should read as:

```
age2 <- ifelse(agegrp=='61-69', 1,0)
```

Page 215: Amend equations 6.11 and 6.12 so that there is a bracket on the 3rd term of each

$$\text{Variance} = (r_1-r_0)^2 * V(\beta_1) + [x(r_1-r_0)]^2 * V(\beta_3) + 2x(r_1-r_0)^2 * CV(\beta_1,\beta_3) \quad (6.11)$$

$$\text{SE} = \text{sqrt}[(r_1-r_0)^2 * V(\beta_1) + [x(r_1-r_0)]^2 * V(\beta_3) + 2x(r_1-r_0)^2 * CV(\beta_1,\beta_3)] \quad (6.12)$$

Page 217: The formula used to calculate a p-value near the bottom of the page is mistaken. See page 104 for explanation. The last Stata code and output on the page should read as:

```
. di (1-normprob(1.404184))*2
.16026407
```

The corresponding R code is (for pages 239/240)

```
> pnorm(1.40184, lower.tail=F)*2
```

Page 259. Add sentence to the end of Section 7.3, just above 7.3.1
"In general, BIC statistics give greater adjustment weight to the number of predictors in the model than does AIC. "

Page 263. Section 7.3.4: The name "Swartz" should read "Schwartz" and the statistic is better known as a BIC statistic. Move Section 7.3.4 to page 267 under section7.3.6, and re-title it

7.3.6 SCHARTZ BIC

Change Section 7.3.5 to 7.3.4 and Equation 7.22 relabeled as 7.21
and Equation 7.23 relabeled as 7.22
Change Section 7.3.6 to 7.3.5 and Equation 7.24 relabeled as 7.23

Change section to read ==>

==========

7.3.6: SWARTZ BIC

The Swartz BIC statistic, designed by Joel Swartz of Harvard University in 1978, is defined as:

$$BIC_S = (-2LL + k*\ln(n))/n \quad (7.24)$$

with the BIC statistic having the lowest absolute value as the preferred model.

```
. di (-638.15957 + 8*ln(4503))/4503  <= preferred
-.12677317

.di (-686.28751 + 6*ln(4503))/4503
-.14119754
```

==========

Page 272. Equation 7.31, Parentheses are needed for the denominator of the second term within brackets, y/(m*μ)). The equation should read as:

$$d = +/- \text{ sqrt}[2\Sigma y * \ln(y/(m*\mu)) - (m - y)*\ln((m - y)/(m*(1-\mu))]$$

Page 293. R code: 3rd line from top.
Use: library(PresenceAbsence)
in place of: library(epicalc)

Page 299; Equation 8.14: the final term should read $ln\binom{m}{y}$.

Page 300: Code in mid-page. Should read as:

$\chi^2 = \Sigma\ (y-\mu)^2\ /\ (\mu*(1-\mu/m))$

$LL = \Sigma\{\ y*\ln(\mu/m)+(m-y)*\ln(1-(\mu/m))\}$

Page 323: Delete "/// a user authorized command" near the top right of the page.

Page 335: Delete the "]" at the end of the long line of Stata code in middle of page.

Page 368: Box 10.1: the section on *white* should read as
white: The expected odds of being admitted to the hospital as an emergency patient is some 40 % less among those who identified themselves as white compared with those who identified themselves as non-white, holding the other predictors constant

Page 376: Line immediately above Section 10.4: change words "a higher level" to "*Emergency*".

Page 387: The first word, “The”, of the paragraph immediately under equation 11.9 is mistaken. The paragraph should start out as:
“It is important to remember that the above parameterization is based on set-“

Page 388: the table about ¼ a page from the top has the 0 and 1 values in the wrong places. It should instead read as:

		Response 0	1
Predictor	0	A	B
	1	C	D

If you find additional errata, please advise. I will post them to this Errata page in the future. Thank you to those who have identified typos. I will list your names in the second edition.

COMMENTS

Page 65: I probably should have added the formula for the second derivative of the Bernoulli link function under Equation 4.12.

$$g''(\mu) = (2\mu - 1)\left(\frac{\partial \eta}{\partial \mu}\right)^2 = \frac{(2\mu - 1)}{\mu^2(1-\mu)^2} \qquad (4.12a)$$

Page 67: Table 4.1 provides a schematic algorithm for the estimation of a binary logistic model. I have provided full working code for estimating a generic logistic regression using Stata and R. I display the code and output below for each below the final Comment in this section.

Page 132 Comment: In R, the *glm quasibinomial* family is the same as scaling the binomial logistic model standard errors by the Pearson dispersion statistic. I recently discovered that the R *vcov()* function that is used by programmers for calculating standard errors in fact creates scaled standard errors. This results in the SEs of models using *sqrt(diag(vcov(modelname)))* for calculating SEs to have different SEs from Stata, SAS, and other applications, particularly when the data is correlated. Dividing the displayed SEs by the square root of the Pearson dispersion statistic produces model SEs. R's *glm()* function, which is used for estimating both binary and grouped logistic models, adjusts SEs so that model SEs are displayed in the results.

Page 300 Suggestion: the deviance function as presented is the standard one shown in texts. However, it does not work properly if used in an R GLM program. A much more simple and suitable expression for the equation, requiring less memory, is the following:

```
Dev = 2Σ{y*ln(1/μ) + (m-y)*ln(1/(m-μ))}
```

Joseph M Hilbe
hilbe@asu.edu or jhilbe@aol.com

STATA USER AUTHORED LOGIT COMMAND. First published in the November 2005 issue of *The American Statistician* in a review of Stata. The review may be obtained from my BePress Selected Works site, http://works.bepress.com/joseph_hilbe/

==========

```
*! version 1: LOGISTIC REGRESSION   :IRLS METHOD OF  ESTIMATION
* Joseph Hilbe: TAS - Stata 9.0 review: 7Jul2005
program define  jhlogit
version 6
set type double
syntax  varlist(default=none) [if] [, EForm]
gettoken y varlist : varlist
if `"`if"' != `"" {
       preserve              /* ensure the dataset returns at end of pgm */
       keep `if'             /* retain only estimation sample */
}
if "`eform"' !=  "" { local eform "eform(Odds Ratio)" }
qui  {
tempvar mu eta u w z dev oldev llike chi2 aic  bic
* INITIALIZATION OF MU AND ETA
count
local nobs =  _result(1)
gen `mu' = (`y' +  0.5)/2
gen `eta' = ln(`mu'/(1-`mu'))
* VARIABLE  INITIALIZATION
local i      1
gen `u'     =0
gen `w'     =0
gen `z'     =0
gen `dev'  =1
gen `oldev'=1
gen `chi2'  =1
local ddev  1
* IRLS SCORING
while (abs(`ddev')> 1e-6 ) {
replace  `u' = (`y'-`mu')/(`mu'*(1-`mu'))
replace  `w' = `mu'*(1-`mu')
replace `z' = `eta'  + `u'
regress `z' `varlist' [iw=`w'], mse1  dep(`y')
drop  `eta'
predict  `eta'
replace `mu'   =  1/(1+exp(-`eta'))
replace `oldev'=  `dev'
replace `dev'  = ln(1/`mu')  if `y'==1
replace `dev' =   ln(1/(1-`mu')) if `y'==0
replace  `dev'  = sum(`dev')
replace  `dev'  = 2*`dev'[_N]
local  ddev     = `dev' -  `oldev'
local  i        = `i'+1
}
local npred =  _result(3)              /* number of predictors */
local df    = `nobs' - `npred' -  1    /* degrees of freedom   */
* CALCULATION OF  LOG-LIKELIHOOD AND GOF STATISTICS
egen `llike' =  sum(`y'*ln(`mu')+(1-`y')*ln(1-`mu'))
gen `aic' = (-2*`llike' +  2*`npred')/`nobs'          //    AIC/observations
}
```

```
* PUT VALUES INTO MATRIX
qui regress, noheader `eform'
tempname b  V
mat `b' =  get(_b)          /*  coefficient vector */
mat `V' =  get(VCE)         /*  variance-covariance matrix */
mat post `b' `V', depname(`y')  obs(`nobs')
* OUTPUT
di " "
di in gr "Logistic  Estimates"
mat mlout, `eform'
di in gr _col(1) "Observations = " in  ye `nobs' in gr _col(53) "Deviance     =   " in ye  `dev'
di in gr _col(53) "Loglikelihood = "  in ye `llike'
di in gr  _col(1)  "AIC Statistic =  " in ye `aic'
set type double
end
=====================================================================
```

USE OF COMMAND

```
. use medpar           /* dataset explained in text, Ch 5.11; p. 159 */

. jhlogit died hmo white

Logistic  Estimates
------------------------------------------------------------------------------
        died |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         hmo |  -.0122465   .1489251    -0.08   0.934    -.3041342    .2796413
       white |   .3033872   .2051795     1.48   0.139    -.0987573    .7055318
       _cons |  -.9261862   .1973903    -4.69   0.000    -1.313064   -.5393082
------------------------------------------------------------------------------
Observations = 1495                               Deviance     =   1920.602
                                                  Loglikelihood = -960.301
AIC Statistic =  1.2873592
```

R -- Bernoulli or binary logistic regression -

```
=============================================================================
# BINARY LOGISTIC REGRESSION. BASIC FUNCTION   7 July, 2011
# From: Hilbe, J.M and A.P Robinson (2012), Methods of Statistical Model
#    Estimation, Chapman & Hall/CRC
irls_logit <- function(formula, data, tol=.000001) {  # irls_logit options
  mf <- model.frame(formula, data)              # define model frame as mf
  y <- model.response(mf, "numeric")            # set model response as y
  X <- model.matrix(formula, data = data)       # predictors in matrix X
  if (any(is.na(cbind(y, X)))) stop("Some data are missing.")
  mu <- (y + .5)/2                              # initialize μ
  eta <- log(mu/(1-mu))                         # initialize η
  dev <- 2 * sum( y*log(1/mu) + (1 - y)* log(1/(1-mu)) )
  deltad <- 1                                   # initialize deltad = 1
  i <- 1                                          # initialize i=1
  while (abs(deltad) > tol ) {                    # IRLS loop begin
    w <-  mu*(1-mu)                               # weight
    z <- eta + (y - mu)/w                         # working response
    mod <- lm(z ~ X-1, weights=w)                 # weighted regression
    eta <- mod$fit                                # linear predictor
```

```
    mu <- 1/(1+exp(-eta))                          # fitted value; probability
    dev.old <- dev                                 # setup for convergence
    dev <- 2 * sum( y*log(1/mu) + (1 - y)* log(1/(1-mu)) ) # deviance
    deltad <- dev - dev.old                        # test of 2 iterations
    cat(i, coef(mod), deltad, "\n")                # iteration log
    i <- i + 1                                     # recalibrate iter number
  }
    beta <- mod$coef                                  # save coefficients
    pr <- sum(residuals(mod, type="pearson")^2)  # calc Pearson disp
    prdisp <- pr/mod$df.residual
    return(list(coef = coef(mod),                  # coef & SE display
            se = sqrt(diag(vcov(mod)))/ sqrt(prdisp)))
}
================================================================================
```

USE -- how *source()* is defined is based on where *irls_logit.r* is stored on your computer. It will be a function in the *msme* library later in 2011 (download fom CRAN).

Coefficients and model standard errors are displayed. Confidence Intervals, Z statistic, and p-values can be easily calculated. Note that the scaled SEs calculated by vcov() are amended to produce true model SEs.

NOTE: A complete description of OLS, IRLS, maximum likelihood, EM, quadrature, simulation, and other major methods of estimation can be found in **Hilbe, Joseph M. and Andrew P. Robinson (2012),** ***Methods of Statistical Model Estimation*****, Chapman & Hall/CRC**. The *irls_logit* function is fully described as an example of IRLS estimation. Other more complex IRLS models are also discussed. In addition, we created a *glm*-like function called *irls*, which corrects what we believe to be shortcomings in *glm()* and *glm.nb()*, describing its modular logic the specifics of the code. After the *msme* library is loaded, *irls()* will be able to be used like *glm()* is now, together with a *summary()* function. *irls()*, however, provides a much more extensive list of post-estimation statistics.

```
> library(COUNT)  # Package associated with my Negative Binomial Regression
> source("c://rfiles/irls_logit.r")    # locate where function is saved
> data(medpar)

> i.logit <- irls_logit(died ~ hmo + white, data=medpar)
1 -1.051936 -0.01343265 0.318181 1064.628                  # iteration log
2 -0.9224268 -0.01216259 0.3017145 -4.193304
3 -0.9261831 -0.01224641 0.3033848 -0.001737683
4 -0.9261862 -0.01224648 0.3033872 -3.808509e-10

COEFFICIENTS and STANDARD ERRORS
> i.logit
X(Intercept)         Xhmo       Xwhite
 -0.92618620  -0.01224648   0.30338724

$se
X(Intercept)         Xhmo       Xwhite
   0.1973903    0.1489251    0.2051795

LOWER 95% CONFIDENCE INTERVAL
> i.logit$coef - 1.96*i.logit$se
```

```
X(Intercept)          Xhmo       Xwhite
 -1.31346050  -0.30443326  -0.09916926
```

```
UPPER 95% CONFIDENCE INTERVAL
> i.logit$coef + 1.96*i.logit$se
X(Intercept)          Xhmo       Xwhite
  -0.5389119    0.2799403    0.7059437
```

The Z-statistic and P-values may be easily calculated from the above, but the confidence intervals will indicate if a predictor is significant as well. When odds ratios are displayed, exp(i.logit$coef), recall that the standard errors are determined using the delta method, which in this case is quite simple: exp(β)*se(β); ie.

ORse <- exp(i.logit$coef)* i.logit$se.

See page 35 in text.

Also of possible interest to readers, James Hardin and I are currently preparing the third edition of *Generalized Linear Models and Extensions* (Stata Press) [GLME3] and then the second edition of *Generalized Estimating Equations* (Chapman & Hall/CRC) [GEE2]. GLME3 is due to the publishers October 15th. We do not have a due date for GEE2, but intend to complete it before the end of the year. I am also working on a book with Justine Shults on a book called *Quasi-Least Squares Regression: Extending GEE methodology* (Chapman & Hall/CRC). It was begun in 2007 and should be completed in early 2012. I discuss QLS and mention the book in LRM (pages 470-480).