The Stata Journal

The *Stata Journal*, electronic version (ISSN 1536-8734) is a publication of Stata Press, and Stata is a registered trademark of StataCorp LP.

The Stata Journal (2005)
5, Number 4, pp. 501–516

Extended generalized linear models: Simultaneous estimation of flexible link and variance functions

Anirban Basu
Section of General Internal Medicine, University of Chicago
and
Decision and Information Sciences Division, Argonne National Laboratory
5841 S Maryland Ave, MC-2007, Chicago IL 60637
abasu@medicine.bsd.uchicago.edu

Abstract. I describe a command that simultaneously solves the extended estimating equations estimator for parameters in the link and variance functions along with those of the linear predictor in a generalized linear model. The method addresses difficulties in choosing the correct link and variance functions in these models. It decouples the scale of estimation for the mean model, determined by the link function, from the scale of interest for the scientifically relevant effects. It also estimates a flexible variance structure from the data, leading to efficient estimation.

Keywords: st0092, pglm, pglmpredict, EEE, GLM, skewed, costs, estimating equations, link functions, variance functions

1 Introduction

Many outcome variables in health economics and biostatistics are characterized by nonnegative values, heteroskedasticity, heavy skewness in the right tail, and kurtosis distributions. Researchers have demonstrated the usefulness of several advanced modeling strategies to address these aspects of the underlying distributions (Blough, Madden, and Hornbrook 1999; Manning and Mullahy 2001). However, in most econometric applications, critical assumptions are made regarding the functional forms of the heteroskedastic variance and mean function that incorporate the effect of a covariate on the outcome. For example, in ordinary least squares (OLS) regression, the effect of a covariate is additive; in log-transformed ordinary least squares (log-OLS) regression and in generalized linear models (GLMs) with a log link, the effect is multiplicative. Similarly, in many GLMs, the variance function is specified *a priori* that corresponds to a particular form of heteroskedasticity. For example, a gamma variance implies that the variance is proportional to the square of the mean function. However, incorrect specifications of the mean function and the variance function can lead to bias and inefficiency in estimation. These biases and inefficiencies can be identified using conventional goodness-of-fit tests. Unfortunately, although these diagnostic tests may detect problems, they provide no guidance on how to fix those problems.

This lack of guidance motivated me to develop a new estimator that relaxes the limitation of prespecifying a scale of estimation and the functional form of heteroskedasticity (Basu and Rathouz 2005). Rather, it allows estimation of a flexible mean function using the data at hand, thereby reducing bias in estimating the conditional mean outcome that may have arisen because of the misspecification of the scale of estimation. It also estimates a flexible variance structure from the data, leading to efficient estimation and allowing for different heteroskedastic specifications. I have shown elsewhere that by allowing flexible mean and variance functions, many of the problems of misspecification can be overcome considerably (Basu and Rathouz 2005). Here I describe the Stata command `pglm`, which I have written to implement this estimator.

Now I briefly explain the differences between the `pglm` command and the commands and materials currently available in Stata that allow for GLM estimation and for extending the GLM estimator to include nonstandard link functions. The `glm` command in Stata fits GLMs, using either iteratively reweighted least squares (IRLS; maximum quasilikelihood) or Newton–Raphson (maximum likelihood) optimization. The additional programs provided by Guan and Gutierrez (2002) extend the `glm` command to include customized link functions. However, whether using the `glm` command or the extensible programs provided by Guan and Gutierrez (2002), the analyst always needs to specify the particular link parameter (e.g., log, identity, or any other power function) and the variance function corresponding to a distribution that he wants to fit. It is in this aspect that `pglm` differs from the remaining commands in Stata. The `pglm` command does not require the analyst to know what the appropriate link or the variance function should be for a given dataset. Instead, it uses additional estimating equations, which are not part of the current `glm` module, to estimate these ancillary parameters. Also the `pglm` command fits a semiparametric model that does not use full distributional assumptions or full likelihood estimation methods but instead uses IRLS (maximum quasilikelihood) optimization.

Hardin and Hilbe (2001) explained how the `glm` command can also be used to search for the optimal link and the variance functions. Based on the profile-extended quasilikelihood approach (Nelder and Pregibon 1987), the `glm` command has to be run for many combinations of values for the link and variance parameters. The combination of values that produces minimal deviance may be used as the optimal link and variance functions. Such a method was also proposed by Blough, Madden, and Hornbrook (1999). Unfortunately, this procedure is extremely time-consuming and does not take into account the uncertainty in estimating the link and variance parameters from the data. In contrast, the `pglm` command is one regression step that simultaneously estimates the link and variance parameters from the data along with the regression coefficients.

2 Background

2.1 Model descriptions

Let Y be the nonnegative outcome variable and $X = (X_0, X_1, X_2, \ldots, X_p)^T$ be the vector of covariates used in a regression model, where X_0 is a vector of ones. Interest is on modeling the mean function $\mu(x) \equiv E(Y_i|X_i = x)$ and functionals thereof. For example, when X_j is binary, the parameter of interest is the *incremental effect* given by $\pi_j \equiv E_{X_{-j}}\{D_j(\mu; X_{-j})\}$, where $D_j(\mu; x_{-j}) \equiv \mu(x_j = 1, x_{-j}) - \mu(x_j = 0, x_{-j})$, x_{-j} is the vector x without x_j and the expected value is over X_{-j}, marginally with respect to X_j. The parameter π_j is the population average contrast in the mean of Y for $X_j = 1$ and $X_j = 0$. The parameter π_j may be interpreted as the effect of X_j on the mean of Y, adjusting for all other covariates in the model, where this adjustment is to the population distribution of X. An analogous effect for a continuous covariate, termed the *marginal effect* of the covariate, may also be of interest where $D_j(\mu; x) \equiv \partial\mu(x)/\partial x_j$ (Basu and Rathouz 2005).

Letting $\mu_i = \mu(X_i)$, a GLM (McCullagh and Nelder 1989) can be framed, wherein $g(\mu_i) = \eta_i$, $\eta_i = X_i^T\beta$, and β is a $p \times 1$ vector of regression parameters. Here $g(.)$ is a strictly monotone differentiable link function that relates μ_i to the linear predictor η_i. Also the variance of the outcome variable is given by, $V_i = \mathrm{Var}(Y_i|X_i)$. Define a parametric family of link functions indexed by λ:

$$\eta_i = g(\mu_i; \lambda) = \begin{cases} (\mu_i^\lambda - 1)/\lambda, & \text{if } \lambda \neq 0 \\ \log(\mu_i), & \text{if } \lambda = 0 \end{cases}$$

(McCullagh and Nelder 1989, chapter 2; Box and Cox 1964; Hardin and Hilbe 2001, 81–83; Wooldridge 1992). Similar to the link function, define a family $h(\mu_i; \theta_1, \theta_2)$ of variance functions indexed by (θ_1, θ_2). Two such families are considered. The first, referred to as the power variance (PV) family (Hardin and Hilbe 2001, 81–83), sets $h(\mu_i; \theta_1, \theta_2) = \theta_1\mu_i^{\theta_2}$. It includes as special cases the variances of several standard distributions used for modeling health outcomes (table 1).

(*Continued on next page*)

Table 1: Special cases of distributions under PV and QV variance formulations

Variance formulation				**Distribution**
PV		**QV**		
$\boldsymbol{\theta}_1$	$\boldsymbol{\theta}_2$	$\boldsymbol{\theta}_1$	$\boldsymbol{\theta}_2$	
1	1	1	0	Poisson
> 0	2	0	> 0	Gamma
> 0	3	–	–	Inverse Gaussian
–	–	1	> 0	Negative binomial

PV = power variance, where $V(y_i) = \theta_1 \mu_i^{\theta_2}$;
QV = quadratic variance, where $V(y_i) = \theta_1 \mu_i + \theta_2 \mu_i^2$.
– = True value unknown since distribution does not conform to the particular variance structure assumed.

An alternative is the quadratic variance (QV) family given by $h(\mu_i; \theta_1, \theta_2) = \theta_1 \mu_i + \theta_2 \mu_i^2$; standard distributions corresponding to this form of variance are also listed in table 1.

2.2 Estimation

The regression and link parameters are estimated via an extension of quasilikelihood (Wedderburn 1974), and the variance parameters are estimated using additional estimating equations. We refer to this method as the extended estimating equations (EEE) estimator.

For the ith individual, the extended set of estimating functions for parameter vector $\gamma = (\beta^T, \lambda, \theta_1, \theta_2)^T$ is given as (Basu and Rathouz 2005):

$$
\begin{aligned}
G^i_{\beta_j} &= (Y_i - \mu_i) V_i^{-1} (\partial \mu_i / \partial \beta_j) \qquad\qquad j = 1, \ldots, p \\
G^i_{\lambda} &= (Y_i - \mu_i) V_i^{-1} (\partial \mu_i / \partial \lambda) \\
G^i_{\theta_1} &= \left\{ (Y_i - \mu_i)^2 - V_i \right\} V_i^{-2} (\partial V_i / \partial \theta_1) \\
G^i_{\theta_2} &= \left\{ (Y_i - \mu_i)^2 - V_i \right\} V_i^{-2} (\partial V_i / \partial \theta_2)
\end{aligned}
$$

Defining $G^i_\gamma = (G^i_{\beta_0}, G^i_{\beta_1}, G^i_{\beta_2}, \ldots, G^i_{\beta_p}, G^i_\lambda, G^i_{\theta_1}, G^i_{\theta_2})^T$ and the extended estimating function for γ as $G_\gamma = \Sigma_{i=1}^N G^i_\gamma$, the parameter vector γ is estimated by solving $G_\gamma = 0$, yielding estimator $\widehat{\gamma}$. The predicted mean in this model is obtained by $\widehat{\mu}(x) = (x^T \widehat{\beta} \cdot \widehat{\lambda} + 1)^{1/\widehat{\lambda}}$ $\forall \widehat{\lambda}$, $\widehat{\lambda} \neq 0$. For $\widehat{\lambda} = 0$, which can almost never happen, $\widehat{\mu}(x)$ is undefined.

To estimate the incremental effect π_j of an indicator variable, X_j, one can use the method of *recycled predictions* (StataCorp 2005, 2: 406; Basu and Rathouz 2005). This method, in obvious analogy to the definition of incremental effect π_j for covariate X_j, estimates π_j as

$$\widehat{\pi}_j = \widehat{E}_{X_{-j}}\left\{D_j(\widehat{\mu}; X_{-j})\right\} = N^{-1}\Sigma_i^N = 1\left\{\widehat{\mu}(X_{i,-j}, X_{ij} = 1) - \widehat{\mu}_i(X_{i,-j}, X_{ij} = 0)\right\}$$

The variance of $\widehat{\pi}_j$ is obtained using Taylor-series approximations and is given by

$$\mathrm{Var}(\widehat{\pi}_j) = \mathrm{Var}\left[\widehat{E}_{X_{-j}}\{D_j(\mu; X_{-j})\}\right] + \left(\frac{\partial \pi_j}{\partial \gamma}\bigg|_{\gamma}\right) A_N \left(\frac{\partial \pi_j}{\partial \gamma}\bigg|_{\gamma}\right)^T \tag{1}$$

In (1), the first term is the sample variance of $\widehat{\pi}_j$ due to using the empirical expected value over $X_{-j}, \widehat{E}_{X_{-j}}\{D_j(\mu; X_{-j})\}$, rather than the population expected value. The second term is due to the fact that γ is estimated, where A_N is the analytical variance–covariance matrix for $\widehat{\gamma}$. An estimator for the variance of the marginal effect analogous to (1) may be obtained through a similar approach. More details on the estimation process and variance calculations can be found in Basu and Rathouz (2005).

The initial values of the regression coefficients come from the estimates of regression coefficients from a gamma GLM with log link. The initial value of the link parameter λ is set to 0.1. For the PV structure, the initial value of θ_1 comes from the shape parameter computed by the gamma GLM. The initial value of θ_2 comes from the modified Park test (Manning and Mullahy 2001). In this test, the logarithm of the squared residuals from the log-link GLM is regressed on the logarithm of the predicated values ($\widehat{\mu}$) from the GLM. The coefficient of the $\log(\widehat{\mu})$ gives the initial estimate for θ_2. For the QV structure, the squared residuals from the log-link GLM is regressed on the predicated values ($\widehat{\mu}$) and the squared predicted values ($\widehat{\mu}^2$) without an intercept. The coefficient of $\widehat{\mu}$ gives an estimate for θ_1 and the coefficient of $\widehat{\mu}^2$ gives an estimate for θ_2.

Parameter estimates are updated using the equality $\widehat{\gamma}^{(k+1)} = \widehat{\gamma}^{(k)} + \mathbf{I}^{(k)-1}G_{\gamma}^{(k)}$, where $\mathbf{I}^{(k)} = E(-\partial G_{\gamma}^{(k)}/\partial\gamma)$. $\mathbf{I}^{(k)}$ and $G_{\gamma}^{(k)}$ are computed using the current value of $\widehat{\gamma}^{(k)}$. This procedure is iterated until the maximum relative difference in parameter estimates between two successive iterations $\gamma^{(k)}$ and $\gamma^{(k+1)}$ is less than 0.0001 (the maximum relative difference criterion may be changed by the user).

3 The pglm and pglmpredict commands

`pglm` (power-GLM) implements the EEE estimator (Basu and Rathouz 2005) and estimates the regression coefficients and the parameters in a flexible Box–Cox link function and a flexible variance function simultaneously. `matvsort` and `matewmf` (Cox 2000; a newer version of `matvsort` is available from the SSC, *http://fmwww.bc.edu/RePEc/bocode/m*) must be installed for this command to work. Also the algorithm shows better convergence properties if the outcome variable is scaled by dividing by its mean.

`pglmpredict` is used to obtain predictions following the `pglm` estimation command and optionally to calculate the variance, standard errors, Wald test statistics, significance levels, and pointwise confidence intervals for these predictions.

3.1 Syntax

`pglm` *depvar* [*indepvars*] [*if*] [*in*] [*weight*] [`, family(`*famname*`)`
`initlambda(#) power(#) vf(q) convergence(#) cluster(`*varname*`)`
`level(#) iteration(#)`]

`pgmlpredict` [*type*] *newvar* [*if*] [*in*] [`,` *predtype* `scale(#) se(`*newvar*`)`
`variance(`*newvar*`) wald(`*newvar*`) p(`*newvar*`) ci(`*newvar_ll newvar_ul*`)`
`level(#) g(`*stub*`) iterate(#) force`]

3.2 Description

`pglm` expects the data to be in the conventional form, as in any other regression analysis. It requires specification of a dependent variable and at least one covariate; i.e., it does not fit a constant-only model. The default command runs the model with both the flexible link and variance function. It uses the PV function as default (see below). `pglm` always reports robust variances since it uses IRLS (maximum quasilikelihood) optimization; therefore, robust variances (Huber 1967; White 1980) are always necessary for obtaining consistent estimates of the variance function. One can use any of the weight options in Stata with `pglm`; however, proper treatment of survey-design effects for standard errors (including, for example, effects of stratification) requires an `svy` estimator, which `pglm` is not.

Unlike `testnl` and `nlcom`, the quantities generated by `pglmpredict` are designed to vary over the observations in the data. The standard errors and other inference-related quantities are based on the "partial method", an approximation appropriate in large samples. `pglmpredict` calls on Stata's `predictnl` command to calculate predictions. The predictions computed by the `pglmpredict` command directly apply the formulas given in Basu and Rathouz (2005). Calculations of standard errors for the incremental and marginal effects are illustrated in the empirical example below.

3.3 Options

Options for pglm

`family(`*famname*`)` specifies the variance function (V) of a particular distribution. When `family()` is specified, one of the variance function parameters becomes fixed and is not estimated. *famname* is one of the following:

`family(power)`: $V = \theta_1\mu^{\#}$, where # is any real value specified using the `power(#)` option and $\mu = E(Y|X)$.

`family(gaussian)`: $V = \theta_1$, corresponding to the Gaussian family.

`family(poisson)`: $V = \theta_1\mu$, corresponding to the Poisson family.

`family(gamma)`: $V = \theta_1\mu^2$, corresponding to the gamma family.

`family(igaussian)`: $V = \theta_1\mu^3$, corresponding to the inverse Gaussian family.

When `vf()` is specified, only `family(poisson)` is available.

`initlambda(#)` specifies the starting value for the link parameter. If no value is specified, log link is used as the starting point.

`power(#)` specifies a value for the power parameter in the variance function. If no value is specified, this parameter is estimated from the data. This option should be used only when `family(power)` is specified and `vf(q)` is not specified.

`vf(q)` requests a QV function for the model, where $V = \theta_1\mu + \theta_2\mu^2$. If this option is not specified, the PV function is used, where $V = \theta_1\mu^{\theta_2}$ and $\mu = E(Y|X)$.

`convergence(#)` changes the convergence criteria. The default is `convergence(0.0001)`, where the estimating algorithm is iterated until the maximum relative difference in parameter estimates between two successive iterations is less than 0.0001.

`cluster(`*varname*`)` specifies that the observations are independent across groups (clusters), but not necessarily independent within groups. *varname* specifies to which group each observation belongs, e.g., `cluster(personid)` in data with repeated observations on individuals.

`level(#)` specifies the confidence level, as a percentage, for the confidence intervals generated by `ci()`. The default is `level(95)` or as set by `set level`.

`iteration(#)` specifies the number of iterations for the estimation algorithm. The default is `iteration(500)`.

Options for pglmpredict

predtype identifies the type of prediction that the *newvar* should store. The type is one of the following:

`xb`, the default, calculates the linear prediction for each observation from the fitted model.

`mu` calculates predicted outcome for each observation.

`ie(`*varname*`)` calculates the incremental effect for an indicator variable via method of recycled predictions for each observation.

`me(`*varname*`)` calculates the marginal effect for a continuous variable for each observation.

`scale(#)` multiplies *predtype*`[i]` with the value of the scale specified. The default is `scale(1)`.

`se(`*newvar*`)` adds *newvar* of storage type *type*, where for each `i` (observation) in the prediction sample, *newvar*`[i]` contains the estimated standard error of *predtype*`[i]`.

`variance(`*newvar*`)` adds *newvar* of storage type *type*, where for each `i` (observation) in the prediction sample, *newvar*`[i]` contains the estimated variance of *predtype*`[i]`.

`wald(`*newvar*`)` adds *newvar* of storage type *type*, where for each `i` (observation) in the prediction sample, *newvar*`[i]` contains the Wald test statistic for the test of the hypothesis H_o: *predtype*`[i]` $= 0$.

`p(`*newvar*`)` adds *newvar* of storage type *type*, where for each `i` (observation) in the prediction sample, *newvar*`[i]` contains the significance level (p-value) of the Wald test of the hypothesis H_o: *predtype*`[i]` $= 0$.

`ci(`*newvar_ll newvar_ul*`)` creates variables containing the limits for 95% confidence intervals for the predicted values. *newvar_ll* contains the lower limit and *newvar_ul* contains the upper limit.

`level(#)` specifies the confidence level, as a percentage, for the confidence intervals generated by `ci()`. The default is `level(95)` or as set by `set level`.

`g(`*stub*`)` specifies that new variables, *stub*`1`, *stub*`2`, ..., *stub*`k` are to be created, where `k` is the dimension of `e(b)`. *stub*`1` will contain the observation-specific derivatives of *predtype* with respect to the first coefficient listed in `e(b)`, *stub*`2` will contain the derivatives of *predtype* with respect to the second coefficient listed in `e(b)`, etc. If the derivative of *predtype* with respect to a particular coefficient in `e(b)` equals zero for all observations in the prediction sample, the *stub* variable for that coefficient is not created.

`iterate(#)` specifies the maximum number of iterations used to find the optimal step size in the calculation of the numerical derivatives of *predtype* with respect to estimated model coefficients. By default, the maximum number of iterations is 100, but convergence is usually achieved after only a few iterations. You should rarely have to use this option.

`force` forces the calculation of standard errors and other inference-related quantities in situations where `pglmpredict` would otherwise refuse to do so. The calculation of standard errors takes place by evaluating the numerical derivative of *predtype* with respect to the coefficient vector `e(b)`. If `pglmpredict` detects that *predtype* is possibly a function of random quantities other than `e(b)`, it will refuse to calculate standard errors, Wald tests, p-values, etc. The `force` option forces the calculation to take place anyway. If you use the `force` option, there is no guarantee that any inference quantities (e.g., standard errors) will be correct or that the values can be interpreted.

4 Example

4.1 Data

The estimator is illustrated using a simple analysis of annual earnings from the 2000 Current Population Survey (CPS) March dataset. Interest lies in estimating the adjusted incremental effect (IE) on earnings of women compared with men and the marginal effect (ME) of age on earnings. We do not incorporate additional features of the labor market, such as school tuition, nonpecuniary costs of schooling, income taxes, and an endogenous length of working life in the model. We ignore these features mainly for the sake of simplicity in illustrating the performance of this estimator. We restrict our analysis to full-time working adults who are 18–64 years old, are not self-employed, and have positive earnings. We adjust for education categories (`hischool`, `somcoll`, `college`, `hiedu`), age centered at 40 years (`age`), age-squared (`age2`), racial categories (`black`, `other`), marital status (`married`, `nvmarried`), and indicator for women (`female`). Less than high school education, whites, and other marital status are used as reference categories. The outcome variable is personal annual earnings (`pearnval`). We use the CPS March Supplement weights (`marsupwt`) to do a weighted analysis. I start by scaling the outcome variable by its mean.

```
. global incvar "pearnval"
. quietly summarize $incvar, meanonly
. global meany = r(mean)
. generate y = $incvar/$meany
```

(*Continued on next page*)

4.2 Analysis

```
. pglm y hischool somcoll college hiedu age age2 black other female married
> nvmarried [pw=marsupwt]
Iter: 1 Max % Diff: 5.1278375 Rel Diff: 94.872162
  (output omitted)
Iter: 10 Max % Diff: .00005663 Rel Diff: .00013594

Extended GEE with Power Variance Function        No of obs       =       45209
Optimization: Fisher's Scoring                   Residual df     =       45194
[pweight]     marsupwt
Variance:     (theta1*mu^theta2)
Link:         (mu^lambda - 1)/lambda
Std Errors:   Robust
```

y	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
y						
hischool	.3515186	.01375	25.56	0.000	.3245691	.3784681
somcoll	.5382999	.0155596	34.60	0.000	.5078035	.5687962
college	.7950556	.0145366	54.69	0.000	.7665644	.8235469
hiedu	1.10507	.0176625	62.57	0.000	1.070452	1.139688
age	.0147786	.0004029	36.68	0.000	.0139889	.0155682
age2	-.0008238	.0000336	-24.49	0.000	-.0008897	-.0007578
black	-.0789464	.0116374	-6.78	0.000	-.1017554	-.0561374
other	-.0376681	.018954	-1.99	0.047	-.0748174	-.0005188
females	-.3676611	.0072936	-50.41	0.000	-.3819563	-.3533659
married	.0906575	.0100474	9.02	0.000	.070965	.1103501
nvmarried	-.0295077	.0129753	-2.27	0.023	-.0549389	-.0040766
_cons	-.4057019	.0156529	-25.92	0.000	-.436381	-.3750228
lambda						
_cons	-.1199986	.0540904	-2.22	0.027	-.2260139	-.0139833
theta1						
_cons	.4280466	.0116302	36.80	0.000	.4052518	.4508414
theta2						
_cons	1.857664	.1989125	9.34	0.000	1.467803	2.247525

The link parameter is estimated to be $\widehat{\lambda} = -0.12$ (95% CI: -0.23, -0.01), indicating that the optimal link for these data is neither identity (as in OLS) nor log (as in log-OLS and gamma with log link) but more likely an inverse 10th root link. This result shows the importance of using the Box–Cox link transform. The estimated PV model parameters indicate that the variance is a quadratic function of the mean ($\widehat{\theta}_1 = 0.43$, 95% CI: 0.41, 0.45, and $\widehat{\theta}_2 = 1.86$, 95% CI: 1.47, 2.25), which suggests that the variance function of $(Y|X)$ is close to that of a gamma distribution.

4.3 Postestimation results

To estimate IE on annual earnings of being a woman rather than a man, we use the `pglmpredict` command as follows:

```
. pglmpredict iefemale, ie(females) var(varfem) scale($meany)
. summarize iefemale varfem
    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
    iefemale |     45209   -12658.96    5588.459  -28689.61  -2996.324
      varfem |     45209    231885.9    514959.9   6594.954    3007460
```

The IE is estimated to be $-\$12,659$, indicating that the adjusted annual earnings for women are on average \$12,659 less than that of men. The variable `varfem` gives the variance of the IE for each observation, conditional on the values of other characteristics for that observation. However, to obtain the variance of the overall IE, we must also include the part that arises out of the uncertainty in Xs. An estimate for this part is given by the variance of the mean IE, i.e., $(5588.459^2/45209)$ (Basu and Rathouz 2005). Therefore, following (1),

$$\text{Var(IE)} = (5588.459^2/45209) + 231885.9 \approx\ 232577$$

and the standard error for IE $= \sqrt{232577} \approx\ 482$. One can also use bootstrap methods to estimate this variance.

One could also have estimated the marginal effects of `age` on adjusted earnings using the `pglmpredict, me(age)` command, had `age` entered the specification only as a main effect. However, since `age` has a quadratic specification in the model, the ME is given by $\partial\mu/\partial\texttt{age} = \mu^{(1-\lambda)} \cdot\ (\beta_{\texttt{age}} + 2 \cdot\ \texttt{age} \cdot\ \beta_{\texttt{age2}})$, and we rely on `predictnl` to estimate this ME.

```
. global lm=e(lambda)
. predictnl dmuage = ((predict(xb)*$lm + 1)^((1-$lm)/$lm)) *
> (_b[y:age]+2*_b[y:age2]*age)*$meany, var(vardmu)
. summarize dmuage vardmu
    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
      dmuage |     45209    430.4602    567.3527  -1909.251   2166.114
      vardmu |     45209    779.9581    969.4231    36.6932   7753.351
```

The output indicates that in this simple analysis, *ceteris paribus*, earnings change by \$430 on average for each one-year increase in age. The standard error of this ME is calculated using a formula analogous to (1) and is 28.0.

The value of this ME can vary with the base age. One can therefore estimate the ME conditional on a base age and its 95% confidence interval (based to the estimated standard error for the ME) and plot the effects across age (figure 1). The ME of age at a specific base age is estimated after averaging over the empirical distribution of the remaining covariates in the model, i.e., $E_{X_{\texttt{age}}}(\partial\mu(X)/\partial\texttt{age}|\texttt{age} = a)$. Figure 1 shows that an increase in age results in increases in earnings up to about age 47, after which an increase in age results in decreases in earnings.

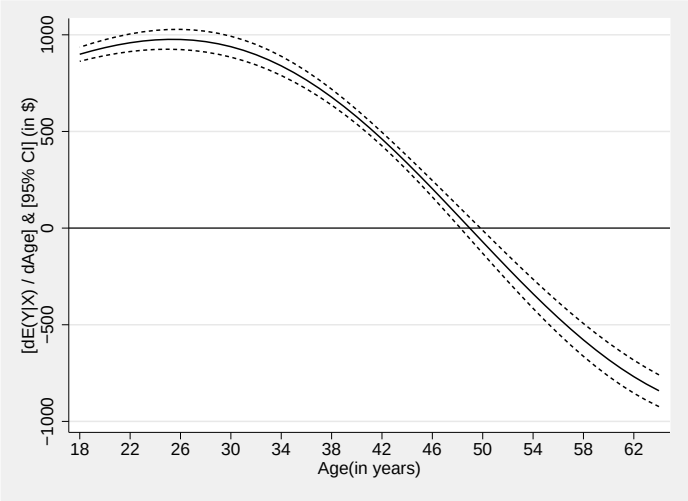

Figure 1: Plot of marginal effect of age and 95% confidence interval on earnings.

4.4 Goodness of fit

It is always useful to carefully study whether any estimator can provide a good fit for the data. There are two simple goodness-of-fit tests that I will illustrate here. One can also run more complex tests of fit.

1. The mean residuals across deciles of the corresponding linear predictor $\widehat{\eta} = x^T\widehat{\beta}$. By looking at the pattern in the residuals as a function of $\widehat{\eta}$, we can determine whether there is a systematic pattern of bias in the forecasts. A formal version of this test is provided by a variant of test of goodness of fit proposed by Hosmer and Lemeshow (2000), using an F test that the mean residuals across all 10 of the deciles are not significantly different from zero. If the residual pattern is U-shaped, there is evidence for a more nonlinear response than was assumed.

2. We present the Pearson correlation between the raw-scale (y-scale) residual and $\widehat{\mu}$. If this statistic is significantly different from zero, the model provides a biased prediction of $E(Y|X)$.

```
. predict xb, xb
. pglmpredict mu, mu scale($meany)
. generate res = $incvar-mu
. summarize res
```

Variable	Obs	Mean	Std. Dev.	Min	Max
res	45209	-1.137649	25347.73	-84020.66	337028.3

```
. pwcorr res mu, sig
             |      res       mu
-------------+------------------
         res |   1.0000
             |
             |
          mu |  -0.0008   1.0000
             |   0.8614
```

```
. xtile xbtile=xb, nq(10)
. quietly tabulate xbtile, generate(xbt)
. regress res xbt1-xbt10, noconstant robust
Linear regression                               Number of obs =   45209
                                                F( 10, 45199) =    1.79
                                                Prob > F      =  0.0559
                                                R-squared     =  0.0003
                                                Root MSE      =   25346
```

res	Coef.	Robust Std. Err.	t	P>|t|	[95% Conf.	Interval]
xbt1	-130.9101	166.8919	-0.78	0.433	-458.021	196.2008
xbt2	-177.1738	188.63	-0.94	0.348	-546.8917	192.5441
xbt3	485.7159	263.5742	1.84	0.065	-30.89386	1002.326
xbt4	-257.1916	252.9759	-1.02	0.309	-753.0286	238.6454
xbt5	-249.918	276.4238	-0.90	0.366	-791.7132	291.8771
xbt6	567.8008	306.2321	1.85	0.064	-32.41916	1168.021
xbt7	589.9669	344.8832	1.71	0.087	-86.00981	1265.944
xbt8	-872.2802	413.2617	-2.11	0.035	-1682.28	-62.28058
xbt9	-242.8651	512.2215	-0.47	0.635	-1246.828	761.0974
xbt10	284.3003	715.8617	0.40	0.691	-1118.8	1687.401

```
. test xbt1 xbt2 xbt3 xbt4 xbt5 xbt6 xbt7 xbt8 xbt9 xbt10
  (output omitted)
       F( 10, 45199) =    1.79
            Prob > F =    0.0559
```

The EEE model seems to fit the data well. It passes both the Hosmer–Lemeshow and the Pearson's correlation tests at the 5% level and shows no systematic patterns in the residuals across the deciles of the linear predictor.

4.5 Comparison with traditional GLM

We compare our results with those given by the traditional `glm` command in Stata, where we use the optimal values for the link and the variance functions for this data.

```
. glm y hischool somcoll college hiedu age age2 black other female married
> nvmarried [pw=marsupwt], link(power $lm) family(gamma) robust irls

Iteration 1:   deviance =  3.49e+07
Iteration 2:   deviance =  3.43e+07
Iteration 3:   deviance =  3.43e+07
Iteration 4:   deviance =  3.43e+07
Iteration 5:   deviance =  3.43e+07
Iteration 6:   deviance =  3.43e+07
Iteration 7:   deviance =  3.43e+07

Generalized linear models                         No. of obs      =     45209
Optimization     : MQL Fisher scoring             Residual df     =     45197
                   (IRLS EIM)                     Scale parameter =         1
Deviance         =   34299298.8                   (1/df) Deviance =  758.8844
Pearson          =  40058395.16                   (1/df) Pearson  =  886.3065

Variance function: V(u) = u^2                     [Gamma]
Link function    : g(u) = u^(-.1199986189117248)  [Power]

                                                  BIC             =  3.38e+07
```

y	Coef.	Semi-Robust Std. Err.	z	P>|z|	[95% Conf.	Interval]
hischool	-.041239	.001776	-23.22	0.000	-.0447199	-.0377582
somcoll	-.0643973	.0019252	-33.45	0.000	-.0681706	-.060624
college	-.0961145	.0018093	-53.12	0.000	-.0996607	-.0925683
hiedu	-.1323003	.0021448	-61.69	0.000	-.1365039	-.1280966
age	-.001768	.000045	-39.26	0.000	-.0018562	-.0016797
age2	.0001046	3.70e-06	28.28	0.000	.0000973	.0001118
black	.0131015	.00139	9.43	0.000	.0103771	.0158259
other	.0053822	.0022476	2.39	0.017	.000977	.0097873
females	.0433806	.0008674	50.01	0.000	.0416804	.0450807
married	-.0111553	.0011778	-9.47	0.000	-.0134637	-.0088469
nvmarried	.003831	.0015362	2.49	0.013	.0008201	.0068418
_cons	1.045521	.0018573	562.91	0.000	1.041881	1.049162

```
. predictnl iefemglm = ((predict(xb) - _b[female]*female + _b[female])^(1/$lm)
> - (predict(xb) - _b[female]*female)^(1/$lm))*$meany, var(varfemglm) force

. sum iefemglm varfemglm
```

Variable	Obs	Mean	Std. Dev.	Min	Max
iefemglm	45209	-12760.9	5708.93	-29075.93	-2875.237
varfemglm	45209	114235.9	116623.9	5883.997	841335.7

The regression coefficients obtained from running this model are not comparable to those obtained from the `pglm` command because `pglm` implements the model $(\mu^\lambda - 1)/\lambda = X^T\beta$, whereas the `power` link option in `glm` implements the model $\mu^\lambda = X^T\beta'$. Clearly, $\beta' \neq \beta$. Therefore, comparison between `pglm` and `glm` estimators is made based on the estimates of IEs and MEs. `glm` with the power-link function estimates the IEs and MEs to be -12760.9 (SE $= 339$) and 434.8 (SE $= 27.5$), respectively, similar to what `pglm` estimated.

5 Conclusions

I have illustrated the use of the `pglm` command that simultaneously solves the EEE estimator for parameters in the link and variance functions along with those of the linear predictor in a GLM (Basu and Rathouz 2005). Our method addresses difficulties in choosing the correct link and variance functions in these models. The method decouples the scale of estimation for the mean model, determined by the link function, from the scale of interest for the scientifically relevant effects. Regardless of what link function is used, MEs and IEs on any scale can be obtained.

A formal test of choosing between scales of estimation can be based on the 95% confidence interval for the link parameter λ. In my example, the 95% confidence interval for λ identifies that the traditional scales of estimation (additive in OLS or proportional in log-OLS or gamma with log link) are incorrect. Thus this test may directly allow researchers to eliminate alternative competing estimators.

On a practical level, the `pglm` estimator works best in analyses with larger sample sizes (say, $N > 5,000$), which are common in health economics and health policy applications.

I hope that this methodology and the `pglm` command will be increasingly used in the health economics and other areas of research that are plagued by data characteristics that makes *a priori* choices of link functions and of estimators with distributional assumptions difficult.

6 Acknowledgments

I thank Paul J. Rathouz and the anonymous reviewer for their helpful suggestions in developing the software.

I also acknowledge the support of the National Institute on Alcohol Abuse and Alcoholism (NIAAA) grant 1 RO1 AA12664-01 A2.

7 References

Basu, A. and P. J. Rathouz. 2005. Estimating marginal and incremental effects on health outcomes using flexible link and variance function models. *Biostatistics* 6(1): 93–109.

Blough, D. K., C. W. Madden, and M. C. Hornbrook. 1999. Modeling risk using generlized linear models. *Journal of Health Economics* 18: 153–171.

Box, G. E. P. and D. R. Cox. 1964. An analysis of transformations. *Journal of the Royal Statistical Society, Series B* 26: 211–252.

Cox, N. J. 2000. dm79: Yet more new matrix commands. *Stata Technical Bulletin* 56: 4–8. In *Stata Technical Bulletin Reprints*, vol. 10, 17–23. College Station, TX: Stata Press.

Guan, W. and R. G. Gutierrez. 2002. Programmable GLM: two user-defined links. *Stata Journal* 2(4): 378–390.

Hardin, J. W. and J. M. Hilbe. 2001. *Generalized Linear Models and Extensions*. College Station, TX: Stata Press.

Hosmer, D. W., Jr., and S. Lemeshow. 2000. *Applied Logistic Regression*. 2nd ed. New York: Wiley.

Huber, P. J. 1967. The behavior of maximum likelihood estimates under nonstandard conditions. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, vol. 1, 221–233. Berkeley, CA: University of California Press.

Manning, W. G. and J. Mullahy. 2001. Estimating log models: to transform or not to transform? *Journal of Health Economics* 20: 461–494.

McCullagh, P. and J. A. Nelder. 1989. *Generalized Linear Models*. 2nd ed. London: Chapman & Hall.

Nelder, J. A. and D. Pregibon. 1987. An extended quasi-likelihood function. *Biometrika* 74: 221–232.

StataCorp. 2005. *Stata 9 Base Reference Manual*. College Station, TX: Stata Press.

Wedderburn, R. W. M. 1974. Quasilikelihood functions, generalized linear models, and the Gauss–Newton method. *Biometrika* 61(3): 439–447.

White, H. 1980. A heteroskedasticity-consistent covariance matrix estimator and a direct test for heteroskedasticity. *Econometrica* 48: 817–838.

Wooldridge, J. M. 1992. Some alternatives to the Box–Cox regression model. *International Economic Review* 33: 935–955.

About the Author

Anirban Basu is an instructor in the Section of General Internal Medicine at the University of Chicago and a project analyst in the Decision and Information Sciences Division at the Argonne National Laboratory.