Beta Regression: Practical Issues in Estimation

Michael Smithson and Jay Verkuilen,

The Australian National University and University of Illinois at Urbana-Champaign

General Procedures

This document supplements the paper by Smithson and Verkuilen (2005) on beta regression, and focuses on maximum likelihood estimation procedures in several statistical packages. Maximizing the likelihood function can be achieved using a Newton-Raphson or a quasi-Newton method. Ferrari and Cribari-Neto (2004) use Fisher scoring. Buckley (2002) has used MCMC estimation in winBUGS, which provides a Bayesian posterior density. Buckley (2002) also provided Stata code and Paolino (2001) provided Gauss code, both of which compute maximum likelihood estimates. We have estimated beta regressions using R, SPlus, SAS, SPSS, Mathematica, and winBUGs. Syntax and/or script files for all of these packages are freely available on this site.

Different packages require more or less information from the user. SAS, for instance, only requires the likelihood and analytically computes the derivatives for Newton methods, whereas Mathematica's Newton-Raphson routine requires the user to supply expressions for the derivatives. The major difference between Newton-Raphson and quasi-Newton is in the number of function evaluations per iteration (more for Newton-Raphson) and the number of iterations necessary (more for quasi-Newton). The domain of convergence for different algorithms, and hence importance of good starting values, will differ across algorithms as well. The trust region Newton method implemented in SAS seems to be particularly stable on hard problems and we recommend its use when convergence might be a problem. In general, speed of execution for ML estimation is proportional to $N + p^2$, where N is the sample size and p is the number of parameters. We have never observed a well-specified model given

good starting values taking longer than a few seconds to converge, even on a fairly modest laptop.

If a Newton or quasi-Newton method is used, asymptotic standard errors usually are estimated from the inverse of the final Hessian matrix. Bayesian estimation gives posterior densities from which the Bayesian analogs of frequentist stability measures can be taken, e.g., the 2.5% and 97.5% quantiles of the posterior density as analogous quantities to a 95% confidence interval. Though it is generally recommended in the literature that the Newton estimate of the Hessian be used to provide asymptotic standard errors, we have tried both methods on several data sets and it has never seemed to make an appreciable difference.

Well-chosen starting values are needed to ensure convergence when more than a few variables are included in the model. We have found two effective approaches to generating starting-values. Ferrari and Cribari-Neto (2004) suggest using the OLS estimators from the regression on the link-transformed dependent variable for the location sub-model. For example, if the location sub-model link is the logit then starting-values for the $\boldsymbol{\beta}$ would be obtained via

$\ln(y_i/(1 - y_i)) = \mathbf{X}\boldsymbol{\beta}_{\text{OLS}} + \varepsilon_i$.

In general the OLS estimates track the location model tolerably well, though of course the standard errors will be inefficient and, more importantly, the variance structure will be limited to an intercept parameter. Unfortunately, to our knowledge there seems to be no similar proposal for starting-values of the coefficients in the dispersion sub-model.

The second approach is to begin with the null model (i.e., intercept-only sub-models with coefficients β_0 and δ_0), using the method of moments to provide starting-values for β_0 and δ_0. Then the resultant maximum-likelihood estimates of β_0 and δ_0 are used along with starting-values close to 0 for of β_1 and/or δ_1 ($\beta_1 = \delta_1 = 0.1$, say), and maximum-likelihood estimates for β_0, δ_0, β_1 and/or δ_1 are obtained. These in turn are used as starting-values for the next

more complex model, with starting-values near 0 used for each new term being introduced into the model. Models can be built up one or more terms at a time in this way, although adding more than one term at a time is riskier.

Another estimation issue is that covariates with particularly large absolute values may result in a loss of precision in estimates or create problems for the estimation algorithms. This is due to the evaluation of exponents, and we have found that absolute values greater than around 30 can cause difficulties. These covariates may need to be rescaled to smaller ranges. The loss of numerical precision due to variables with widely different scales, e.g., one variable in millions and the other in thousandths, is common to all estimation procedures and is not unique to beta regression. Rescaling in this case is required.

A third practical issue is the treatment of endpoints. Although 0 and 1 may be genuine outcome values, their logits are undefined. Two obvious remedies are proportionally "shrinking" the range to a sub-range nearly covering the unit interval (e.g., [.01, .99]) or simply adding a small amount to 0-valued observations and subtracting the same amount from 1-valued observations while leaving the other observations unchanged. Both methods bias the estimates toward no effect. A method that is frequently used in practice in areas such as signal detection theory is to add $1/2N$ to a 0 observation and subtract $1/2N$ from a 1 observation, where N is the total number of observations (MacMillan and Creelman 2005, pp. 8-9). This has no effect on the interior points but could introduce bias if there are a non-trivial number of boundary values. The best option may be to experiment with different endpoint handling schemes and see whether the parameter estimates change in any appreciable way.

Our recommendation applies to a sample of scores from a continuous bounded scale that has already been linearly transformed to the [0,1] interval. We may transform the sample scores to a variable in the open (0,1) interval by the weighted average:

$$y'' = [y(N - 1) + s]/N,$$

where N is the sample size and s is a constant between 0 and 1. From a Bayesian standpoint, s acts as if we are taking a prior into account. A reasonable choice for s would be .5.

Finally, researches will need to consider whether beta regressions can be applied to a discrete dependent variable, even an interval-level one. It is common practice to apply normal-theory regression to variables that have only a few scale values, e.g., survey items with a response range from 1 to 7, and researchers are likely to be motivated to do the same with beta regression. Tamhane, Ankenman and Yang (2002) discuss the use of a beta distributed underlying variable for ordinal data. They indicate that using the raw discrete scores works reasonably well, though they provide a "continuizing" procedure that they show improves the mean square error somewhat. Their procedure simply amounts to uniformly and randomly assigning values to identical observations within the range of their "bin."

For the present, researchers will have to rely on practical experience in the absence of theoretical results regarding this issue. A simulation study showing how beta regression degrades in the presence of discretization would be desirable—if in doubt compare an ordinal regression and beta regression. In the meantime, we recommend a linear transformation of a discrete equal-interval scale that assigns values to bin midpoints. A variable Y with n+1 scale values $\{a, a+b, a+2b, \ldots, a+nb\}$ is transformed into

$y' = (2y - a)/2(nb + a)$.

For example, the seven scale values {1, 2, 3, 4, 5, 6, 7} are transformed into {1/14, 3/14, …, 13/14} which are the midpoints of the seven bins equally dividing the [0,1] interval.

Implementation in statistical packages

In this section we briefly describe the implementations in R/SPlus, SPSS, and SAS. All three implementations require the user to input model terms and starting-values, but otherwise make no special demands. Whenever possible, we recommend fitting beta regression models in more than one package and using more than one set of starting-values

and optimization method to check convergence. We have done this for many models and beta regression seems to converge readily to the optimum, assuming starting values are good. Models with several terms in the dispersion sub-model, however, are more difficult to estimate.

The current implementation in R and SPlus provides the maximum likelihood value attained by the model, coefficients, gradient for the coefficients, asymptotic standard-error estimates and p-values. Predicted values and residuals also can be output or saved in the data-file. It uses two functions ("betareg" and "grad") and a small set of output commands, all of which can be used in the Commands window or saved as scripts. The SPlus version uses the nlminb quasi-Newton routine for MLE and the Venables and Ripley (1999) vcov.nlminb in the MASS library for computing the Hessian and the asymptotic estimates. The R version uses the optim routine for MLE with the BFGS quasi-Newton method. A version of optim for SPlus also is available in the MASS library.

In SPSS, beta regression models can be fitted under its Nonlinear Regression (NLR and CNLR) procedure. We have written a syntax shell and documentation so that users need only supply a model and starting-values. The CNLR procedure outputs an iteration history and a convergence message. SPSS uses numerical approximations to the derivatives, so the analytic score function and Hessian are not needed. There are subcommands that provide predicted values, residuals, gradient values, and bootstrap standard-error estimates for the coefficients. However, the Hessian and asymptotic standard error estimates are not currently available.

In SAS, `proc nlmixed` and `proc glimmix` will estimate a beta regression. The `glimmix` program does not allow a heteroscedasticity model but does generate reasonable estimates for variance component models with beta responses, whereas the examples we have tried using `nlmixed` did not converge. Anyone who has a basic familiarity with SAS can code up their own model easily using our heavily commented syntax file as an example.

There are many choices for the numerical optimizer to be used. Even for Newton methods, the analytic score function and Hessian are not needed as SAS will use automatic differentiation to obtain the derivatives. The quasi-Newton method BFGS seems to be a solid all-around choice in practice, with the trust region Newton method being useful for more difficult problems. The output of `nlmixed` is extensive. Detailed iteration information is available for each estimated parameter, along with the likelihood, *AIC*, *BIC*, and asymptotic confidence intervals. In addition to the basic output, it is possible to use additional command statements to generate predicted values, residuals, and to save output in RTF and LaTex format. To obtain bootstrap or jackknife statistics, SAS Institute has the freely downloadable `jackboot` macro on their website.

References

Buckley, J. (2002). Estimation of models with beta-distributed dependent variables: A replication and extension of Paolino (2001). *Political Analysis, 11*, 1-12.

Cribari-Neto, F. and Vasconcellos, K. L. P. (2002). Nearly unbiased maximum likelihood estimation for the beta distribution. *Journal of Statistical Computation and Simulation, 72*, 107-118.

Ferrari, S. L. P. and Cribari-Neto, F. (2004). Beta Regression for Modeling Rates and Proportions. *Journal of Applied Statistics, 10*, 1-18.

McMillan, N. A. and Creelman, C. D. (2005). *Detection Theory: A User's Guide*, Mahwah, NJ: Lawrence Erlebaum Associates.

Paolino, P. (2001). Maximum likelihood estimation of models with beta-distributed dependent variables. *Political Analysis 9*, 325-346.

SAS Institute (2005). *The GLIMMIX Procedure*. Cary, NC: SAS Institute.

Smithson, M. and Verkuilen, J. (2005). A Better Lemon-Squeezer? Maximum Likelihood Regression with Beta-Distributed Dependent Variables.

Tamhane, A. C., Ankenman, B.E. and Yang, Y. (2002). *The Beta Distribution as a Latent Response Model for Ordinal Data (I): Estimation of Location and Dispersion Parameters*. Unpublished manuscript, Department of Industrial Engineering and Management Sciences, Northwestern University, Evanston, IL.

Venables, W. N. and Ripley, B. D. (1999). *Modern applied statistics with S-PLUS*. New York: Springer-Verlag.