The Stata Journal

The *Stata Journal* publishes reviewed papers together with shorter notes or comments, regular columns, book reviews, and other material of interest to Stata users. Examples of the types of papers include 1) expository papers that link the use of Stata commands or programs to associated principles, such as those that will serve as tutorials for users first encountering a new field of statistics or a major new technique; 2) papers that go "beyond the Stata manual" in explaining key features or uses of Stata that are of interest to intermediate or advanced users of Stata; 3) papers that discuss new commands or Stata programs of interest either to a wide spectrum of users (e.g., in data management or graphics) or to some large segment of Stata users (e.g., in survey statistics, survival analysis, panel analysis, or limited dependent variable modeling); 4) papers analyzing the statistical properties of new or existing estimators and tests in Stata; 5) papers that could be of interest or usefulness to researchers, especially in fields that are of practical importance but are not often included in texts or other journals, such as the use of Stata in managing datasets, especially large datasets, with advice from hard-won experience; and 6) papers of interest to those who teach, including Stata with topics such as extended examples of techniques and interpretation of results, simulations of statistical concepts, and overviews of subject areas.

The *Stata Journal* is indexed and abstracted by *CompuMath Citation Index*, *Current Contents/Social and Behavioral Sciences*, *RePEc: Research Papers in Economics*, *Science Citation Index Expanded* (also known as *SciSearch*), *Scopus*, and *Social Sciences Citation Index*.

For more information on the *Stata Journal*, including information for authors, see the webpage

http://www.stata-journal.com

Subscriptions are available from StataCorp, 4905 Lakeway Drive, College Station, Texas 77845, telephone 979-696-4600 or 800-STATA-PC, fax 979-696-4601, or online at

http://www.stata.com/bookstore/sj.html

Subscription rates listed below include both a printed and an electronic copy unless otherwise mentioned.

U.S. and Canada		Elsewhere	
Printed & electronic		**Printed & electronic**	
1-year subscription	$115	1-year subscription	$145
2-year subscription	$210	2-year subscription	$270
3-year subscription	$285	3-year subscription	$375
1-year student subscription	$ 85	1-year student subscription	$115
1-year institutional subscription	$345	1-year institutional subscription	$375
2-year institutional subscription	$625	2-year institutional subscription	$685
3-year institutional subscription	$875	3-year institutional subscription	$965
Electronic only		**Electronic only**	
1-year subscription	$ 85	1-year subscription	$ 85
2-year subscription	$155	2-year subscription	$155
3-year subscription	$215	3-year subscription	$215
1-year student subscription	$ 55	1-year student subscription	$ 55

Back issues of the *Stata Journal* may be ordered online at

http://www.stata.com/bookstore/sjj.html

Individual articles three or more years old may be accessed online without charge. More recent articles may be ordered online.

http://www.stata-journal.com/archives.html

The *Stata Journal* is published quarterly by the Stata Press, College Station, Texas, USA.

Address changes should be sent to the *Stata Journal*, StataCorp, 4905 Lakeway Drive, College Station, TX 77845, USA, or emailed to sj@stata.com.

The *Stata Journal* (ISSN 1536-867X) is a publication of Stata Press. Stata, STATA, Stata Press, Mata, MATA, and NetCourse are registered trademarks of StataCorp LP.

The Stata Journal (2016)
16, Number 1, pp. 159–184

bicop: A command for fitting bivariate ordinal regressions with residual dependence characterized by a copula function and normal mixture marginals

Mónica Hernández-Alava
School of Health and Related Research (ScHARR)
Health Economics and Decision Science
University of Sheffield
Sheffield, UK
monica.hernandez@sheffield.ac.uk

Stephen Pudney
Institute for Social and Economic Research (ISER)
University of Essex
Colchester, UK
spudney@essex.ac.uk

Abstract. In this article, we describe a new Stata command, `bicop`, for fitting a model consisting of a pair of ordinal regressions with a flexible residual distribution, with each marginal distribution specified as a two-part normal mixture, and stochastic dependence governed by a choice of copula functions. The `bicop` command generalizes the existing `biprobit` and `bioprobit` commands, which assume a bivariate normal residual distribution. We present and explain the `bicop` estimation command and the available postestimation commands using data on financial well-being from the UK Understanding Society Panel Survey.

Keywords: st0429, bicop, bivariate ordinal regression, copula, mixture model

1 Introduction

We are often interested in modeling the joint distribution of two observed measures conditional on a set of observed covariates. For example, income and wealth are two strongly related aspects of economic welfare that should, arguably, be studied jointly; drinking and smoking, particularly when combined, have important health implications and should thus be studied jointly; and joint analysis of different domains of satisfaction has been used in “happiness” research. Methodological issues also often take this form and ask how two alternative measures of the same theoretical concept may be related.

Frequently, the indicators concerned are coarse binary or ordinal measures rather than direct observations on the relevant theoretical concepts, and this naturally suggests using a pair of correlated ordinal probit or logit regressions. Stata already provides the command `biprobit` for the case of a pair of binary indicators and the user-written command `bioprobit` (Sajaia 2008) for the more general ordinal case. However, `biprobit` and `bioprobit` are based on the assumption of joint normality, which may be hard to defend. In many applications, the influence of observed covariates has a pronounced nonnormal distributional shape, and there is no compelling reason to assume that the factors we cannot observe conform to normality when the factors we can observe do not. Moreover, the linear form of stochastic dependence implied by bivariate normality may be unduly restrictive: there is no reason why the nature and degree of dependence should not vary across different parts of the population.

Models of this type are not distribution free, and misspecification of the joint residual distribution may cause significant bias in the estimated coefficients of the covariates and may give a distorted picture of stochastic dependence. We developed the `bicop` command as a method of estimating a more general specification of the bivariate ordinal model, using mixtures to allow for nonnormality and copula representations to allow for complex forms of dependence.

The article is organized as follows: in section 2, we give an overview of the generalized bivariate ordinal regression model and the approach we use to allow for nonnormality in the residual distribution. In section 3, we discuss two hypothesis tests that are relevant to `bicop`. In section 4, we explain the predictors that are provided postestimation. In section 5, we describe the `bicop` syntax and options, including the syntax for `predict`. In section 6, we conclude with an empirical example using the `bicop` command.

2 The generalized bivariate ordinal regression model

The generalized bivariate ordinal regression model is

$$Y_{i1}^* = X_{i1}\beta_1 + U_i \tag{1}$$

$$Y_{i2}^* = X_{i2}\beta_2 + V_i \tag{2}$$

where Y_{i1}^* and Y_{i2}^* are latent variables, X_{i1} and X_{i2} are row vectors of covariates, and β_1 and β_2 are conformable column vectors of coefficients. U_i and V_i are unobserved residuals that may be stochastically dependent and nonnormal. The covariate vectors X_{i1} and X_{i2} may contain the same or different variables.

The observable counterparts of Y_{i1}^* and Y_{i2}^* are generated by the threshold-crossing conditions

$$Y_{ij} = r \quad \text{iff} \quad \Gamma_{rj} \leq Y_{ij}^* < \Gamma_{r+1j} \qquad r = 1, \ldots, R_j \text{ and } j = 1, 2$$

where R_j is the number of categories of Y_{ij} and Γ_{rj} are threshold parameters, with $\Gamma_{1j} = -\infty$ and $\Gamma_{R_j j} = +\infty$. (Note that in practice, the Y_{ij} do not have to be scored as 1, 2, 3, ...; `bicop` will work, whatever numerical values are used to index outcomes—only their ordering matters.)

The likelihood function requires evaluation of the probability that (Y^*_{i1}, Y^*_{i2}) falls in a rectangle corresponding to the observed values of (Y_{i1}, Y_{i2}). For given parameter values, that probability can be computed using the joint distribution function $F(U_i, V_i)$, which allows the likelihood to be maximized numerically. However, if the assumed form for $F(U_i, V_i)$ is incorrect, the probabilities in the likelihood function will be misspecified, and the (pseudo) maximum likelihood estimator will be inconsistent. This means that the standard approach using a bivariate normal form for $F(.,.)$ is potentially vulnerable to bias. On the other hand, a full nonparametric specification for $F(.,.)$ would be complicated and unlikely to provide reliable estimates except in large samples, so an intermediate degree of flexibility is desirable.

The model specification is based on a copula representation of the joint distribution of the residuals U and V. A bivariate copula is any function $c(u, v) : [0, 1]^2 \to [0, 1]$ that is (weakly) increasing and satisfies $c(u, 0) = c(0, v) = 0$, $c(u, 1) = u$, and $c(1, v) = v$ for all $u, v \in [0, 1]$. By adding a parameter θ governing the stochastic dependence of U and V, we can write the joint residual distribution function as

$$F(U, V) = c\{F_u(U), F_v(V); \theta\}$$

where $F_u(U) \equiv F(U, +\infty)$ and $F_v(V) \equiv F(+\infty, V)$ are the marginal distribution functions of U and V. The `bicop` command generalizes the standard bivariate normal model in the following ways:

- Marginals: `bicop` allows the marginal distributions $F_u(.)$ and $F_v(.)$ to be specified as mixtures of two normal components. For $F_u(.)$,

$$F_u(u) = \pi_u \Phi\left(\frac{u - \mu_{u1}}{\sigma_{u1}}\right) + (1 - \pi_u)\Phi\left(\frac{u - \mu_{u2}}{\sigma_{u2}}\right) \tag{3}$$

  where π_u is the mixing probability, and (μ_{u1}, μ_{u2}) and $(\sigma_{u1}, \sigma_{u2})$ are location and dispersion parameters constrained to satisfy the mean and variance normalizations $\pi_u\mu_{u1} + (1 - \pi_u)\mu_{u2} \equiv 0$ and $\pi_u(\sigma^2_{u1} + \mu^2_{u1}) + (1 - \pi_u)(\sigma^2_{u2} + \mu^2_{u2}) = 1$. A similar specification can be used for $F_v(.)$. These normal mixtures can capture various distributional shapes, especially those involving skewness or bimodality.

  The `bicop` command performs the optimization with respect to $\ln\{\pi_u/(1 - \pi_u)\}$ rather than π_u, but both values are reported in the output. In the Stata output log, the mixing parameters π_u, $(1-\pi_u)$, μ_{u1}, μ_{u2}, σ^2_{u1}, and σ^2_{u2} are labeled `pi_u_1`, `pi_u_2`, `mean_u_1`, `mean_u_2`, `var_u_1`, and `var_u_2` for (1) and, analogously, `pi_v_1`, `pi_v_2`, `mean_v_1`, `mean_v_2`, `var_v_1`, and `var_v_2` for (2).[1]

- Dependence: The `bicop` command offers the following six forms as options:

  – Independent: $c(u, v) = uv$.

1. The auxiliary parameters that are optimized during estimation are also written to the output log, with labels `/pu1`, `/mu2`, `/su2`, `/pv1`, `/mv2`, and `/sv2`. These parameters are transformations of the mixing parameters and can be ignored when interpreting the output of the model.

- Gaussian: $c(u,v) = \Phi\left\{\Phi^{-1}(u), \Phi^{-1}(v); \theta\right\}$, where $\Phi(.,.;\theta)$ is the distribution function of the bivariate normal with correlation coefficient $-1 \le \theta \le 1$, and $\Phi^{-1}(.)$ is the inverse of the univariate $N(0,1)$ distribution function.
- Clayton: $c(u,v) = \left\{\max\left(u^{-\theta} + v^{-\theta} - 1, 0\right)\right\}^{-1/\theta}$ for $0 < \theta \le \infty$ and $c(u,v) = uv$ for $\theta = 0$.
- Frank: $-(1/\theta)\ln\left\{1 + \frac{(e^{-\theta u}-1)(e^{-\theta v}-1)}{e^{-\theta}-1}\right\}$ for $\theta \neq 0$ and $c(u,v) = uv$ for $\theta = 0$.
- Gumbel: $\exp\left[-\left\{(-\ln u)^{\theta} + (-\ln v)^{\theta}\right\}^{1/\theta}\right]$ for $\theta \ge 1$.
- Joe: $1 - \left\{(1-u)^{\theta} + (1-v)^{\theta} - (1-u)^{\theta}(1-v)^{\theta}\right\}^{1/\theta}$ for $\theta \ge 1$.

These copulas can represent various dependence structures. The Gaussian and the Frank copulas are similar in that both allow for positive and negative dependence, and dependence is symmetric in both tails. However, compared with the Gaussian copula, the Frank copula exhibits weaker dependence in the tails, and dependence is strongest in the middle of the distribution. In contrast, the Clayton, Gumbel, and Joe copulas do not allow for negative dependence, and dependence in the tails is asymmetric. The Clayton copula exhibits strong left-tail dependence and relatively weak right-tail dependence. Thus, if two variables are strongly correlated at low values but not so correlated at high values, then the Clayton copula is a good choice. The Gumbel and Joe copulas display the opposite pattern with weak left-tail dependence and strong right-tail dependence. The right-tail dependence is stronger in the Joe copula than in the Gumbel, and thus the Joe copula is closer to the opposite of the Clayton copula.

`bicop` maximizes the likelihood with respect to an unrestricted constant $\delta \in [-\infty, +\infty]$, with θ related to δ in the following ways:

$$\theta = \begin{cases} \tanh(\delta) & \text{Gaussian} \\ e^{\delta} & \text{Clayton} \\ \delta & \text{Frank} \\ e^{\delta} + 1 & \text{Gumbel, Joe} \end{cases}$$

The output from `bicop` reports both δ (labeled as `/depend`) and θ.

Both mixture and copula models can be difficult to fit in some circumstances (see McLachlan and Peel [2000] on the former and Trivedi and Zimmer [2005] on the latter). Two distinct problems await the unwary. Nonconvergence of the likelihood optimizer often occurs in copula models, typically for some choices of copula function but not others. The problem occurs when the chosen copula function does a poor job of representing the pattern of dependence between the two residuals, and it can often be resolved by switching to a different copula function; we see an example of this in section 6, where convergence cannot be achieved for the Gumbel and Joe copulas. Poor starting values can also cause nonconvergence; restarting the optimizer from a different point in the parameter space will work in some cases.

Another possible reason for nonconvergence is local nonidentification of the mixture parameters. For the normal mixture (3), the parameter π_u is not identified at interior points in the parameter space where $\mu_{u1} = \mu_{u2}$ and $\sigma_{u1} = \sigma_{u2}$. Boundary problems also arise because μ_{u1}, σ_{u1} are not identified when $\pi_u = 0$, nor are μ_{u2}, σ_{u2} identified when $\pi_u = 1$. All three regions correspond to a pure $N(0,1)$ distribution.[2] Consequently, if either of the marginal distributions is approximately normal, identification will be weak and nonconvergence a likely result. These cases usually become evident if the `log` and `trace` options are used to display current parameter values during optimization. When this occurs, the relevant marginal can be respecified as an unmixed normal in a subsequent run.

Related to this last type of nonconvergence problem is the problem of testing for the appropriate number of mixture components. Standard likelihood-ratio tests of $H_0 : U \sim N(0,1)$ or $V \sim N(0,1)$ against a two-component normal mixture do not work correctly in this nonregular context (Titterington, Smith, and Makov 1985, 154), and we are not aware of any alternative formal procedure that is entirely satisfactory.

The problem of multiple optima is less obvious than nonconvergence—and, therefore, more dangerous. The existence of multiple optima poses problems for likelihood maximization in many mixture models and should be assumed to be a potential pitfall. The `bicop` command offers the standard Stata optimization options for starting values (see [R] **maximize**), and the application in section 6 provides an example of a recommended starting-values strategy.

3 Hypothesis tests

Two hypothesis tests may be of special interest in particular applications of `bicop`. One is the hypothesis test of conditional independence: $Y_1 \perp\!\!\!\perp Y_2 | X_1, X_2$, which holds if and only if $c(u,v) = uv$ for all $u, v \in [0,1]$. This independence condition is equivalent to $\theta = 0$ for the Gaussian, Clayton, and Frank copulas and $\theta = 1$ for the Gumbel and Joe copulas. For the Gaussian and Frank copulas, this involves a regular likelihood-ratio or Wald test, which can be done in the usual way. For these copula functions, `bicop` produces a Wald test automatically. For the Clayton, Gumbel, and Joe functions, the null hypothesis is on the boundary of the parameter space, and the likelihood-ratio and Wald tests are not valid (see Chernoff [1954] and Andrews [2001]). Because these copulas are a natural choice in applications only where we are confident of positive dependence, `bicop` does not produce an automatic test in these cases. Instead, if the test is required, the user could fit the model unrestrictedly using the Clayton, Gumbel, or Joe copula, repeat estimation while imposing independence by specifying the copula $c = uv$, and then construct the usual statistic of minus twice the log-likelihood ratio. The complication here is that the test statistic has a nonstandard limiting distribution, that is, $\overline{\chi}^2$ [a 50:50 mixture of a degenerate probability mass at zero and a $\chi^2(1)$ distribution]. This amounts to performing a standard $\chi^2(1)$ likelihood-ratio test and then halving the p-value (see Chernoff [1954]).

2. The variance of the distribution is normalized to 1 for identification purposes in an ordered probit model.

The second special hypothesis test of interest in some applications of `bicop` is the hypothesis of equal coefficients, $H_0: \beta_1 = \beta_2$, which will normally arise when X_1 and X_2 contain the same variables. This null hypothesis arises naturally when Y_1 and Y_2 are interpreted as alternative measures of the same concept—for example, they might be responses to the same survey questions, repeated with different response scales. A test can be performed easily using the standard Stata command `test`, which implements the Wald test, but for convenience, `bicop` does the test automatically. If X_1 and X_2 are different, the test is made on the coefficients of any variables that are common to both.

4 Prediction

The `bicop` command allows the usual Stata prediction options postestimation, through the evaluation of the linear indices $X_{i1}\beta_1$ and $X_{i2}\beta_2$, the associated prediction standard errors, and the probabilities of specific outcomes for (Y_{i1}, Y_{i2}) conditional on the covariates (X_{i1}, X_{i2}). However, `bicop` additionally has options for conditional prediction. These can be used, for instance, to convert (or "map" or "cross-walk") a measurement scale represented by the dependent variable Y_{i1} into another scale represented by Y_{i2}. Following the use of `bicop`, the `predict` command can convert a measurement scale by constructing estimates of the distribution of one dependent variable conditional on the observed outcome for the other. For example,

$$\Pr(Y_{i2} = s | Y_{i1} = r, X_{i1}, X_{i2}) = \frac{\Pr(Y_{i1} = r, Y_{i2} = s | X_{i1}, X_{i2})}{\sum_{s=1}^{R_2} \Pr(Y_{i1} = r, Y_{i2} = s | X_{i1}, X_{i2})}$$

where $r \in [1, R_1]$ and $s \in [1, R_2]$ are specified levels for the two outcomes.

5 Command syntax

5.1 bicop

Syntax

There are two forms of the syntax:

X_1 and X_2 contain the same covariates

`bicop` *depvar1 depvar2* [*indepvars*] [*if*] [*in*] [*weight*] [`,` *syntax1_options*]

X_1 and X_2 contain different covariates

`bicop (`*equation1*`) (`*equation2*`)` [*if*] [*in*] [*weight*] [`,` *syntax2_options*] *syntax1_options* and *syntax2_options* are as listed in the *Options* section below.

equation1 and *equation2* are specified as

(`[` *eqname*: `]` *depvar* `[` = `]` `[` *indepvars* `]` `[` , offset(*varname*) `]`)

pweights, fweights, and iweights are allowed; see [U] **11.1.6 weight**.

Description

bicop is a user-written command that fits a generalized bivariate ordinal regression model using maximum likelihood estimation. It is implemented as an lf1 ml evaluator. The model involves a pair of latent regression equations, each with a standard threshold-crossing condition to generate ordinal observed dependent variables. The bivariate residual distribution is specified to have marginals, each with the form of a two-part normal mixture, and a choice of copula functions to represent the pattern of dependence between the two residuals.

Options

Options common to both syntax 1 and syntax 2 are the following:

mixture(*mixturetype*) specifies the marginal distribution of each residual. There are five choices for *mixturetype*: none specifies that each marginal distribution be $N(0,1)$; mix1 specifies that the residual from equation 1 has a two-part normal mixture distribution but that the residual from equation 2 be $N(0,1)$; mix2 specifies $N(0,1)$ for equation 1 and a normal mixture for equation 2; both allows each residual to have a different normal mixture distribution; and equal specifies that both residuals have the same normal mixture distribution. The default is mixture(none).

copula(*copulatype*) specifies the copula function to be used to control the pattern of stochastic dependence of the two residuals. There are six choices for *copulatype*: indep, which specifies the special form $c(u,v) = uv$, gaussian, clayton, frank, gumbel, and joe. The default is copula(gaussian). Note that if both mixture() and copula() are omitted, the bicop command produces the same results as the existing bioprobit and (if both dependent variables are binary) biprobit commands.

constraints(*numlist*) applies specified linear constraints; see [R] **constraint**.

collinear retains collinear variables. Usually, there is no reason to leave collinear variables in place, and doing so would cause the estimation to fail because of matrix singularity. However, in some constrained cases, the model may be fully identified despite the collinearity. The collinear option then allows estimation to occur, leaving the equations with collinear variables intact. This option is seldom used.

`vce(`*vcetype*`)` specifies how to estimate the variance–covariance matrix corresponding to the parameter estimates. The supported options are `oim`, `opg`, `robust`, and `cluster`. The current version of the command does not allow `bootstrap` or `jackknife` estimators. See [R] ***vce_option***.

`level(`*#*`)` sets the significance level to be used for confidence intervals; see [R] **level**.

`from(`*init_specs*`)`, where *init_specs* is either *matname*, the name of a matrix containing the starting values, or *matname*`,` `copy` | `skip`. The `copy` suboption specifies that the initialization vector be copied into the initial-value vector by position rather than by name, and the `skip` suboption specifies that any irrelevant parameters found in the specified initialization vector be ignored. Poor values in `from()` may lead to convergence problems.

`search(`*spec*`)` specifies whether `ml`'s ([R] **ml**) initial search algorithm is used. *spec* may be `on` or `off`.

`repeat(`*#*`)` specifies the number of random attempts to be made to find a better initial-value vector. This option should be used in conjunction with `search()`.

maximize_options specifies the maximization options; *maximize_options* are `difficult`, `technique(`*algorithm_spec*`)`, `iterate(`*#*`)`, [`no`]`log`, `trace`, `gradient`, `showstep`, `hessian`, `showtolerance`, `tolerance(`*#*`)`, `ltolerance(`*#*`)`, `gtolerance(`*#*`)`, `nrtolerance(`*#*`)`, and `nonrtolerance`; see [R] **maximize**.

Additional options for syntax 1 only are as follows:

`offset1(`*varname*`)` specifies an offset variable for the first equation.

`offset2(`*varname*`)` specifies an offset variable for the second equation.

5.2 predict

Syntax

`predict` *varname* [*if*] [*in*] [`,` *predicttype* `outcome(`*r*`,`*s*`)`]

Description

Following `bicop`, the `predict` command can be used to construct several alternative predictions. The predictions include the linear indices $X_{i1}\beta_1$ and $X_{i2}\beta_2$ and corresponding standard errors; probabilities of the form $\Pr(Y_{ij} = r|X_{ij})$ or $\Pr(Y_{i1} = r, Y_{i2} = s|X_{i1}, X_{i2})$; and conditional probabilities of the form $\Pr(Y_{ij} = r|Y_{ik} = s, X_{i1}, X_{i2})$.

Options

predicttype specifies the type of prediction required. If *predicttype* is `xb1` or `xb2`, the variable *varname* is constructed as $X_{i1}\beta_1$ or $X_{i2}\beta_2$, respectively. Set *predicttype* to `std1` or `std2` to construct *varname* as the corresponding prediction standard error. If *predicttype* is `pr`, the prediction is calculated as a probability $\Pr(Y_{i1} = r|X_{ij})$, $\Pr(Y_{i2} = r|X_{ij})$, or $\Pr(Y_{i1} = r, Y_{i2} = s|X_{i1}, X_{i2})$ with r and s specified by the `outcome()` option. The *predicttype*s `pcond1` and `pcond2` specify the conditional probabilities $\Pr(Y_{i1} = r|Y_{i2} = s, X_{i1}, X_{i2})$ or $\Pr(Y_{i2} = s|Y_{i1} = r, X_{i1}, X_{i2})$, respectively, with r and s supplied by `outcome()`.

`outcome(`*r*`,`*s*`)` specifies the outcome levels to be used in predicting probabilities for Y_{i1} and Y_{i2}. The possibilities for *predicttype* and `outcome(`*r*`,`*s*`)` are as follows:

Option	Predicted probability
`pr outcome(`*r*`, . )`	$\Pr(Y_{i1} = r\|X_{i1})$
`pr outcome( . ,`*s*`)`	$\Pr(Y_{i2} = s\|X_{i2})$
`pr outcome(`*r*`,`*s*`)`	$\Pr(Y_{i1} = r, Y_{i2} = s\|X_{i1}, X_{i2})$
`pcond1 outcome(`*r*`,`*s*`)`	$\Pr(Y_{i1} = r\|Y_{i2} = s, X_{i1}, X_{i2})$
`pcond2 outcome(`*r*`,`*s*`)`	$\Pr(Y_{i2} = s\|Y_{i1} = r, X_{i1}, X_{i2})$

6 An illustrative application: Financial well-being

We now show how to use the `bicop` command to model bivariate ordinal data. Our example uses data from Understanding Society: the UK Household Longitudinal Survey (UKHLS). See Knies (2015) for a detailed description of the survey. The main UKHLS sample began in 2009 with approximately 30,000 households. Interviewing proceeds continuously through the year with households interviewed annually, but each wave takes two years to complete and thus overlaps with the preceding and succeeding waves. We use a simple dataset comprising a cross-section of 5,482 individual respondents drawn from the calendar years 2011–2012. The dataset is supplied to users with the `bicop` code.

We analyze the responses to the following two questions about financial well-being (FWB), and we construct the variables Y_1 and Y_2 as the corresponding five-level and three-level ordinal indicators, both recoded to give scales increasing in current or expected FWB (see Pudney [2011] for discussion and analysis of this FWB measure).

- "How well would you say you yourself are managing financially these days? Would you say you are ..." [1. Living comfortably 2. Doing alright 3. Just about getting by 4. Finding it quite difficult 5. or finding it very difficult?].
- "Looking ahead, how do you think you will be financially a year from now, will you be ..." [1. Better off 2. Worse off than you are now 3. or about the same?].

Three binary explanatory covariates distinguish people who are female, homeowners, and unemployed or long-term sick and disabled.[3]

The following code fits all six copula models with the `mixture(none)` option. The Clayton copula clearly provides the best likelihood fit. Note that the Gumbel estimate is a boundary solution with $\theta \approx 1$; thus it is also identical to the Joe estimate and the result produced by the `copula(indep)` option (neither of which are reproduced here). The superior fit of the Clayton model and failure of the Gumbel and Joe models to detect any dependence suggest a pattern of strong dependence in the left tail of the residual distribution but not in the right tail.

```
. use ukhlsfwb

. local maxll=minfloat()

. foreach cop in gaussian frank clayton gumbel joe indep {
  2.         local xvars female homeowner unempsick
  3.         bicop finnow finfut `xvars´, copula(`cop´)
  4.         estimates store `cop´
  5.         if e(ll)>`maxll´&e(converged) {
  6.                 local maxll=e(ll)
  7.                 local bestcop="`cop´"
  8.                 matrix bestb=e(b)
  9.         }
 10. }
```

3. A more substantial application with 10 explanatory variables can be found in an earlier version of this paper (Hernández-Alava and Pudney 2015). We cannot make that dataset publicly available because of respondent confidentiality, but the full UKHLS data files are obtainable on application to the UK Data Archive (Study Number 6614) at http://discover.ukdataservice.ac.uk/catalogue/?sn=6614&type=Data%20catalogue.

```
LogL for independent ordered probit model -13062.773
initial:       log likelihood = -16992.008
rescale:       log likelihood = -15050.038
rescale eq:    log likelihood = -13062.146
Iteration 0:   log likelihood = -13062.146
Iteration 1:   log likelihood = -13062.145
Iteration 2:   log likelihood = -13062.145
Generalized bivariate ordinal regression model (copula: gaussian, mixture: none)
                                                Number of obs     =      5,482
                                                Wald chi2(6)      =     907.33
Log likelihood = -13062.145                     Prob > chi2       =     0.0000

------------------------------------------------------------------------------
             |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
finnow       |
      female |  -.1549272   .0296466    -5.23   0.000    -.2130335    -.096821
   homeowner |   .5237826   .0303863    17.24   0.000     .4642266    .5833386
   unempsick |  -.7196592   .0399321   -18.02   0.000    -.7979247   -.6413936
-------------+----------------------------------------------------------------
finfut       |
      female |   -.046568   .0313308    -1.49   0.137    -.1079753    .0148393
   homeowner |  -.2102546   .0320044    -6.57   0.000    -.2729822    -.147527
   unempsick |  -.1461849   .0419871    -3.48   0.000    -.2284782   -.0638916
-------------+----------------------------------------------------------------
   /cuteq1_1 |  -1.592359   .0394148   -40.40   0.000    -1.669611   -1.515108
   /cuteq1_2 |  -.9077473   .0343043   -26.46   0.000    -.9749824   -.8405122
   /cuteq1_3 |   .0811928   .0326669     2.49   0.013     .0171667    .1452188
   /cuteq1_4 |   1.056313   .0348781    30.29   0.000     .9879537    1.124673
   /cuteq2_1 |  -1.054656   .0360324   -29.27   0.000    -1.125278   -.9840339
   /cuteq2_2 |    .475085   .0343894    13.81   0.000      .407683    .5424871
     /depend |   .0179149    .015992     1.12   0.263    -.0134287    .0492586
-------------+----------------------------------------------------------------
       theta |    .017913   .0159868
------------------------------------------------------------------------------
Wald test of equality of coefficients chi2(df = 3)= 521.974 [p-value=0.000]
Wald test of independence chi2(df = 1)= 1.255 [p-value=0.263]
```

```
LogL for independent ordered probit model -13062.773
initial:       log likelihood = -13132.429
rescale:       log likelihood = -13132.429
rescale eq:    log likelihood = -13062.443
Iteration 0:   log likelihood = -13062.443
Iteration 1:   log likelihood = -13062.442
Generalized bivariate ordinal regression model (copula: frank, mixture: none)
                                                Number of obs     =      5,482
                                                Wald chi2(6)      =     907.06
Log likelihood = -13062.442                     Prob > chi2       =     0.0000

------------------------------------------------------------------------------
             |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
finnow       |
      female |  -.1547791   .0296449    -5.22   0.000    -.2128821   -.0966761
   homeowner |   .5239715   .0303861    17.24   0.000     .4644159    .5835271
   unempsick |  -.7196971    .039929   -18.02   0.000    -.7979565   -.6414376
-------------+----------------------------------------------------------------
finfut       |
      female |  -.0465239   .0313291    -1.49   0.138    -.1079278    .0148801
   homeowner |  -.2104716   .0320053    -6.58   0.000    -.2732008   -.1477423
   unempsick |  -.1466051   .0419866    -3.49   0.000    -.2288973   -.0643129
-------------+----------------------------------------------------------------
   /cuteq1_1 |  -1.592102    .039411   -40.40   0.000    -1.669346   -1.514858
   /cuteq1_2 |  -.9075391   .0343019   -26.46   0.000    -.9747696   -.8403086
   /cuteq1_3 |   .0814055   .0326654     2.49   0.013     .0173825    .1454285
   /cuteq1_4 |   1.056547   .0348752    30.30   0.000     .9881931    1.124901
   /cuteq2_1 |   -1.05421   .0360321   -29.26   0.000    -1.124831   -.9835879
   /cuteq2_2 |   .4754599   .0343918    13.82   0.000     .4080533    .5428665
     /depend |   .0770508   .0947965     0.81   0.416    -.1087471    .2628486
-------------+----------------------------------------------------------------
       theta |   .0770508   .0947965
------------------------------------------------------------------------------
Wald test of equality of coefficients chi2(df = 3)= 519.878 [p-value=0.000]
Wald test of independence chi2(df = 1)= 0.661 [p-value=0.416]
```

```
LogL for independent ordered probit model -13062.773
initial:       log likelihood = -17203.534
rescale:       log likelihood = -15145.713
rescale eq:    log likelihood = -13101.382
Iteration 0:   log likelihood = -13101.382
Iteration 1:   log likelihood = -13051.968
Iteration 2:   log likelihood = -13051.923
Iteration 3:   log likelihood = -13051.923
Generalized bivariate ordinal regression model (copula: clayton, mixture: none)
                                                Number of obs     =      5,482
                                                Wald chi2(6)      =     906.32
Log likelihood = -13051.923                     Prob > chi2       =     0.0000

              Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]

finnow
      female   -.1589312   .0296393    -5.36   0.000    -.2170231   -.1008392
   homeowner    .5218558   .0303621    17.19   0.000     .4623471    .5813644
   unempsick   -.7157391   .0399348   -17.92   0.000    -.7940098   -.6374684

finfut
      female   -.0499395   .0313101    -1.59   0.111    -.1113061    .0114272
   homeowner   -.2087876   .0319862    -6.53   0.000    -.2714794   -.1460957
   unempsick   -.1427097   .0419863    -3.40   0.001    -.2250013   -.0604181

   /cuteq1_1   -1.595204   .0393811   -40.51   0.000    -1.672389   -1.518018
   /cuteq1_2   -.9106284   .0342863   -26.56   0.000    -.9778283   -.8434286
   /cuteq1_3    .0782122   .0326526     2.40   0.017     .0142143    .1422101
   /cuteq1_4    1.053243   .0348613    30.21   0.000     .9849157     1.12157
   /cuteq2_1   -1.054752   .0360137   -29.29   0.000    -1.125338   -.9841665
   /cuteq2_2    .4757612   .0343553    13.85   0.000      .408426    .5430964
     /depend    -2.53765    .228445   -11.11   0.000    -2.985394   -2.089906

       theta    .0790519    .018059

Wald test of equality of coefficients chi2(df = 3)= 537.459 [p-value=0.000]
Wald test of independence chi2(df = 1)= 19.162 [p-value=0.000]
```

```
LogL for independent ordered probit model -13062.773
initial:       log likelihood = -19774.602
rescale:       log likelihood = -15862.755
rescale eq:    log likelihood = -13330.654
Iteration 0:   log likelihood = -13330.654
Iteration 1:   log likelihood = -13067.223
Iteration 2:   log likelihood = -13062.777
Iteration 3:   log likelihood = -13062.773
Iteration 4:   log likelihood = -13062.773
Generalized bivariate ordinal regression model (copula: gumbel, mixture: none)
                                                Number of obs     =      5,482
                                                Wald chi2(6)      =     905.06
Log likelihood = -13062.773                     Prob > chi2       =     0.0000

------------------------------------------------------------------------------
             |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
finnow       |
      female |  -.1548403   .0296463    -5.22   0.000    -.2129459   -.0967347
   homeowner |   .5238116   .0303864    17.24   0.000     .4642554    .5833677
   unempsick |  -.7195785   .0399312   -18.02   0.000    -.7978422   -.6413147
-------------+----------------------------------------------------------------
finfut       |
      female |  -.0465534   .0313303    -1.49   0.137    -.1079598    .0148529
   homeowner |  -.2102062   .0320046    -6.57   0.000     -.272934   -.1474784
   unempsick |  -.1461061   .0419857    -3.48   0.001    -.2283965   -.0638157
-------------+----------------------------------------------------------------
   /cuteq1_1 |  -1.592221   .0394137   -40.40   0.000    -1.669471   -1.514972
   /cuteq1_2 |  -.9077398    .034304   -26.46   0.000    -.9749744   -.8405052
   /cuteq1_3 |   .0812235   .0326666     2.49   0.013     .0171981    .1452488
   /cuteq1_4 |   1.056424   .0348779    30.29   0.000     .9880641    1.124783
   /cuteq2_1 |  -1.054634   .0360286   -29.27   0.000    -1.125249   -.9840194
   /cuteq2_2 |   .4750151     .03439    13.81   0.000     .4076119    .5424184
     /depend |      -38.4          .        .       .            .           .
-------------+----------------------------------------------------------------
       theta |          1          .
------------------------------------------------------------------------------
Wald test of equality of coefficients chi2(df = 3)= 514.359 [p-value=0.000]
```

```
LogL for independent ordered probit model -13062.773
initial:       log likelihood = -16915.294
rescale:       log likelihood = -15450.545
rescale eq:    log likelihood = -13207.919
Iteration 0:   log likelihood = -13207.919
Iteration 1:   log likelihood = -13066.981
Iteration 2:   log likelihood = -13062.777
Iteration 3:   log likelihood = -13062.773
Iteration 4:   log likelihood = -13062.773
Generalized bivariate ordinal regression model (copula: joe, mixture: none)
                                                Number of obs     =      5,482
                                                Wald chi2(6)      =     905.06
Log likelihood = -13062.773                     Prob > chi2       =     0.0000
```

	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
finnow						
female	-.1548403	.0296463	-5.22	0.000	-.2129459	-.0967347
homeowner	.5238116	.0303864	17.24	0.000	.4642554	.5833677
unempsick	-.7195785	.0399312	-18.02	0.000	-.7978422	-.6413147
finfut						
female	-.0465534	.0313303	-1.49	0.137	-.1079598	.0148529
homeowner	-.2102062	.0320046	-6.57	0.000	-.272934	-.1474784
unempsick	-.1461061	.0419857	-3.48	0.001	-.2283965	-.0638157
/cuteq1_1	-1.592221	.0394137	-40.40	0.000	-1.669471	-1.514972
/cuteq1_2	-.9077398	.034304	-26.46	0.000	-.9749744	-.8405052
/cuteq1_3	.0812235	.0326666	2.49	0.013	.0171981	.1452488
/cuteq1_4	1.056424	.0348779	30.29	0.000	.9880641	1.124783
/cuteq2_1	-1.054634	.0360286	-29.27	0.000	-1.125249	-.9840194
/cuteq2_2	.4750151	.03439	13.81	0.000	.4076119	.5424184
/depend	-38.4	.	.	.	.	.
theta	1	.				

```
Wald test of equality of coefficients chi2(df = 3)= 514.359 [p-value=0.000]
```

```
LogL for independent ordered probit model -13062.773
initial:       log likelihood = -13062.773
rescale:       log likelihood = -13062.773
rescale eq:    log likelihood = -13062.773
Iteration 0:   log likelihood = -13062.773
Iteration 1:   log likelihood = -13062.773
Generalized bivariate ordinal regression model (copula: indep, mixture: none)
                                                Number of obs     =      5,482
                                                Wald chi2(6)      =     905.06
Log likelihood = -13062.773                     Prob > chi2       =     0.0000
```

	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
finnow						
female	-.1548403	.0296463	-5.22	0.000	-.2129459	-.0967347
homeowner	.5238116	.0303864	17.24	0.000	.4642554	.5833677
unempsick	-.7195785	.0399312	-18.02	0.000	-.7978422	-.6413147
finfut						
female	-.0465534	.0313303	-1.49	0.137	-.1079598	.0148529
homeowner	-.2102062	.0320046	-6.57	0.000	-.272934	-.1474784
unempsick	-.1461061	.0419857	-3.48	0.001	-.2283965	-.0638157
/cuteq1_1	-1.592221	.0394137	-40.40	0.000	-1.669471	-1.514972
/cuteq1_2	-.9077398	.034304	-26.46	0.000	-.9749744	-.8405052
/cuteq1_3	.0812235	.0326666	2.49	0.013	.0171981	.1452488
/cuteq1_4	1.056424	.0348779	30.29	0.000	.9880641	1.124783
/cuteq2_1	-1.054634	.0360286	-29.27	0.000	-1.125249	-.9840194
/cuteq2_2	.4750151	.03439	13.81	0.000	.4076119	.5424184

```
Wald test of equality of coefficients chi2(df = 3)= 514.359 [p-value=0.000]
. estimates stats _all
Akaike´s information criterion and Bayesian information criterion
```

Model	Obs	ll(null)	ll(model)	df	AIC	BIC
gaussian	5,482	.	-13062.15	13	26150.29	26236.21
frank	5,482	.	-13062.44	13	26150.88	26236.8
clayton	5,482	.	-13051.92	13	26129.85	26215.77
gumbel	5,482	.	-13062.77	12	26149.55	26228.86
joe	5,482	.	-13062.77	12	26149.55	26228.86
indep	5,482	.	-13062.77	12	26149.55	26228.86

```
Note: N=Obs used in calculating BIC; see [R] BIC note.
```

Now using the preferred Clayton copula, we allow for the same nonnormal distribution in both residuals, using the `mixture(equal)` option, and we check for local optima by running the optimizer from 10 randomly perturbed starting points. We generate these random points over a region with $\ln\theta \in [-3,1]; \ln\{\pi_u/(1-\pi_u)\} \in [-2,2]; \mu_{u2} \in [-1,1]; \sigma^2_{u2} \in [0,2]$.

```
. quietly bicop finnow finfut `xvars´, copula(`bestcop´) mixture(equal)
> iterate(25)
. local k=e(k)-3    //    position of /depend in parameter vector
. local k1=`k´+1    //    position of /pu1
. local k2=`k´+2    //    position of /mu2
. local k3=`k´+3    //    position of /su2
. local nstarts=10  //    no. of random starts
. local nits=7      //    no. iterations from each start
. set seed 22246
. matrix bequal=e(b)
. matrix maxpar=bequal
. local maxll=e(ll)
. matrix ttt=bequal
. forvalues r=1/`nstarts´ {
  2.  quietly {
  3.   matrix ttt[1,`k´]=4*runiform()-3       // start value for /depend
  4.   matrix ttt[1,`k1´]=4*(runiform()-0.5)  // start value for /pu1
  5.   matrix ttt[1,`k2´]=2*(runiform()-0.5)  // start value for /mu2
  6.   matrix ttt[1,`k3´]=2*runiform()        // start value for /su2
  7.   capture bicop finnow finfut `xvars´, copula(`bestcop´) mixture(equal)
>         from(ttt) log iterate(`nits´) search(off)
  8.   local retcode=_rc
  9.   if e(ll)>`maxll´&`retcode´==0 {
 10.    matrix maxpar=e(b)
 11.    local maxll=e(ll)
 12.   }
 13.   noisily display "Replication... " `r´ ":  logL = " e(ll) "  best so far =
> " `maxll´
 14.  }
 15. }
Replication... 1:  logL = -13047.235  best so far = -13047.235
Replication... 2:  logL = -769989.06  best so far = -13047.235
Replication... 3:  logL = -13047.243  best so far = -13047.235
Replication... 4:  logL = -13054.104  best so far = -13047.235
Replication... 5:  logL = -13047.781  best so far = -13047.235
Replication... 6:  logL = -13047.235  best so far = -13047.235
Replication... 7:  logL = -13048.043  best so far = -13047.235
Replication... 8:  logL = -13048.43  best so far = -13047.235
Replication... 9:  logL = -769989.06  best so far = -13047.235
Replication... 10:  logL = -13047.484  best so far = -13047.235
```

```
. bicop finnow finfut `xvars´, copula(`bestcop´) mixture(equal) from(maxpar)
> iterate(50)
LogL for independent ordered probit model -13062.773
initial:       log likelihood = -13047.235
rescale:       log likelihood = -13047.235
rescale eq:    log likelihood = -13047.235
Iteration 0:   log likelihood = -13047.235
Iteration 1:   log likelihood = -13047.235
Generalized bivariate ordinal regression model (copula: clayton, mixture: equal)
                                                Number of obs     =      5,482
                                                Wald chi2(6)      =     881.17
Log likelihood = -13047.235                     Prob > chi2       =     0.0000

------------------------------------------------------------------------------
             |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
finnow       |
      female |  -.1684891   .0294299    -5.73   0.000    -.2261707   -.1108075
   homeowner |   .5239865   .0302579    17.32   0.000     .4646821    .5832909
   unempsick |  -.7059392   .0403595   -17.49   0.000    -.7850423   -.6268362
-------------+----------------------------------------------------------------
finfut       |
      female |  -.0594949   .0304459    -1.95   0.051    -.1191677    .0001779
   homeowner |  -.2108235    .031232    -6.75   0.000    -.2720372   -.1496098
   unempsick |  -.1182963   .0416947    -2.84   0.005    -.2000165   -.0365761
-------------+----------------------------------------------------------------
   /cuteq1_1 |  -1.645347   .0442619   -37.17   0.000    -1.732098   -1.558595
   /cuteq1_2 |  -.9100796   .0364174   -24.99   0.000    -.9814565   -.8387027
   /cuteq1_3 |   .1139415   .0346303     3.29   0.001     .0460674    .1818155
   /cuteq1_4 |   1.038169   .0373606    27.79   0.000      .964944    1.111395
   /cuteq2_1 |  -1.056057   .0376342   -28.06   0.000    -1.129819   -.9822956
   /cuteq2_2 |   .4788605    .036865    12.99   0.000     .4066064    .5511145
     /depend |  -2.508975   .2246021   -11.17   0.000    -2.949187   -2.068763
        /pu1 |    1.71723   .7078914     2.43   0.015     .3297886    3.104672
        /mu2 |   .4726607   .1257883     3.76   0.000     .2261201    .7192013
        /su2 |   .5318347   .1646931     3.23   0.001     .2090421    .8546273
-------------+----------------------------------------------------------------
       theta |   .0813516   .0182717
      pi_u_1 |   .8477717   .0913568
      pi_u_2 |   .1522283   .0913568
    mean_u_1 |  -.0848723    .059337
    mean_u_2 |   .4726607   .1257883
     var_u_1 |   1.081455   .0422284
     var_u_2 |   .2828481    .175179
------------------------------------------------------------------------------
Wald test of equality of coefficients chi2(df = 3)= 559.003 [p-value=0.000]
Wald test of independence chi2(df = 1)= 19.823 [p-value=0.000]
. matrix bequal=e(b)
. estimates store clayton_equ
. estimates stats clayton clayton_equ
Akaike´s information criterion and Bayesian information criterion

-----------------------------------------------------------------------------
       Model |        Obs  ll(null)  ll(model)      df         AIC        BIC
-------------+---------------------------------------------------------------
     clayton |      5,482         .  -13051.92      13    26129.85   26215.77
 clayton_equ |      5,482         .  -13047.23      16    26126.47   26232.22
-----------------------------------------------------------------------------
               Note: N=Obs used in calculating BIC; see [R] BIC note.
```

The estimated residual distribution is a mixture of a dominant component (`pi_u_1=0.85`) centered close to zero (`mean_u_1=-0.08`), with a secondary (`pi_u_2=0.15`), less dispersed (`var_u_2=0.28`) component centered above zero (`mean_u_2=0.47`).

However, the evidence for nonnormality in the marginal residual distributions is not strong. The Akaike information criterion (AIC) favors the model with equal mixture marginals over the model with normal marginals, while the Bayesian information criterion (BIC), which penalizes model complexity more heavily, gives the opposite result. The following code shows a procedure for plotting the fitted mixture density in comparison with the standard $N(0,1)$ density. To do this, we recover the transformed parameters composing θ and all the mixing parameters from the matrix returned in `e(extpar)`. The resulting plot is shown in figure 1, which reveals a negatively skewed mixture distribution.

```
. matrix mixparams=e(extpar)
. matrix list mixparams
mixparams[1,7]
         theta      pi_u_1      pi_u_2    mean_u_1    mean_u_2     var_u_1
r1   .08135157   .84777173   .15222827  -.08487228    .4726607   1.0814547

       var_u_2
r1   .28284813
. matrix pu1=mixparams[1,"pi_u_1"]
. scalar pu1 = pu1[1,1]
. matrix pu2=mixparams[1,"pi_u_2"]
. scalar pu2 = pu2[1,1]
. matrix mu1=mixparams[1,"mean_u_1"]
. scalar mu1 = mu1[1,1]
. matrix mu2=mixparams[1,"mean_u_2"]
. scalar mu2 = mu2[1,1]
. matrix su1=mixparams[1,"var_u_1"]
. scalar su1 = sqrt(su1[1,1])
. matrix su2=mixparams[1,"var_u_2"]
. scalar su2 = sqrt(su2[1,1])
. twoway (function pu1*normalden(x,mu1,su1)+pu2*normalden(x,mu2,su2),
> range(-3 3) lpattern(solid) lcolor(black)) (function normalden(x),
> range(-3 3) lpattern(longdash) lcolor(black)),
> graphregion(fcolor(white) ilcolor(white) icolor(white) lcolor(white)
> ifcolor(white)) legend(col(2) label(1 "Mixture") label(2 "N(0,1)"))
> xtitle(" ") xscale(titlegap(2)) xlabel(-3(1)3)
```

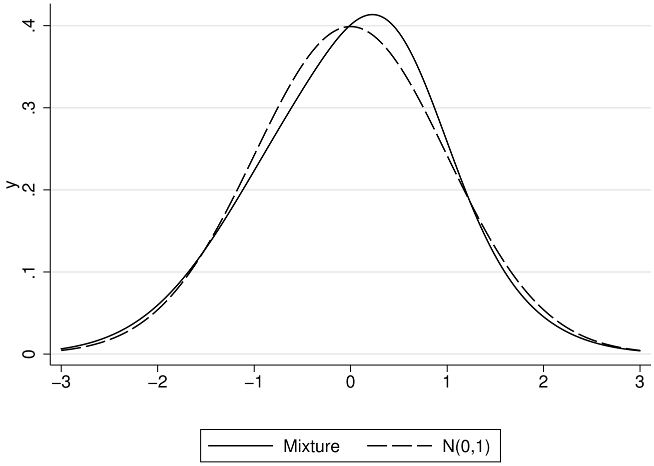

Figure 1. Estimated normal mixture density for the Clayton model residuals

We now allow for different distributional forms in the two residuals by using the option `mixture(both)` and again using multiple starting values. Here convergence is not achieved by using the default initial values but by restarting the optimization from random points, although the estimated mixture is poorly determined. A likelihood-ratio test against the equal-marginals specification gives a marginal result (Pr = 0.0871), and there is conflict between the AIC and the BIC, with the AIC favoring these estimates and the BIC favoring the equal-mixtures model.

```
. local k4=`k´+4    //    position of /pv1
. local k5=`k´+5    //    position of /mv2
. local k6=`k´+6    //    position of /sv2
. matrix a=bequal[1,`k1´..`k3´]  //  initial values for mixing parameters for V
. matrix colnames a=  pv1:_cons mv2:_cons sv2:_cons
. matrix b0=bequal,a
. quietly bicop finnow finfut `xvars´, copula(`bestcop´) mixture(both)
> iterate(25)
. quietly matrix maxpar=e(b)
. quietly local maxll=e(ll)
. set seed 22246
. matrix ttt=b0
. forvalues r=1/`nstarts´ {
  2.  quietly {
  3.   matrix ttt[1,`k´]=4*runiform()-3       // start value for /depend
  4.   matrix ttt[1,`k1´]=4*(runiform()-0.5)  // start value for /pu1
  5.   matrix ttt[1,`k2´]=2*(runiform()-0.5)  // start value for /mu2
  6.   matrix ttt[1,`k3´]=2*runiform()        // start value for /su2
  7.   matrix ttt[1,`k4´]=4*(runiform()-0.5)  // start value for /pv1
  8.   matrix ttt[1,`k5´]=2*(runiform()-0.5)  // start value for /mv2
  9.   matrix ttt[1,`k6´]=2*runiform()        // start value for /sv2
```

```
 10.   capture bicop finnow finfut `xvars´, copula(`bestcop´) mixture(both)
>          from(ttt) log iterate(`nits´) search(off)
 11.   local retcode=_rc
 12.   if e(ll)>`maxll´&`retcode´==0 {
 13.    matrix maxpar=e(b)
 14.    local maxll=e(ll)
 15.   }
 16.   noisily display "Replication... " `r´ ":  logL = " e(ll) "  best so far =
>          " `maxll´
 17.  }
 18. }
Replication... 1:  logL = -769989.06  best so far = -13043.952
Replication... 2:  logL = -769989.06  best so far = -13043.952
Replication... 3:  logL = -298620.15  best so far = -13043.952
Replication... 4:  logL = -13044.151  best so far = -13043.952
Replication... 5:  logL = -769989.06  best so far = -13043.952
Replication... 6:  logL = -769989.06  best so far = -13043.952
Replication... 7:  logL = -769989.06  best so far = -13043.952
Replication... 8:  logL = -769989.06  best so far = -13043.952
Replication... 9:  logL = -13046.226  best so far = -13043.952
Replication... 10:  logL = -769989.06  best so far = -13043.952
. bicop finnow finfut `xvars´, copula(`bestcop´) mixture(both) from(maxpar)
> iterate(50) search(off)
LogL for independent ordered probit model -13062.773
```

(output omitted)

```
Generalized bivariate ordinal regression model (copula: clayton, mixture: both)
                                                Number of obs     =      5,482
                                                Wald chi2(6)      =     866.37
Log likelihood = -13043.952                     Prob > chi2       =     0.0000
```

	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
finnow						
female	-.1659407	.0297574	-5.58	0.000	-.2242643	-.1076172
homeowner	.5343804	.0308191	17.34	0.000	.473976	.5947847
unempsick	-.7092343	.0404555	-17.53	0.000	-.7885255	-.629943
finfut						
female	-.008524	2.094358	-0.00	0.997	-4.11339	4.096342
homeowner	-.0196506	4.821656	-0.00	0.997	-9.469922	9.430621
unempsick	-.0031946	.7761486	-0.00	0.997	-1.524418	1.518029
/cuteq1_1	-1.643005	.045037	-36.48	0.000	-1.731276	-1.554734
/cuteq1_2	-.9138487	.0375694	-24.32	0.000	-.9874833	-.8402141
/cuteq1_3	.1137748	.0354499	3.21	0.001	.0442942	.1832554
/cuteq1_4	1.060686	.0348293	30.45	0.000	.9924221	1.12895
/cuteq2_1	.2728202	33.24207	0.01	0.993	-64.88043	65.42607
/cuteq2_2	.4271055	4.731994	0.09	0.928	-8.847433	9.701643
/depend	-2.504887	.2234805	-11.21	0.000	-2.9429	-2.066873
/pu1	-2.228777	1.243522	-1.79	0.073	-4.666035	.2084806
/mu2	.1549464	.1383337	1.12	0.263	-.1161827	.4260755
/su2	.9007953	.0681421	13.22	0.000	.7672393	1.034351
/pv1	-1.702438	.590641	-2.88	0.004	-2.860073	-.5448025
/mv2	.4089723	.6733854	0.61	0.544	-.9108388	1.728783
/sv2	.068307	16.76491	0.00	0.997	-32.79031	32.92692

```
-------------+----------------------------------------------------------------
       theta |   .0816849    .018255
      pi_u_1 |   .0971959   .1091176
      pi_u_2 |   .9028041   .1091176
    mean_u_1 |   -1.43922   .5266053
    mean_u_2 |   .1549464   .1383337
     var_u_1 |   .4571569     2.1114
     var_u_2 |   .8114322   .1227641
      pi_v_1 |   .1541472   .0770112
      pi_v_2 |   .8458528   .0770112
    mean_v_1 |  -2.244156   3.605769
    mean_v_2 |   .4089723   .6733854
     var_v_1 |   .5076687   17.47826
     var_v_2 |   .0046658    2.29032
------------------------------------------------------------------------------
Wald test of equality of coefficients chi2(df = 3)= 560.139 [p-value=0.000]
Wald test of independence chi2(df = 1)= 20.023 [p-value=0.000]

. matrix bunequal=e(b)

. estimates store clayton_both

. estimates stats clayton_equ clayton_both

Akaike´s information criterion and Bayesian information criterion

-----------------------------------------------------------------------------
       Model |        Obs  ll(null)  ll(model)      df         AIC        BIC
-------------+---------------------------------------------------------------
 clayton_equ |      5,482         .  -13047.23      16    26126.47   26232.22
clayton_both |      5,482         .  -13043.95      19     26125.9   26251.48
-----------------------------------------------------------------------------
               Note: N=Obs used in calculating BIC; see [R] BIC note.

. lrtest clayton_equ clayton_both

Likelihood-ratio test                                 LR chi2(3)  =      6.57
(Assumption: clayton_equ nested in clayton_both)      Prob > chi2 =    0.0871
```

Next, to demonstrate the second form of the `bicop` syntax, we revert to the option `mixture(equal)` and refit the model with the marginally insignificant gender effect dropped from equation 2. Except for the scaling of the coefficients in equation 2, the results change little. Again the AIC and BIC are in conflict over whether this is the best-fitting model.

```
. local xvars1 female homeowner unempsick

. local xvars2 homeowner unempsick

. bicop (finnow=`xvars1´) (finfut=`xvars2´), copula(`bestcop´) mixture(equal)
> from(bequal, skip) iterate(50) search(off)
LogL for independent ordered probit model -13063.877

Iteration 0:   log likelihood = -13052.337
Iteration 1:   log likelihood = -13049.144
Iteration 2:   log likelihood = -13049.134
Iteration 3:   log likelihood = -13049.134

Generalized bivariate ordinal regression model (copula: clayton, mixture: equal)
                                                Number of obs     =      5,482
                                                Wald chi2(5)      =     881.41
Log likelihood = -13049.134                     Prob > chi2       =     0.0000

------------------------------------------------------------------------------
             |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
finnow       |
      female |  -.1651425   .0294366    -5.61   0.000    -.2228371   -.1074479
   homeowner |   .5245214   .0303039    17.31   0.000     .4651268     .583916
   unempsick |  -.7072024   .0402962   -17.55   0.000    -.7861816   -.6282232
-------------+----------------------------------------------------------------
finfut       |
   homeowner |  -.2079559   .0312757    -6.65   0.000    -.2692553   -.1466566
   unempsick |  -.1139064   .0417335    -2.73   0.006    -.1957025   -.0321102
-------------+----------------------------------------------------------------
   /cuteq1_1 |  -1.640419   .0441888   -37.12   0.000    -1.727027    -1.55381
   /cuteq1_2 |   -.908388    .036365   -24.98   0.000    -.9796621   -.8371138
   /cuteq1_3 |   .1135852   .0346939     3.27   0.001     .0455864    .1815841
   /cuteq1_4 |    1.04191   .0374511    27.82   0.000      .968507    1.115313
   /cuteq2_1 |  -1.016769   .0319153   -31.86   0.000    -1.079322   -.9542159
   /cuteq2_2 |   .5183207   .0309565    16.74   0.000     .4576471    .5789943
     /depend |  -2.520257   .2268808   -11.11   0.000    -2.964936   -2.075579
        /pu1 |   1.760666   .7653936     2.30   0.021      .260522     3.26081
        /mu2 |   .4740137   .1367034     3.47   0.001     .2060798    .7419475
        /su2 |   .5423894   .1763895     3.07   0.002     .1966723    .8881065
-------------+----------------------------------------------------------------
       theta |   .0804389     .01825
      pi_u_1 |    .853293   .0958151
      pi_u_2 |    .146707   .0958151
    mean_u_1 |  -.0814973   .0611763
    mean_u_2 |   .4740137   .1367034
     var_u_1 |   1.076078   .0428315
     var_u_2 |   .2941863   .1913436
------------------------------------------------------------------------------
Wald test of equality of coefficients chi2(df = 2)= 552.485 [p-value=0.000]
Wald test of independence chi2(df = 1)= 19.427 [p-value=0.000]

. estat ic

Akaike´s information criterion and Bayesian information criterion

-----------------------------------------------------------------------------
       Model |        Obs  ll(null)  ll(model)      df         AIC        BIC
-------------+---------------------------------------------------------------
           . |      5,482         .  -13049.13      15    26128.27   26227.41
-----------------------------------------------------------------------------
               Note: N=Obs used in calculating BIC; see [R] BIC note.
```

To show the differences in results that can follow from using `bicop` rather than `bioprobit`, we now use the `predict` command to construct predictions for expectations of the change in FWB conditional on current reported FWB. These are sample means of estimates of $\Pr(Y_2 = s|Y_1 = r, X_i)$. The following code computes the predictions for the Gaussian model and the equal-mixtures Clayton specification for $s = 1$ (expected worsening of FWB) and $s = 3$ (expected improvement) and all $r = 1, \ldots, 5$, summarizing the relationship by plotting them against r.

```
. generate tee=_n if _n<=5
(5,477 missing values generated)

. foreach c in clayton_equ gaussian {
  2.   generate up`c´=.
  3.   generate down`c´=.
  4.   forvalues t=1/5 {
  5.     quietly {
  6.       estimates restore `c´
  7.       capture drop tmp*
  8.       predict tmp if e(sample), pcond2 outcome(`t´,3)
  9.       predict tmp1 if e(sample), pcond2 outcome(`t´,1)
 10.       summarize tmp, meanonly
 11.       replace up`c´=r(mean) if tee==`t´
 12.       summarize tmp1, meanonly
 13.       replace down`c´=r(mean) if tee==`t´
 14.         }
 15.   }
 16. }
(5,482 missing values generated)
(5,482 missing values generated)
(5,482 missing values generated)
(5,482 missing values generated)

. drop tmp*

. line upgaussian upclayton tee, graphregion(fcolor(white) ilcolor(white)
> icolor(white) lcolor(white) ifcolor(white)) msymbol(none) xtick(1(1)5)
> xtitle("Current financial wellbeing") xscale(titlegap(2)) xlabel(1(1)5)
> ytitle("Pr(better)") yscale(titlegap(5)) lpattern(solid longdash)
> lcolor(black black)
> legend(col(2) label(1 "Bivariate ordered probit") label(2 "Generalized model"))

. line downgaussian downclayton tee, graphregion(fcolor(white) ilcolor(white)
> icolor(white) lcolor(white) ifcolor(white)) msymbol(none) xtick(1(1)5)
> xtitle("Current financial wellbeing") xscale(titlegap(2)) xlabel(1(1)5)
> ytitle("Pr(worse)") yscale(titlegap(5)) lpattern(solid longdash)
> lcolor(black black) legend(col(2) label(1 "Bivariate ordered probit")
> label(2 "Generalized model"))
```

Figures 2 and 3 show these plots. The most striking feature is that the generalized `bicop` model suggests considerably more pessimistic expectations conditional on a low current level of FWB, particularly for the expectation of further worsening. Note that the data come from a period of government austerity targeted particularly on welfare recipients following a deep recession, so these pessimistic predictions are not implausible.

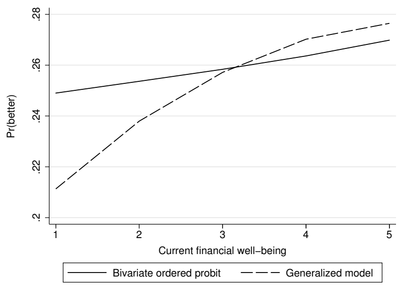

Figure 2. Predicted probability of expectation of better FWB conditional on current FWB

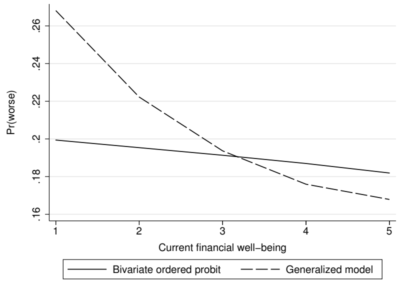

Figure 3. Predicted probability of expectation of worse FWB conditional on current FWB

The source of the difference is the different patterns of dependence built into the Clayton and Gaussian copulas: the former model implies strong positive dependence in only the left tail (low actual and anticipated FWB), whereas the latter implies uniform dependence. Although the Clayton model used to generate the plot also allows for a departure from normality in each residual, in this particular application, the form of the marginals makes much less difference to the properties of the fitted model than the choice of copula.

7 Acknowledgments

We thank the editor and an anonymous referee for helpful comments and suggestions. This work was supported by the Medical Research Council under grant MR/L022575/1. It uses data from the Understanding Society survey administered by ISER, University of Essex, funded by the Economic and Social Research Council. Pudney acknowledges further ESRC funding through the UK Centre for Longitudinal Studies and the Centre for Micro-Social Change (grants RES-586-47-0002 and RES-518-28-5001). The views expressed in this article, and any errors or omissions, are those of the authors only.

8 References

Andrews, D. W. K. 2001. Testing when a parameter is on the boundary of the maintained hypothesis. *Econometrica* 69: 683–734.

Chernoff, H. 1954. On the distribution of the likelihood ratio. *Annals of Mathematical Statistics* 25: 573–578.

Hernández-Alava, M., and S. Pudney. 2015. bicop: A Stata command for fitting bivariate ordinal regressions with residual dependence characterised by a copula function and normal mixture marginals. Understanding Society Working Paper No. 2015-02, Economic & Social Research Council. http://www.understandingsociety.ac.uk/research/publications/working-paper/understanding-society/2015-02.pdf.

Knies, G., ed. 2015. *Understanding Society: The UK Household Longitudinal Study Waves 1–5 User Manual. Version 1.1.* Colchester, UK: Economic & Social Research Council. https://www.understandingsociety.ac.uk/d/218/6614_UserManual_Wave1to5_v1.1.pdf?1446134584.

McLachlan, G., and D. Peel. 2000. *Finite Mixture Models*. New York: Wiley.

Pudney, S. 2011. Perception and retrospection: The dynamic consistency of responses to survey questions on wellbeing. *Journal of Public Economics* 95: 300–310.

Sajaia, Z. 2008. bioprobit: Stata module for bivariate ordered probit regression. Statistical Software Components S456920, Department of Economics, Boston College. https://ideas.repec.org/c/boc/bocode/s456920.html.

Titterington, D. M., A. F. M. Smith, and U. E. Makov. 1985. *Statistical Analysis of Finite Mixture Distributions*. New York: Wiley.

Trivedi, P. K., and D. M. Zimmer. 2005. Copula modeling: An introduction for practitioners. *Foundations and Trends in Econometrics* 1: 1–111.

About the authors

Mónica Hernández-Alava is an applied microeconometrician in the health economics and decision science section in ScHARR at the University of Sheffield in Sheffield, UK.

Steve Pudney is a professor of economics at ISER at the University of Essex in Essex, UK.