ggt Documentation

Syntax

```
ggt, outcomevar(varname) orgchoice(varname) indID(varname) orgID(varname)
choicechar(varlist) [options]
```

Description

This program estimates the parameters of the Geweke, Gowrisankaran, and Town (2003) "GGT" model. The GGT model estimates the posterior distribution of organizational performance where there are many organizations from which individuals can choose to receive services. In this framework, individuals may select organizations based, in part, on information that is unobserved to the researcher and is correlated with the binary outcome. If this is the case, then standard approaches to inferring organization performance will yield biased estimates. The GGT model corrects for this unobserved selection allowing for flexible correlation in the error structure across the organizational choice and outcome equations. The estimation approach is Bayesian. **In sum, the model combines an organization choice multinomial probit model with an individual outcome binary probit model, allowing for correlation across equations for each individual.**[1]

The parameters are estimated using Bayesian inference through Markov chain Monte Carlo techniques to simulate parameters and latent variables conditional on data to determine the posterior distribution of parameters. While we present the basics of the model below, we encourage all users of this Stata program to read the GGT paper to fully understand the model, assumptions underneath the model, and parameters used in the estimation.

Required Files

To speed up the computing process, the program code calls an included backend C plugin file. Thus, in addition to the Stata .ado files, the user additionally needs a .plugin file. These will be automatically downloaded with the ggt installation. The files necessary for the code to run are the following:

- ggt.ado
- callCcode.ado
- bayesqual15.plugin

Methods and Equations

The model presented below comes from the GGT application in which the authors estimate hospital quality measures. The authors assume that patient mortality is a function of individual risk factors and hospital quality. However, GGT make the important note that patients may "select" into which hospital to receive care, which would bias the hospital quality estimate if not accurately controlled for. Thus, GGT define and estimate a model that allows for unobserved patient characteristics to be correlated with both hospital choice and patient mortality.

We include a brief explanation of the model here to show which variables and parameters are referenced in the calling of the `ggt` Stata program.

The binary individual outcome equation: $m_i^* = c_i'\beta + x_i'\gamma + \varepsilon_i$ (equation (1) in GGT)

Here, m_i^* is the latent outcome variable for the observed binary variable m_i. This is patient mortality in the GGT application. The outcome variable depends on individual characteristics, x_i, and which organization the individual chooses, c_i (hospital choice in GGT).

The organization choice model is: $c_i^* = Z_i\alpha + \eta_i$ (equation (3) in GGT)

[1] As noted in GGT, some possible applications for this model include: hospital quality based on mortality, school performance based on graduation rates, prison rehabilitation programs based on recidivism rates, and job training programs based on incidence of harassment complaints.

Here, c_i* is the latent choice vector for the observed choice vector c_i where c_{ij}=1 if patient i chose organization j and 0 otherwise. The individual choice is allowed to depend on individual-organization characteristic matrix, Z_i, such as distance to hospital in GGT.

Selection is modeled as follows: $\varepsilon_i = \eta_i'\delta + \xi_i$ (equation (5) in GGT)

Here, GGT allow the error term in the organization choice equation to be correlated with the error term in the binary outcome equation via the parameter, δ.

The `ggt` Stata program estimates the latent variables and parameters (m_i* c_i*, α, β, γ, δ) through MCMC methods. Every 100th draw is recorded in a csv file and saved wherever the user specifies in the `ggt` option, `savedraws()`.

Because the process uses Bayesian inference, the estimation of the model depends not only on the necessary model variables, but also the prior distributions for each parameter. See section 2.2. of GGT for more information on prior distributions.

- Following GGT, the estimation method assumes independent prior distributions for α, γ, and δ. For these parameters, we assume mean 0 for each prior and allow user options for prior variances.
- For the parameter, β, we again follow GGT and use hyperpriors to allow correlation between organizations based on their fixed characteristics. β can be written as the sum of organization dummies and organization category dummies. For example, in GGT, there are four hospital ownership categories, k={1,2,3,4}, four hospital size categories, l={1,2,3,4}, and 144 unique hospitals, j. Thus, $\beta_j=p_k+s_l+u_j$ where p_k=1 if hospital j is in ownership category k, s_l=1 if hospital j is in size category l, and u_j=1 for hospital j. We assume p, s, u are jointly Normal with mean, 0, and mutually independent, but allow dependence within each organization characteristic through definition of hyper-prior distributions. Specifically, we assume p, s, and u have variance τ_p^2, τ_s^2, and τ_u^2 respectively, with a hyper-prior distribution defined as $\underline{s}^2 / \tau_{p,s,u}^2 \sim \chi^2(\underline{v})$, allowing user options for $\underline{s}^2$ and $\underline{v}$.

Technical Notes: Users may notice some slight differences in the above description of prior distributions from that in GGT Section 2.2. These do not change the model but do make the Stata code more tractable. We describe these changes below.

- We remove the constant term, β_1, from the linear equation defining β_j. Instead, we combine this constant term with constant term in γ. Thus, users should not specify a different prior variance for β_1 and should keep in mind that this term will be included γ.
- Within each model parameter vector, we request users to specify a single value for prior variances. For example, suppose γ is the coefficient for two variables, risk score and age. This Stata code allows users to specify σ_γ^2 in the prior distribution: $\begin{pmatrix}\gamma_1\\ \gamma_2\end{pmatrix} \sim N\left[\begin{pmatrix}0\\0\end{pmatrix}, \begin{pmatrix}\sigma_\gamma^2 & 0\\ 0 & \sigma_\gamma^2\end{pmatrix}\right]$. Since we do not allow σ_γ^2 to differ by elements of γ, users should modify/rescale variables to fit into this framework.
- We do not allow non-zero elements in the off-diagonal prior variance specifications.

The remainder of this documentation explains the different function options and presents three example program runs.

Options

Required Model Variables

outcomevar(*varname*) is required. It is the name of the variable that indicates the individual outcome in the binary probit model. This variable should be either 0 or 1, and constant within each individual.

orgchoice(*varname*) is required. It is the name of the variable that indicates which organization the individual selects/chooses. This variable should equal 1 for the chosen organization and 0 otherwise, with each individual choosing exactly one organization.

indID(*varname*) is required. It provides a unique identifier for each individual.

orgID(*varname*) is required. It provides a unique identifier for each organization.

choicechar(*varlist*) is required. It specifies the numeric variables that should be included in the choice equation. These are the Z variables in the Methods and Equations section above. Note that the choice model is time-intensive; therefore, it is recommended to limit the number of specified choice variables to fewer than 10.

Optional Model Variables

orgchar(*varlist*) specifies the organization characteristic variables. These are the k and l variables in the Methods and Equations section above. The maximum number of variables in this varlist is 10. The variables must be categorical in nature and must be constant within organization.

indchar(*varlist*) specifies the name of the variables that should be included in the individual outcome probit equation. These are the X variables in the Methods and Equations section above. The maximum number of variables in this varlist is 100. The variables must be numeric and consistent within individual.

Optional Model Specifications

niter(*integer*) is the number of iterations for Gibbs sampling. The default is 100000. Must be a multiple of 100.

alphapriorvar(*real*) is the diagonal of the α prior variance-covariance matrix. The default is 1.

gammapriorvar(*real*) is the diagonal of the γ prior variance-covariance matrix. The default is 1.

deltapriorvar(*real*) is the $\sigma_\delta{}^2$ term in the prior distribution: $\delta \sim N\left(0, (\sigma_\delta{}^2 \Sigma)^{-1}\right)$ where, as in GGT, $\Sigma = I_{J-1} + e_{J-1}e'_{J-1}$ for identity matrix I, number of organizations, J, and vector of units, e. See footnote 17 in GGT for information on choosing $\sigma_\delta{}^2$. The default is 0.038416.

priortau(*real*, *integer*) are the hyper-parameters for the organization characteristic variances. From the Methods and Equations section above, `ggt` allows users to specify $\underline{s}^2$ and $\underline{\nu}$ in the hierarchical prior, $\underline{s}^2 / \tau_o{}^2 \sim \chi^2(\underline{\nu})$ for organization characteristic, o. The first number in priortau refers $\underline{s}^2$, and the second number refers to $\underline{\nu}$. The default is priortau(1.25,5). Users must specify both elements if choosing to use this option.

noselection – This option should be specified if the user does not want to apply the selection correction. In this case, the program will estimate the model restricting the individual outcome error and choice model error to be 0 (i.e., will estimate the model with $\delta = 0$). This is equivalent to independent estimations of the multinomial probit choice model and binomial probit outcome model.

noconstant- This option should be specified if the user does not want to include a constant in the outcome probit equation, i.e. γ will not include a constant term.

Reporting

savedraws(*string)*- This option allows the user to specify the name of the csv file that holds every 100 draws of each parameter via the MCMC Gibbs Sampling routine. The default is "temp_GGT_output.csv".

Remarks and Examples

In this section, we present three examples to demonstrate how to estimate the GGT model. We start with a description of how the data should be structured in order to use `ggt`. An example dataset is included in the installation package.

```
use ggt_test_data.dta, clear
```

For these examples, we are interested in estimating hospital quality. We have data on 300 patients and 8 hospitals. The individual patient variables include the following: the mortality outcome ("mortality"), the hospital choice variable ("hosp_choice"), and an *observed* risk score measure ("risk_score"). Additionally, we have two variables "dist" and "dist2" representing the distance from each patient to each hospital along with its square (normalized to have similar scales, necessary since the program requires prior variances to be the same for each model parameter vector). We also have hospital characteristic variables, "hosp_size" and "hosp_ownership". The categories for hosp_size are "small" and "large", and the categories for hosp_ownership are "public" and "private".

The individual patient ID variable is called "indnumber" and the hospital ID variable is called "hospnum". In the Stata dataset, there should be an observation for each individual-hospital pair, even if the individual did not choose that hospital. For example, with 300 patients and 8 hospitals, we have 300*8=2400 observations in the data. The table below shows the structure of the data for the first 2 patients. You can see that individual 1 went to hospital 7 and died while patient number 2 went to hospital 3 and did not die. The risk score measure is constant within an individual while the

distance and distance^2 measures differ for each patient-hospital pair. Additionally, notice the hospital characteristics are constant within hospitals, e.g., hosp_size and hosp_ownership is always "small" and "public" for the row in which hospnum==1.

indnumber	hospnum	hosp_choice	mortality	risk_score	dist	dist2	hosp_size	hosp_ownership
1	1	0	1	1.549	0.015	0.000	small	public
1	2	0	1	1.549	0.250	0.013	large	public
1	3	0	1	1.549	0.259	0.014	large	public
1	4	0	1	1.549	0.080	0.001	large	private
1	5	0	1	1.549	0.097	0.002	large	public
1	6	0	1	1.549	0.160	0.005	large	public
1	7	1	1	1.549	0.459	0.042	small	private
1	8	0	1	1.549	0.491	0.048	small	public
2	1	0	0	0.723	0.052	0.001	small	public
2	2	0	0	0.723	0.162	0.005	large	public
2	3	1	0	0.723	0.067	0.001	large	public
2	4	0	0	0.723	0.097	0.002	large	private
2	5	0	0	0.723	0.187	0.007	large	public
2	6	0	0	0.723	0.019	0.000	large	public
2	7	0	0	0.723	0.110	0.002	small	private
2	8	0	0	0.723	0.058	0.001	small	public

Example 1:
If we want to estimate the model parameters and recover hospital quality using all the default settings, we would simply type the command:

```
ggt, outcomevar(mortality) orgchoice(hosp_choice) indID(indnumber) orgID(hospnum)
choicechar(dist dist2)
```

This will apply the selection model using dist and dist2 as the choice variables. Since we did not specify the `indchar()` option, the program will assume only a constant and the hospital choice vector within the individual probit equation. Additionally, since we did not specify `orgchar()`, the program will assume no correlation across hospitals via hospital size or ownership. The sampling algorithm will assume the default prior variance options and number of iterations.

Once complete, the output on the screen will show `complete. Success=1.` This indicates that the program ran successfully and parameter draws are saved locally.

Note #1: The example above may take several minutes to complete running. Run times will increase substantially with the size of the data and parameters to estimate. Once the code is complete, the word "complete" will display on the Stata screen. Return value will be 1 if successful and 0 otherwise. If the code does not complete or Stata simply quits, this is likely due to an error with the prior variance specifications which are not compatible with the data. We suggest trying to call the program again using different prior variance values and lower number of iterations. If you receive an unspecified error code, please contact the program authors.

Since we did not use the `savedraws()` option, the default was used, "temp_GGT_output.csv". The csv file has 20 columns, with the first column denoting the iteration number, followed by the values of $\tau, \beta, \gamma, \alpha, \delta$. There are 1001 rows corresponding to a header row followed by every 100th iteration up to the default number of iterations, 100,000. The screenshot below provides a visual of the output file for example 1. Descriptions of column names and example hospital quality calculation follow.

Note #2: The first 10,000 iterations (rows 2 through 101) should be considered "burn-in" for sampling purposes and should be removed before proceeding with additional analysis of the model draws. The screenshot of output below hides these burn-in iterations for the purpose of the example quality calculation that follows.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
1	iter	t_orgID	b_orgID_1	b_orgID_2	b_orgID_3	b_orgID_4	b_orgID_5	b_orgID_6	b_orgID_7	b_orgID_8	g_constant	a_dist	a_dist2	delta1	delta2	delta3	delta4	delta5	delta6	delta7
102	10100	0.1949	-0.56553	-0.44398	-0.38544	-0.05846	-0.28722	-0.32393	0.17738	-0.68213	-0.164002	6.59737	3.40969	-0.29568	0.08193	0.04354	0.39646	0.14957	-0.06935	-0.31393
103	10200	0.25358	-0.12293	0.09252	-0.26405	0.35003	0.17768	-0.03647	0.19037	-0.65946	-0.413918	6.61958	0.7053	0.1267	0.1983	0.07175	-0.02269	-0.24373	-0.0613	-0.26983
104	10300	0.23595	-0.314	0.43805	-0.06106	0.30926	0.06264	-0.01777	0.53459	0.01778	-0.47806	6.60813	1.98411	0.09104	-0.15217	0.1864	0.0031	-0.0601	-0.14127	0.01414
105	10400	0.10086	-0.06129	-0.05428	0.43701	-0.14279	0.12346	0.00658	0.06974	-0.38357	-0.435161	6.27437	0.29333	-0.32951	0.16726	0.02363	0.26687	-0.21474	-0.23105	0.03759
106	10500	0.37981	-0.61478	0.42095	-0.1693	-0.02224	-0.02696	0.25192	0.99272	-0.49193	-0.487631	5.82089	0.75822	0.07765	0.14246	-0.00282	0.21116	0.01176	-0.00836	-0.12354
107	10600	0.17821	-0.35345	0.12781	0.11282	0.2146	0.00506	0.20864	0.41228	-0.20513	-0.560994	6.31223	1.49082	-0.18117	0.01211	0.30656	-0.26538	0.01728	0.12825	-0.03278
108	10700	0.12479	-0.17643	-0.14374	-0.20649	-0.24177	-0.45482	-0.40413	-0.00146	-0.13661	-0.227937	5.87498	1.1133	-0.14728	0.07072	0.17235	0.18586	0.08115	-0.06153	0.02418
109	10800	0.2888	-0.32451	0.17597	-0.18409	0.22376	0.07623	0.38006	-0.00894	-0.72729	-0.346078	5.77203	1.5897	0.06257	0.16122	-0.32126	-0.18213	0.1479	-0.20176	0.10467
110	10900	0.11227	-0.53364	-0.35763	-0.19149	0.17133	0.00955	0.12519	0.33315	-0.20031	-0.401014	5.94981	0.54487	-0.17162	0.36359	0.09527	-0.10035	-0.01262	-0.46755	0.05229

Column Names

- **iter:** denotes the iteration number (intervals of 100), starting at 100 and ending at either the default number of iterations, 100,000, or at the user specified `niter()`. Note that the first 10,000 iterations (rows 2 through 101) should be considered "burn-in" for sampling purposes and subsequently removed before proceeding with additional analysis of the model draws.
- **t_*:** are the draws for τ. Columns with prefix t_* correspond to the organizational characteristics, specified by the user in `orgchar()`. There will always be at least one τ reported, corresponding to the organization ID fixed effect. Since we did not specify additional hospital characteristics `orgchar()`, the output only has one column with prefix t_*, named t_orgID.
- **b_*:** are the draws for β. Columns with prefix b_* correspond to the organizational fixed effect and any organizational characteristics specified by the user in `orgchar()`. There will always be at least one β for each unique organization in the sample. In this example, there were 8 hospitals, with numeric values saved in the variable hospnum, ranging from 1 to 8. Thus, the β corresponding the hospital with hospnum=1 will be in the column named, b_orgID_1. See example 2 for a model with additional hospital characteristics.
- **g_*:** are the draws for γ. Columns with prefix g_* correspond to the coefficient on individual characteristics specified by the user in `indchar()`. In this example, we did not specify any variables in `indchar()`, nor did we specify the `noconstant` option. Therefore, there is only one γ component in the model, corresponding to the constant, and thus stored in the column g_constant. See example 2 for a model that includes additional individual specific characteristics.
- **a_*:** are the draws for α. Columns with prefix a_* correspond to the coefficients on the choice model variables specified by the user in `choicechar()`. In this example, we had two choice variables, dist and dist2, with corresponding output coefficients of a_dist and a_dist2, respectively.
- **delta1-delta*:** are the draws for δ. There are J-1 columns with prefix delta*, where J is the total number of organizations. These are used for quality selection adjustment (see below).

Quality Calculation

Via equation 7 in GGT, the quality value for hospital *j* and iteration draw *d* is adjusted for selection as follows:

$q_j^d = -\beta_j^d / \sqrt{\left(\delta^{d\prime} \Sigma \delta^d + 1\right)}$ where $\Sigma = I_{J-1} + e_{J-1} e'_{J-1}$ for identity matrix I, number of organizations, J, and vector of units, e. We can use the draws in temp_GGT_output.csv to recover the distribution of quality for each hospital.

- As an example, let's take iteration draw *d=10,100* and hospital *j=3*. Here, we take the values from the row corresponding to iter=10,100. β_j^d is the value in column b_orgID_3 and δ^d is the vector, delta1-delta7. In other

words, β_j^d =-0.385 and δ^d =(-0.296, 0.082, 0.044, 0.396, 0.150, -0.069, -0.314). Thus, for this iteration-hospital, we have $q_3^{10,100} = \frac{0.385}{1.174} = 0.328$.

- Applying this procedure to all iterations, (after removing burn-in following note #2 above), we can recover the full distribution of hospital #3 quality. The mean of this distribution can be used as the point estimate for hospital #3 quality. Similarly, we can apply the same procedure to all hospitals and recover the full distribution of quality for each hospital, necessary for across hospital comparisons.

Example 2:

Now suppose we want to include the patient risk score measure in the morality equation, and we also want to allow hospital quality to be correlated based on their size and ownership. Additionally, we want to rescale the prior variances based on the structure of the data. Specifically, we want the prior variance of alpha to be 5, the prior variance of gamma to be 3, the selection term for delta to be 0.1, and the parameters for the hyperpriors to be 1 and 5. Finally, we only want to simulate 50000 draws and we want to name the file that saves the parameter draws "ggt_example2.csv".

To do this, we would type the command:

```
ggt, outcomevar(mortality) orgchoice(hosp_choice) indID(indnumber) orgID(hospnum)
choicechar(dist dist2) indchar(risk_score) orgchar(hosp_size hosp_ownership)
alphapriorvar(5) gammapriorvar(3) deltapriorvar(.1) priortau(1,5) niter(50000)
savedraws("ggt_example2.csv")
```

The output is saved in a csv file named ggt_example2.csv. In this file, there are 501 rows corresponding to a header row followed by every 100th iteration up to the specified number of iterations, 50,000. There are 27 columns, with the following headers: *iter - t_hosp_size - t_hosp_ownership - t_orgID - b_hosp_size_large - b_hosp_size_small - b_hosp_ownership_private - b_hosp_ownership_public - b_orgID_1 - b_orgID_2 - b_orgID_3 - b_orgID_4 - b_orgID_5 - b_orgID_6 - b_orgID_7 - b_orgID_8 - g_constant - g_risk_score - a_dist - a_dist2 - delta1 - delta2 - delta3 - delta4 - delta5 - delta6 - delta7*

Compared to example 1, we now have additional columns with prefix t_* (corresponding to the hyperprior draws for variances of hosp_size and hosp_ownership), b_* (corresponding to the coefficient on indicators for hosp_size==large, hosp_size==small, hosp_ownerhsip==private, and hosp_ownership==public), and g_* (corresponding to the coefficient on observed patient risk score in the data).

Quality Calculation

- Continuing from example 1, let's take iteration 10,100 and hospital #3. Compared to the model in example 1, the model in this example includes hospital characteristics specified in `orgchar()`. Therefore, we must adjust the numerator in the quality calculation accordingly. Since hospital #3 is a large public hospital, then β_j^d is now the value in b_orgID_3 added to b_hosp_size_large and b_hosp_ownership_public. As before, δ^d is the vector, delta1-delta7. The relevant row and columns are shown below. For this model, the quality estimate for hospital #3 and iteration 10,100 is $q_3^{10,100} = \frac{-(0.854-1.213+0.832)}{1.384} = -0.342$
- As before, one should do this for all hospital-iterations (after removing burn-in) to obtain the hospital quality distribution necessary for across hospital quality comparisons.

	A	E	H	K	U	V	W	X	Y	Z	AA
1	**iter**	**b_hosp_size_large**	**b_hosp_ownership_public**	**b_orgID_3**	**delta1**	**delta2**	**delta3**	**delta4**	**delta5**	**delta6**	**delta7**
102	10100	0.85369998	-1.21251	0.832103	-0.02639	0.30974	0.03653	-0.06687	0.04557	-0.21818	-0.65707

Example 3:
Finally, suppose we wish to determine the role of unobserved patient selection by comparing the selection adjusted hospital quality measure in example 2 to a version that does not apply the selection correction. In this case, we should add the `noselection` option to the command above.

Note #3: Even though the equation we wish to estimate does not depend on patient-organization choice characteristics, the program will still require choice characteristics and provide estimates of α. Further, since the `nonselection` model assumes that δ=0, the program will ignore the `deltapriorvar()` option.

```
ggt, outcomevar(mortality) orgchoice(hosp_choice) indID(indnumber) orgID(hospnum)
choicechar(dist dist2) indchar(risk_score) orgchar(hosp_size hosp_ownership)
alphapriorvar(5) gammapriorvar(3) priortau(1,5) niter(50000)
savedraws("ggt_example3.csv") noselection
```

The output is saved in a csv file named ggt_example3.csv. As in example 2, there are 501 rows corresponding to a header row followed by every 100th iteration up to the specified number of iterations, 50,000. There are 27 columns, with the same headers as in example 2: *iter - t_hosp_size - t_hosp_ownership - t_orgID - b_hosp_size_large - b_hosp_size_small - b_hosp_ownership_private - b_hosp_ownership_public - b_orgID_1 - b_orgID_2 - b_orgID_3 - b_orgID_4 - b_orgID_5 - b_orgID_6 - b_orgID_7 - b_orgID_8 - g_constant - g_risk_score - a_dist - a_dist2 - delta1 - delta2 - delta3 - delta4 - delta5 - delta6 - delta7*

However, since the `noselection` option was included in the command, then the values for delta1-delta7 will be 0 for every draw (screenshot of selected rows and columns below).

	A	E	H	K	U	V	W	X	Y	Z	AA
1	**iter**	**b_hosp_size_large**	**b_hosp_ownership_public**	**b_orgID_3**	**delta1**	**delta2**	**delta3**	**delta4**	**delta5**	**delta6**	**delta7**
102	10100	0.77948099	-0.32002601	-0.579086	0	0	0	0	0	0	0
103	10200	0.84243101	-0.379843	-0.38979	0	0	0	0	0	0	0
104	10300	0.14232901	-0.00562378	-0.224818	0	0	0	0	0	0	0
105	10400	-0.37555301	-0.57935703	-0.0316414	0	0	0	0	0	0	0
106	10500	0.15527301	-0.8545	0.0717566	0	0	0	0	0	0	0

Quality Calculation

- Continuing from example 2, let's take iteration 10,100 and hospital #3. Since hospital #3 is a large public hospital, then β_j^d is the value in b_orgID_3 added to b_hosp_size_large and b_hosp_ownership_public, for the row with iter=10,100. As before, δ^d is the vector, delta1-delta7. However, because δ^d is a 0 vector, then the denominator of the quality calculation is 1: $\sqrt{(\delta^{d\prime}\Sigma\delta^d + 1)} = \sqrt{(0+1)} = 1$. Therefore, the quality for a given hospital draw will simply be based on values in β. The relevant columns are shown in the screenshot above. For this model, the quality estimate for hospital #3 and iteration 10,100 is $q_3^{10,100} = \frac{-(0.779-0.320-0.579)}{1} = 0.120$
- As before, one should do this for all hospital-iterations (after removing burn-in) to obtain the hospital quality distribution necessary for across hospital quality comparisons.

References

Geweke, J., Gowrisankaran, G., & Town, R. J. (2003). Bayesian inference for hospital quality in a selection model. *Econometrica*, *71*(4), 1215-1238.

Contacts

Kelli Marquardt- kelli.r.marquardt@gmail.com
Gautam Gowrisankaran- gautamg2@gmail.com