# Quick-start guide to the Stat-JR 1.0.4 TREE interface

This document provides a ‘quick-start’ guide to using Stat-JR, via its *TREE (Template Reading and Execution Environment)* interface. The development of Stat-JR was a collaborative project, funded by the UK’s ESRC, between the Universities of Bristol and Southampton. More details about Stat-JR can be found on its webpage http://www.bristol.ac.uk/cmm/software/statjr/.

*TREE* is one of the interfaces available for Stat-JR, alongside an eBook-reading interface (*DEEP*), a workflow system (*LEAF*) released as a beta version with Stat-JR 1.0.4, and also a command line interface. For more detailed instructions on how to use TREE, together with worked point-and-click examples, see the *Beginner’s Guide to Stat-JR’s TREE interface* and also the *Advanced User’s Guide to Stat-JR*; the DEEP and LEAF interfaces also have their own user guides.

This quick-start guide assumes you have installed Stat-JR (for details on how to obtain and install Stat-JR, see http://www.bristol.ac.uk/cmm/software/statjr/order-statjr/). Once installed, Stat-JR’s *TREE* interface is started by selecting the *Stat-JR TREE* link from the *Centre for Multilevel Modelling* suite on the start-up menu. This action opens a command prompt window in the background to which commands are printed out, and should also open *TREE’s* welcome page in your browser (Chrome or Firefox are recommended).

**Quick-start guide to the Stat-JR 1.0.4 TREE interface**

ISBN: To be confirmed

Printed in the United Kingdom

This documentation was written by Richard Parker*

If you use Stat-JR in your research, then please cite it as:

Charlton, C.M.J., Michaelides, D.T., Parker, R.M.A., Cameron, B., Szmaragd, C., Yang, H., Zhang, Z., Frazer, A.J., Goldstein, H., Jones, K., Leckie, G., Moreau, L. and Browne, W.J. (2016) *Stat-JR version 1.0.4*. Centre for Multilevel Modelling, University of Bristol & Electronics and Computer Science, University of Southampton, UK.

The initials of Stat-JR are taken from those of the late Jon Rasbash, whose vision was instrumental to its conception. The Stat-JR software system has been primarily developed by Chris Charlton* and Danius Michaelides**, with algebra system development by Bruce Cameron*, and with additional input from William Browne* and Richard Parker*. Core template development by Chris Charlton*, William Browne*, Richard Parker*, Camille Szmaragd* and Zhengzheng Zhang*.

The Stat-JR:TREE software interface was primarily developed by Chris Charlton* and Danius Michaelides**, with additional input from Richard Parker* and William Browne*.

* Centre for Multilevel Modelling, University of Bristol, UK

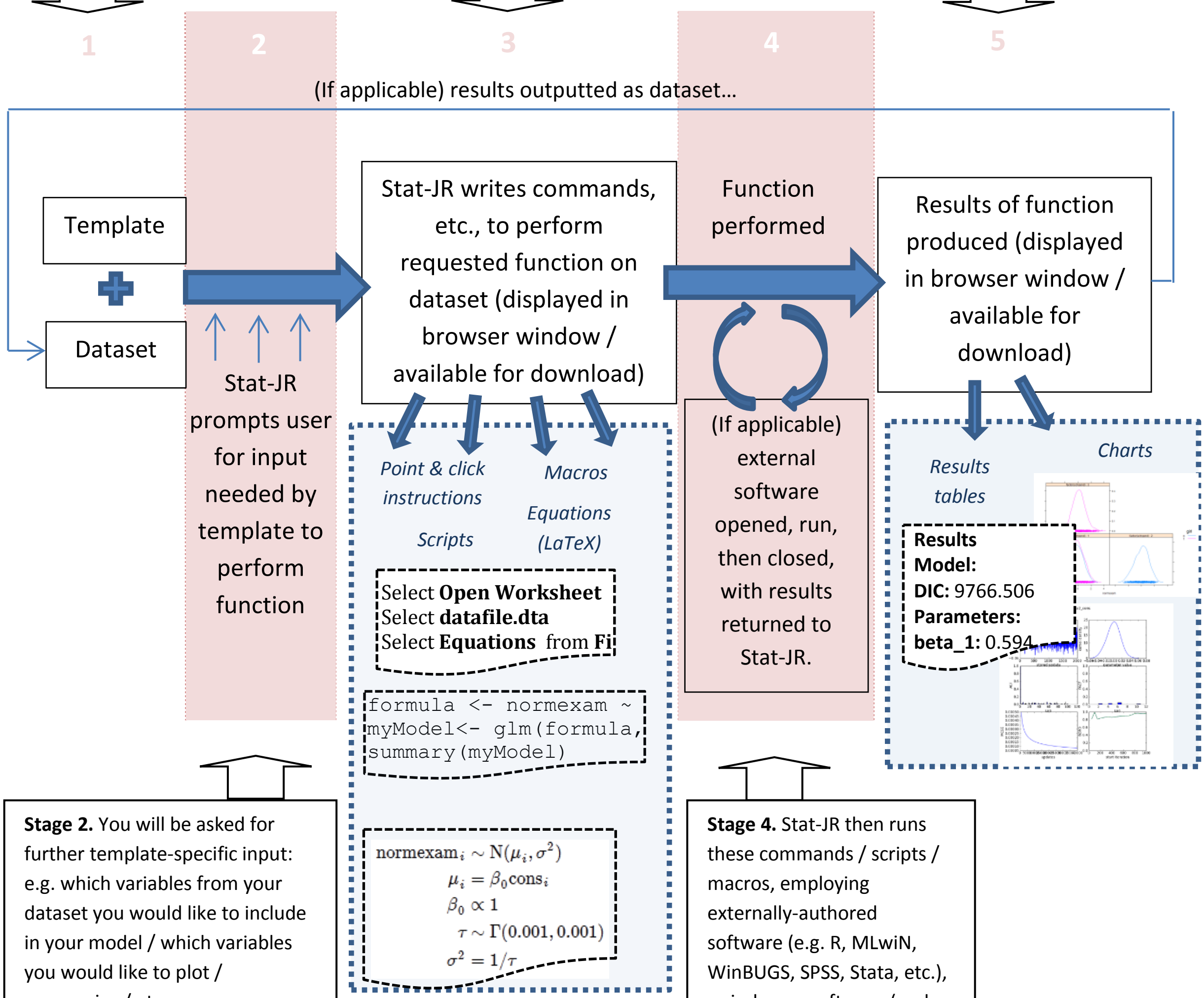
** Electronics and Computer Science, University of Southampton, UK.

When operating Stat-JR through *TREE*, you generally proceed through the following five stages:

**Stage 1.** Firstly, choose the dataset you want to analyse / plot / summarise / etc., and the template you want to use to do so. Each template contains commands to perform certain functions: some run models, others plot graphs, or provide summary statistics, and so on…

**Stage 3.** Once you've answered all the input queries, Stat-JR generates all the commands, scripts, macros, equations, and instructions necessary to perform, or describe, the function you've requested. You can view these within *TREE*, and can download them too…

**Stage 5.** Finally, the results are returned; depending on the template these can include model estimates, graphs, summary tables, and so on. Again, these can be viewed within *TREE*, and are also downloadable. The output may also include datasets (e.g. MCMC chains), which you can then feed back into the system by matching them up with a template back in Stage 1.


1
2
3
4
5
(If applicable) results outputted as dataset…
Template
Dataset
Stat-JR prompts user for input needed by template to perform function
Stat-JR writes commands, etc., to perform requested function on dataset (displayed in browser window / available for download)
Point & click instructions
Macros
Equations (LaTeX)
Scripts
Function performed
(If applicable) external software opened, run, then closed, with results returned to Stat-JR.
Results of function produced (displayed in browser window / available for download)
Results tables
Charts

Select **Open Worksheet**
Select **datafile.dta**
Select **Equations** from **Fi**

```
formula <- normexam ~
myModel<- glm(formula,
summary(myModel)
```

$$
\begin{aligned}
\text{normexam}_i &\sim \mathrm{N}(\mu_i, \sigma^2)\\
\mu_i &= \beta_0 \text{cons}_i\\
\beta_0 &\propto 1\\
\tau &\sim \Gamma(0.001, 0.001)\\
\sigma^2 &= 1/\tau
\end{aligned}
$$

**Results**
**Model:**
**DIC:** 9766.506
**Parameters:**
**beta_1:** 0.594

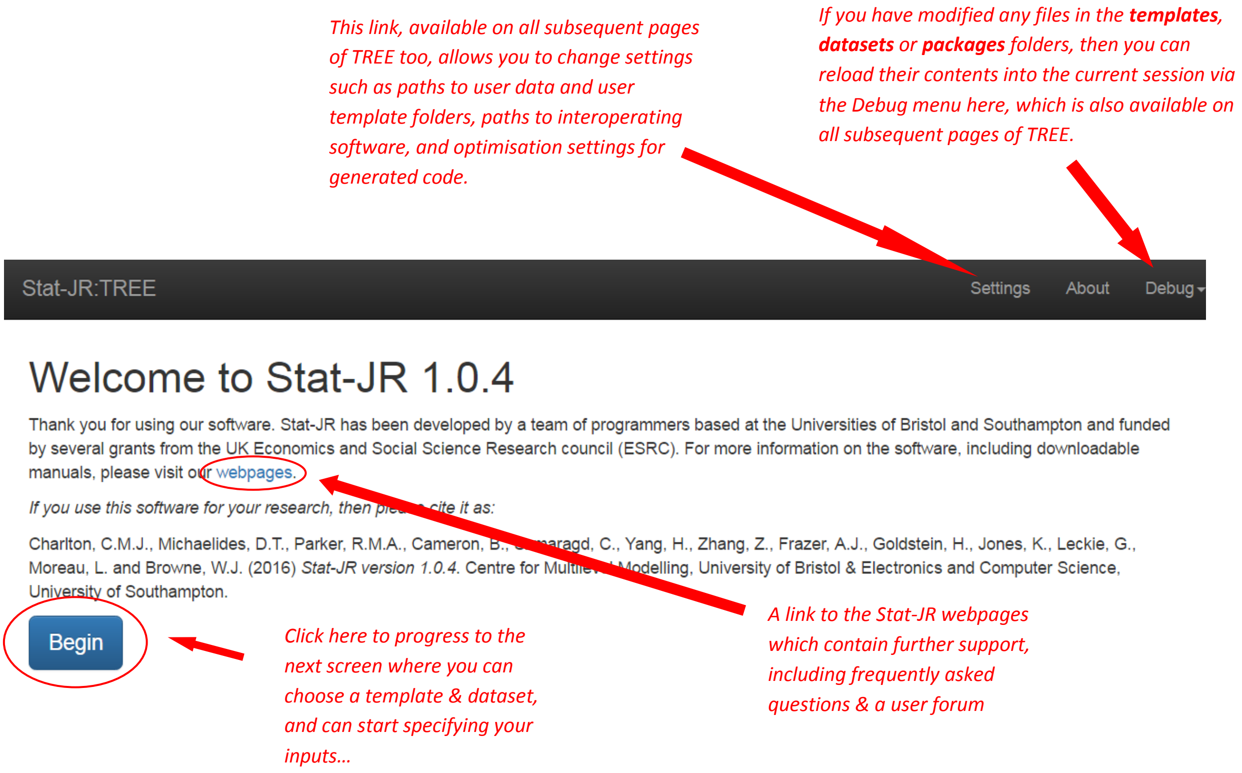
**Stage 2.** You will be asked for further template-specific input: e.g. which variables from your dataset you would like to include in your model / which variables you would like to plot / summarise / etc.

**Stage 4.** Stat-JR then runs these commands / scripts / macros, employing externally-authored software (e.g. R, MLwiN, WinBUGS, SPSS, Stata, etc.), or in-house software (such as the eStat engine), as appropriate.

Below we briefly highlight the main features, with screenshots, of each of these five stages.

### Stage 1: Selecting a template & dataset

- On opening Stat-JR, the page below, containing introductory information, will be displayed in a web browser. To proceed to choosing a template and dataset, click on the **Begin** button.



*This link, available on all subsequent pages of TREE too, allows you to change settings such as paths to user data and user template folders, paths to interoperating software, and optimisation settings for generated code.*

*If you have modified any files in the **templates**, **datasets** or **packages** folders, then you can reload their contents into the current session via the Debug menu here, which is also available on all subsequent pages of TREE.*

Welcome to Stat-JR 1.0.4

Thank you for using our software. Stat-JR has been developed by a team of programmers based at the Universities of Bristol and Southampton and funded by several grants from the UK Economics and Social Science Research council (ESRC). For more information on the software, including downloadable manuals, please visit our webpages.

*If you use this software for your research, then please cite it as:*

Charlton, C.M.J., Michaelides, D.T., Parker, R.M.A., Cameron, B., Szmaragd, C., Yang, H., Zhang, Z., Frazer, A.J., Goldstein, H., Jones, K., Leckie, G., Moreau, L. and Browne, W.J. (2016) *Stat-JR version 1.0.4*. Centre for Multilevel Modelling, University of Bristol & Electronics and Computer Science, University of Southampton.

*Click here to progress to the next screen where you can choose a template & dataset, and can start specifying your inputs…*

*A link to the Stat-JR webpages which contain further support, including frequently asked questions & a user forum*

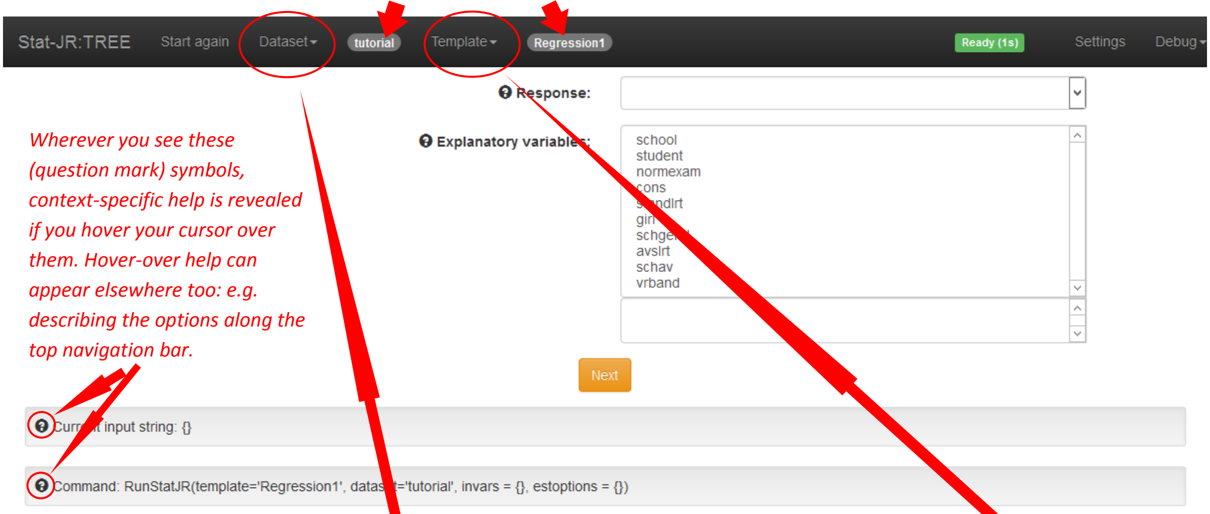
- Having pressed **Begin**, the page below will be displayed. Note that here, and on other screens, wherever you see the question mark symbols, **context-specific help** is revealed if you hover your cursor over them. Hover-over help can appear elsewhere too: e.g. describing the options along the top navigation bar.

- Here you can specify the **template** and **dataset** you want to use, and then begin to specify your **inputs**.

- Selecting **Dataset > Choose** or **Template > Choose** from the top bar will reveal lists of available datasets and templates. For each, find the one you want from the list, and then press **Use**.

- Note, when choosing a template, you can use the **cloud terms to help your search**: the blue tags describe functional aspects of the templates, whilst the red terms describe which engines / packages the templates support (you can combine search terms by clicking on more than one, and cancel your selections by pressing **[reset]**).

*Here you can see which dataset and template are currently selected. Hovering your cursor over these names will reveal a description of each (if available).*



Response:

Explanatory variables: school, student, normexam, cons, standlrt, girl, schgend, avslrt, schav, vrband

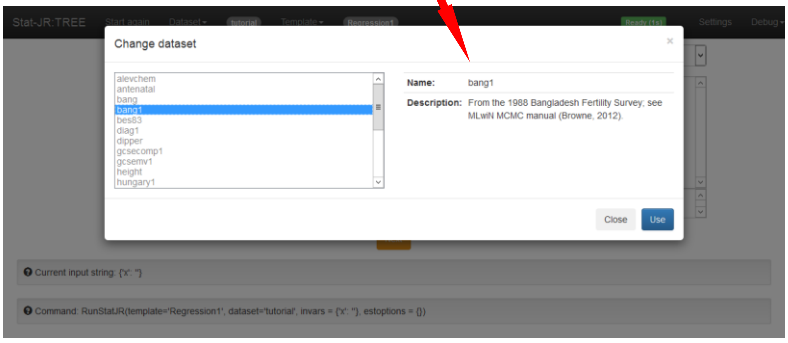
Current input string: {}

Command: RunStatJR(template='Regression1', dataset='tutorial', invars = {}, estoptions = {})

*Wherever you see these (question mark) symbols, context-specific help is revealed if you hover your cursor over them. Hover-over help can appear elsewhere too: e.g. describing the options along the top navigation bar.*

*Clicking on the down arrow symbol just to the right of the* ***Dataset*** *heading in the top bar will bring up a menu. Select* ***Choose*** *to bring up the window, below, allowing you to nominate a dataset other than that currently selected…*

*…and likewise for the* ***Template…***



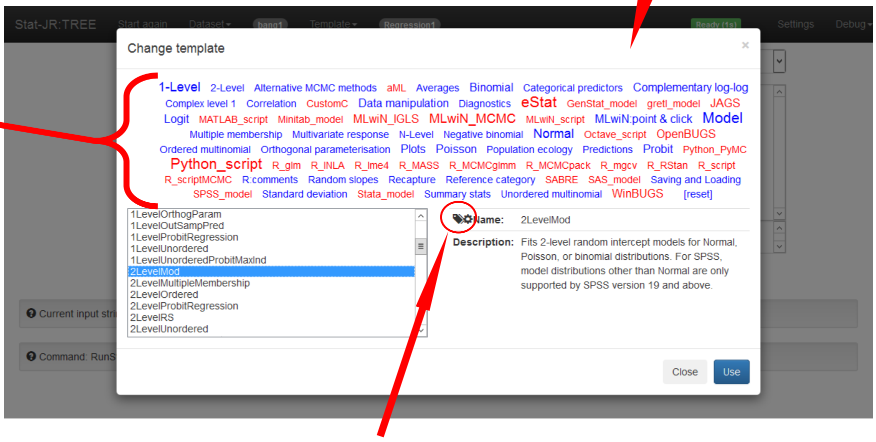
Name: bang1

Description: From the 1988 Bangladesh Fertility Survey; see MLwiN MCMC manual (Browne, 2012).

Current input string: {'x': ''}

Command: RunStatJR(template='Regression1', dataset='tutorial', invars = {'x': ''}, estoptions = {})

*You can select one or more of these terms to help you find relevant templates; the blue tags describe the functional aspects of a template, whilst the red terms describe the engines / packages supported by a template. To unselect terms, press [reset]*



Name: 2LevelMod

Description: Fits 2-level random intercept models for Normal, Poisson, or binomial distributions. For SPSS, model distributions other than Normal are only supported by SPSS version 19 and above.

Current input stri

Command: RunS

*Clicking the 'label' symbol brings up a list of tags, whilst clicking the 'cog' symbol brings up a list of supported engines / packages.*

## Stage 2: Providing template-specific input

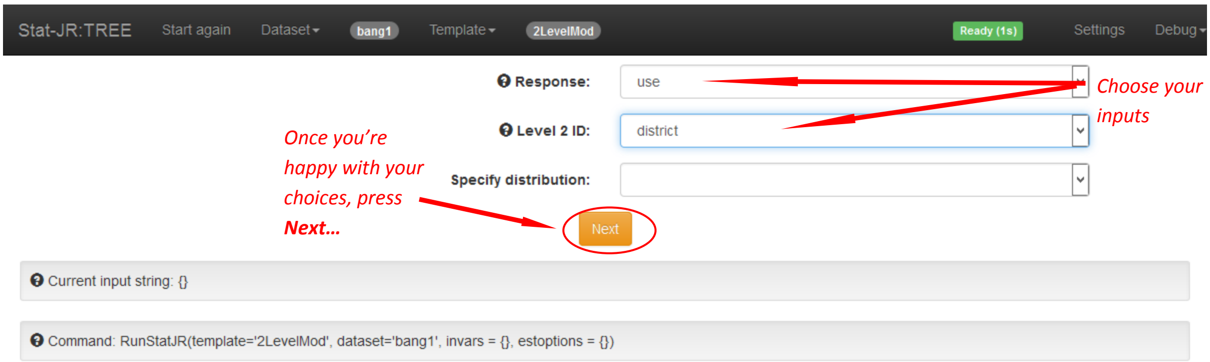
- Once your desired **Dataset** and **Template** are selected, you can start answering the input questions back on the main page. These are required by Stat-JR to allow the template to perform the appropriate executions with your dataset; these inputs vary between templates, and also within templates too, depending on your earlier choices as you progress through the screens.

- For multi-choice lists you can de-select variables by simply clicking on their name in the list of selected items.

- Press **Next** each time you've completed the input questions on the current page.

- Then, if applicable, more inputs will be revealed, and those you have already selected will be greyed-out. However, you can still change each input via the **remove** button which you'll see next to each one. Alternatively, to re-specify *all* your inputs, press **Start again** (in the top bar).

- When asked for the **Name of the output results**, this will be the name given to any outputted dataset which results from running the template (see Stage 5).



Response: use

Level 2 ID: district

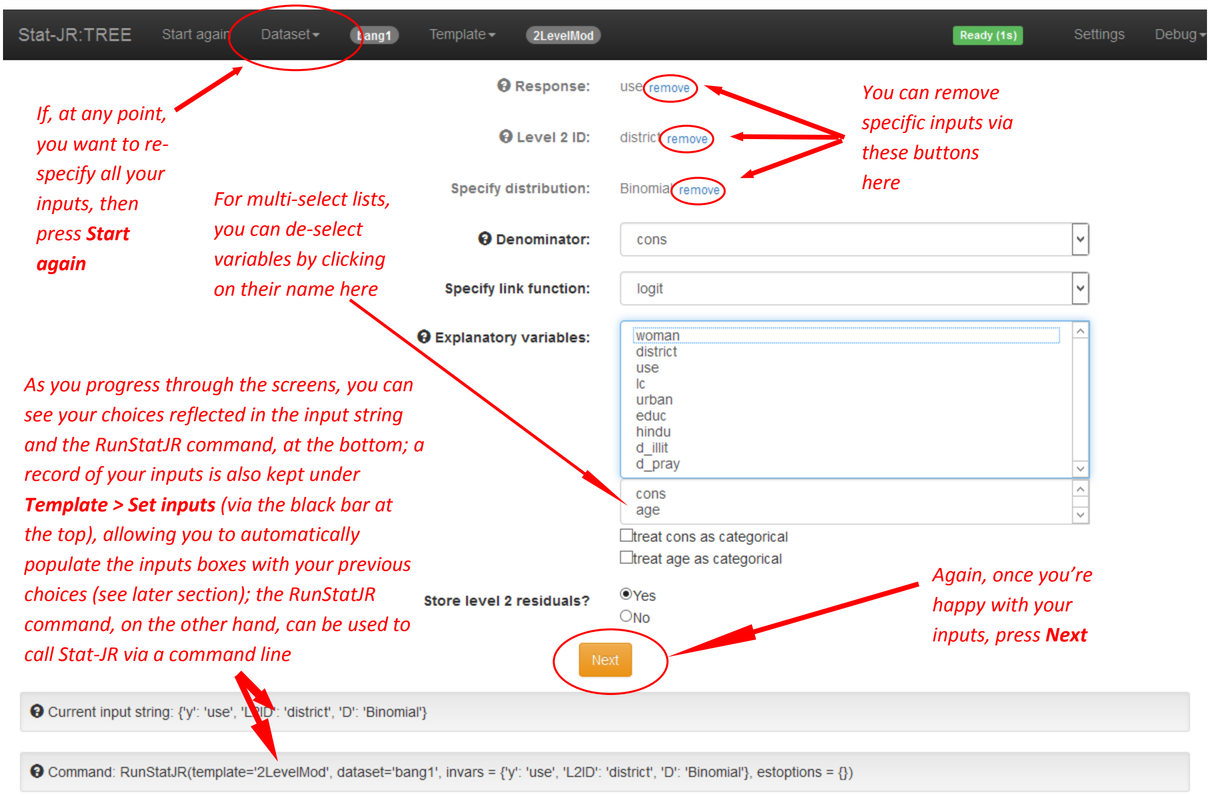
Specify distribution:

*Choose your inputs*

*Once you're happy with your choices, press **Next…***

Current input string: {}

Command: RunStatJR(template='2LevelMod', dataset='bang1', invars = {}, estoptions = {})



Response: use remove

Level 2 ID: district remove

Specify distribution: Binomial remove

Denominator: cons

Specify link function: logit

Explanatory variables: woman, district, use, lc, urban, educ, hindu, d_illit, d_pray; cons, age

□ treat cons as categorical

□ treat age as categorical

Store level 2 residuals? ● Yes ○ No

Next

Current input string: {'y': 'use', 'L2ID': 'district', 'D': 'Binomial'}

Command: RunStatJR(template='2LevelMod', dataset='bang1', invars = {'y': 'use', 'L2ID': 'district', 'D': 'Binomial'}, estoptions = {})

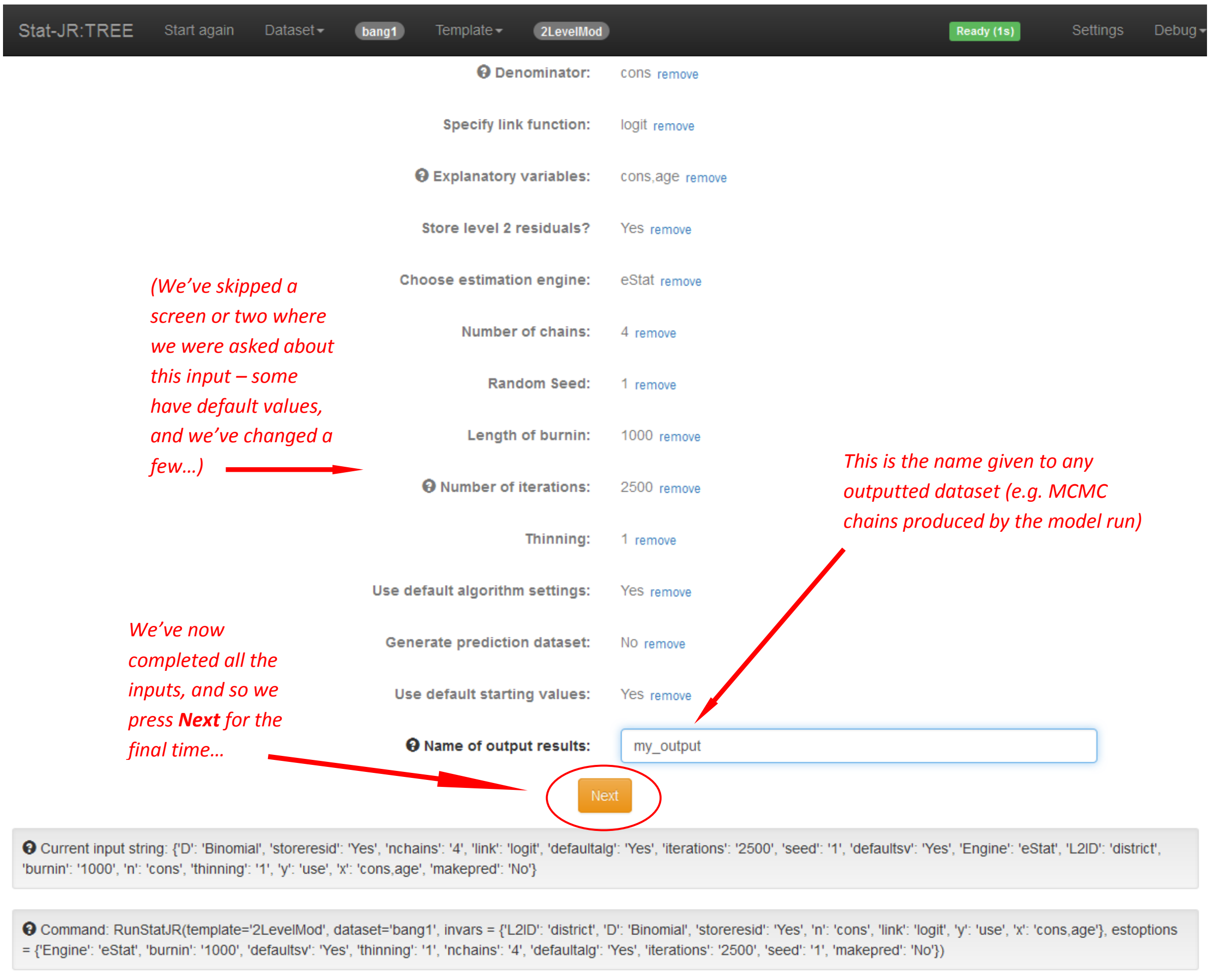
*If, at any point, you want to re-specify all your inputs, then press* ***Start again***

*For multi-select lists, you can de-select variables by clicking on their name here*

*You can remove specific inputs via these buttons here*

*As you progress through the screens, you can see your choices reflected in the input string and the RunStatJR command, at the bottom; a record of your inputs is also kept under* ***Template > Set inputs*** *(via the black bar at the top), allowing you to automatically populate the inputs boxes with your previous choices (see later section); the RunStatJR command, on the other hand, can be used to call Stat-JR via a command line*

*Again, once you're happy with your inputs, press* ***Next***



Denominator: cons remove

Specify link function: logit remove

Explanatory variables: cons,age remove

Store level 2 residuals? Yes remove

Choose estimation engine: eStat remove

Number of chains: 4 remove

Random Seed: 1 remove

Length of burnin: 1000 remove

Number of iterations: 2500 remove

Thinning: 1 remove

Use default algorithm settings: Yes remove

Generate prediction dataset: No remove

Use default starting values: Yes remove

Name of output results: my_output

Next

Current input string: {'D': 'Binomial', 'storeresid': 'Yes', 'nchains': '4', 'link': 'logit', 'defaultalg': 'Yes', 'iterations': '2500', 'seed': '1', 'defaultsv': 'Yes', 'Engine': 'eStat', 'L2ID': 'district', 'burnin': '1000', 'n': 'cons', 'thinning': '1', 'y': 'use', 'x': 'cons,age', 'makepred': 'No'}

Command: RunStatJR(template='2LevelMod', dataset='bang1', invars = {'L2ID': 'district', 'D': 'Binomial', 'storeresid': 'Yes', 'n': 'cons', 'link': 'logit', 'y': 'use', 'x': 'cons,age'}, estoptions = {'Engine': 'eStat', 'burnin': '1000', 'defaultsv': 'Yes', 'thinning': '1', 'nchains': '4', 'defaultalg': 'Yes', 'iterations': '2500', 'seed': '1', 'makepred': 'No'})

*(We've skipped a screen or two where we were asked about this input – some have default values, and we've changed a few…)*

*This is the name given to any outputted dataset (e.g. MCMC chains produced by the model run)*

*We've now completed all the inputs, and so we press* ***Next*** *for the final time…*

## Stage 3: Outputting the files to run the desired execution

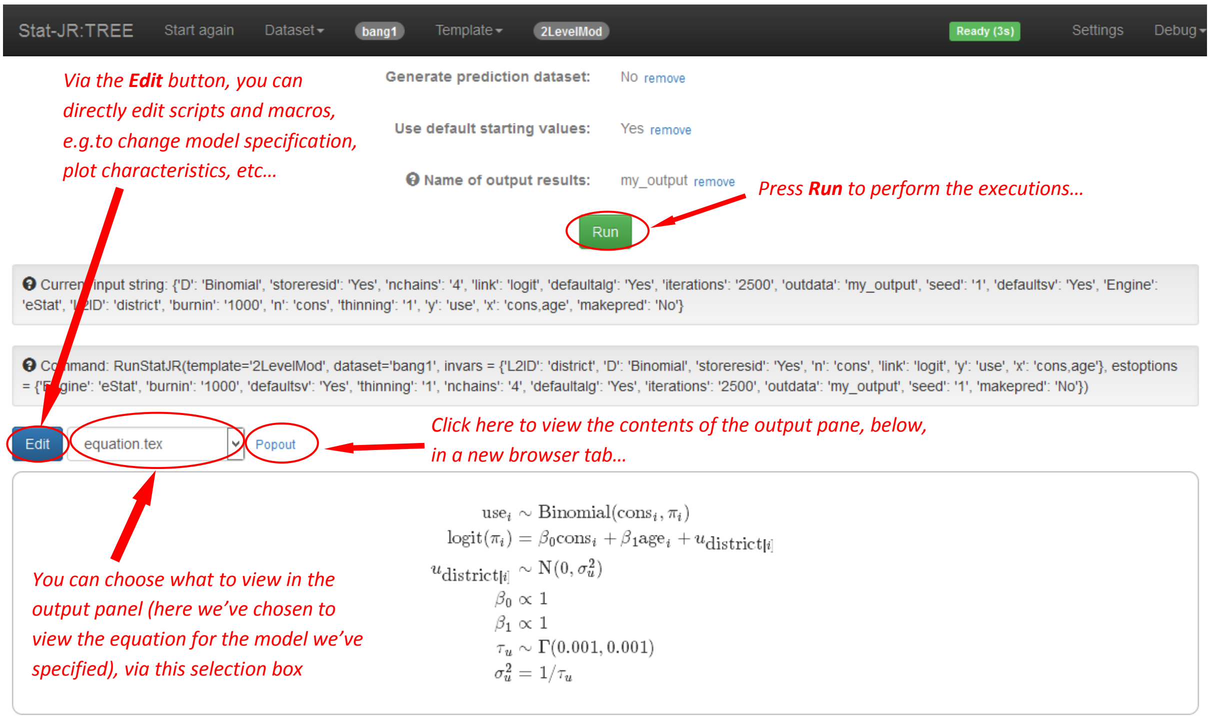
- Once you're pressed **Next** after the final input, Stat-JR returns a number of initial outputs which you can view in the output pane at the bottom of the window.

- Note that Stat-JR hasn't done everything you want it to do yet: it's just producing preliminary files telling you what it's going to do, and how it's going to do it.

- To select particular content to view in the output pane, use the drop-down menu just above it.

- The **Popout** button, just above the output pane, allows you to view its contents in a new browser tab.

- Pressing **Run** performs the executions described by the scripts, etc, returned in the output pane.



*Via the **Edit** button, you can directly edit scripts and macros, e.g.to change model specification, plot characteristics, etc…*

Generate prediction dataset: No remove

Use default starting values: Yes remove

Name of output results: my_output remove

*Press **Run** to perform the executions…*

Current input string: {'D': 'Binomial', 'storeresid': 'Yes', 'nchains': '4', 'link': 'logit', 'defaultalg': 'Yes', 'iterations': '2500', 'outdata': 'my_output', 'seed': '1', 'defaultsv': 'Yes', 'Engine': 'eStat', 'L2ID': 'district', 'burnin': '1000', 'n': 'cons', 'thinning': '1', 'y': 'use', 'x': 'cons,age', 'makepred': 'No'}

Command: RunStatJR(template='2LevelMod', dataset='bang1', invars = {'L2ID': 'district', 'D': 'Binomial', 'storeresid': 'Yes', 'n': 'cons', 'link': 'logit', 'y': 'use', 'x': 'cons,age'}, estoptions = {'Engine': 'eStat', 'burnin': '1000', 'defaultsv': 'Yes', 'thinning': '1', 'nchains': '4', 'defaultalg': 'Yes', 'iterations': '2500', 'outdata': 'my_output', 'seed': '1', 'makepred': 'No'})

*Click here to view the contents of the output pane, below, in a new browser tab…*

$$\begin{aligned}
\mathrm{use}_i &\sim \mathrm{Binomial}(\mathrm{cons}_i, \pi_i)\\
\mathrm{logit}(\pi_i) &= \beta_0\mathrm{cons}_i + \beta_1\mathrm{age}_i + u_{\mathrm{district}[i]}\\
u_{\mathrm{district}[i]} &\sim \mathrm{N}(0, \sigma_u^2)\\
\beta_0 &\propto 1\\
\beta_1 &\propto 1\\
\tau_u &\sim \Gamma(0.001, 0.001)\\
\sigma_u^2 &= 1/\tau_u
\end{aligned}$$

*You can choose what to view in the output panel (here we've chosen to view the equation for the model we've specified), via this selection box*

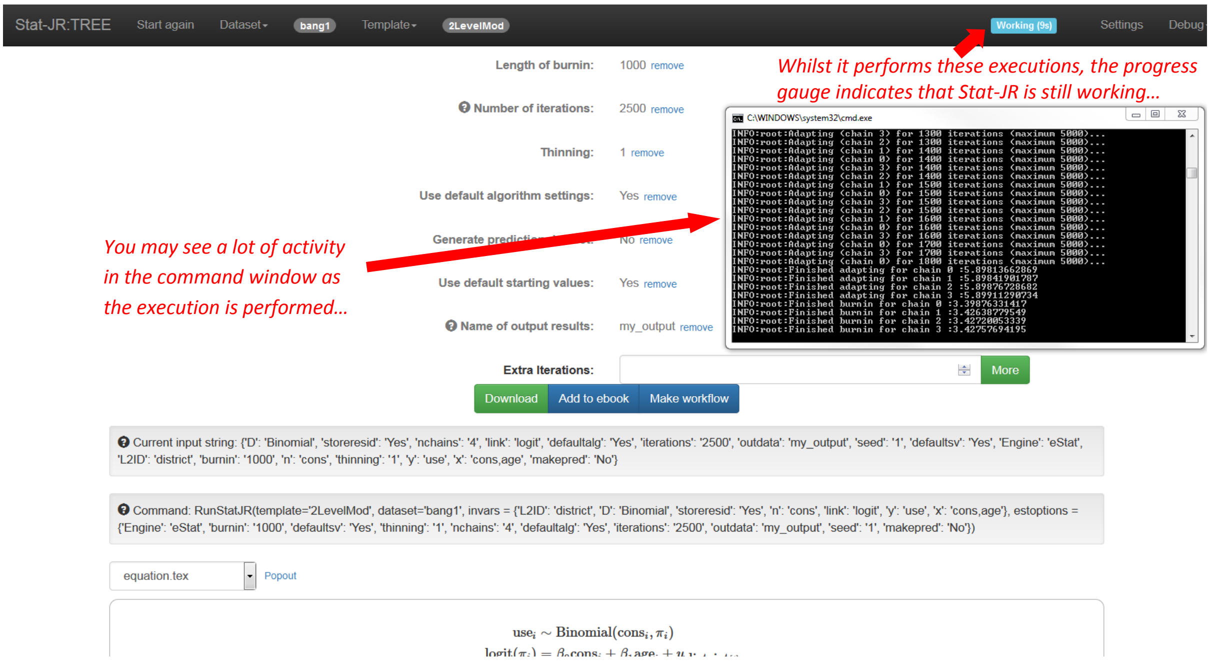
### Stage 4: Running the execution

- Once you've pressed **Run**, the executions specified by you are peformed.

- Depending on your choices, this may take anything from a second or two (e.g. to produce a simple plot, fit a model using a non-iterative method of estimation, produce summary data, etc.), to many minutes (e.g. to run MCMC chains for a large number of iterations).

- If appropriate (e.g. if the template supports inter-operability, and if you have chosen to employ it when prompted), externally-authored software packages (e.g. R, MLwiN, WinBUGS, SPSS, etc.) are opened, run, then closed, and the results are returned to Stat-JR.

- Whilst the execution runs, you may see a lot of activity in the black command window, which may help you keep a track of progress.



*Whilst it performs these executions, the progress gauge indicates that Stat-JR is still working…*

*You may see a lot of activity in the command window as the execution is performed…*

Length of burnin: 1000 remove

Number of iterations: 2500 remove

Thinning: 1 remove

Use default algorithm settings: Yes remove

Generate predictio: No remove

Use default starting values: Yes remove

Name of output results: my_output remove

Extra Iterations:

```
INFO:root:Adapting (chain 3) for 1300 iterations (maximum 5000)...
INFO:root:Adapting (chain 2) for 1300 iterations (maximum 5000)...
INFO:root:Adapting (chain 1) for 1400 iterations (maximum 5000)...
INFO:root:Adapting (chain 0) for 1400 iterations (maximum 5000)...
INFO:root:Adapting (chain 3) for 1400 iterations (maximum 5000)...
INFO:root:Adapting (chain 2) for 1400 iterations (maximum 5000)...
INFO:root:Adapting (chain 1) for 1500 iterations (maximum 5000)...
INFO:root:Adapting (chain 0) for 1500 iterations (maximum 5000)...
INFO:root:Adapting (chain 3) for 1500 iterations (maximum 5000)...
INFO:root:Adapting (chain 2) for 1500 iterations (maximum 5000)...
INFO:root:Adapting (chain 1) for 1600 iterations (maximum 5000)...
INFO:root:Adapting (chain 0) for 1600 iterations (maximum 5000)...
INFO:root:Adapting (chain 3) for 1600 iterations (maximum 5000)...
INFO:root:Adapting (chain 0) for 1700 iterations (maximum 5000)...
INFO:root:Adapting (chain 3) for 1700 iterations (maximum 5000)...
INFO:root:Adapting (chain 0) for 1800 iterations (maximum 5000)...
INFO:root:Finished adapting for chain 0 :5.89813662869
INFO:root:Finished adapting for chain 1 :5.89841901787
INFO:root:Finished adapting for chain 2 :5.89876728682
INFO:root:Finished adapting for chain 3 :5.89911290734
INFO:root:Finished burnin for chain 0 :3.39876331417
INFO:root:Finished burnin for chain 1 :3.42638779549
INFO:root:Finished burnin for chain 2 :3.42720053339
INFO:root:Finished burnin for chain 3 :3.42757694195
```

Current input string: {'D': 'Binomial', 'storeresid': 'Yes', 'nchains': '4', 'link': 'logit', 'defaultalg': 'Yes', 'iterations': '2500', 'outdata': 'my_output', 'seed': '1', 'defaultsv': 'Yes', 'Engine': 'eStat', 'L2ID': 'district', 'burnin': '1000', 'n': 'cons', 'thinning': '1', 'y': 'use', 'x': 'cons,age', 'makepred': 'No'}

Command: RunStatJR(template='2LevelMod', dataset='bang1', invars = {'L2ID': 'district', 'D': 'Binomial', 'storeresid': 'Yes', 'n': 'cons', 'link': 'logit', 'y': 'use', 'x': 'cons,age'}, estoptions = {'Engine': 'eStat', 'burnin': '1000', 'defaultsv': 'Yes', 'thinning': '1', 'nchains': '4', 'defaultalg': 'Yes', 'iterations': '2500', 'outdata': 'my_output', 'seed': '1', 'makepred': 'No'})

equation.tex

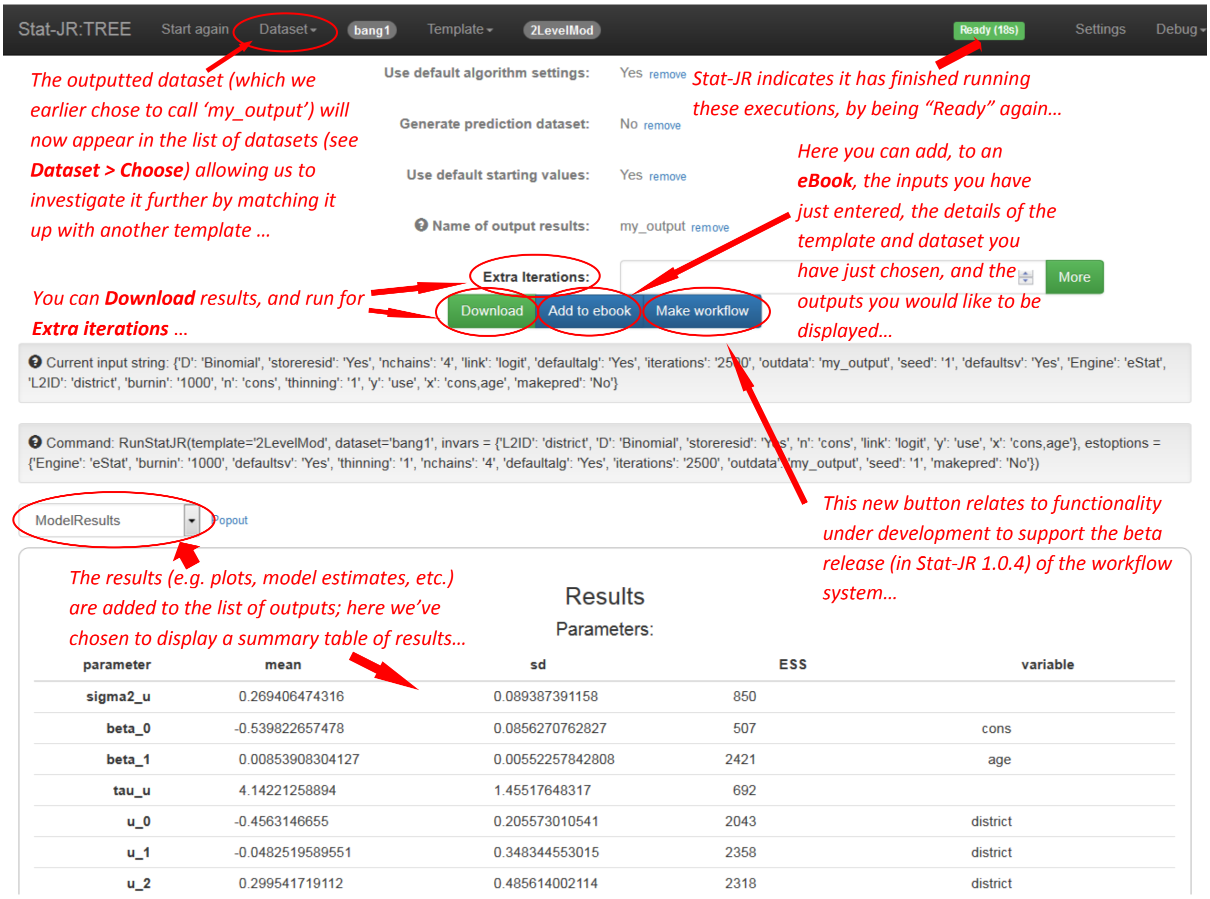
$$\mathrm{use}_i \sim \mathrm{Binomial}(\mathrm{cons}_i, \pi_i)$$

## Stage 5: The results

- Once the executions have run, the progress guage, in the top-right corner, will change from "Working" to "Ready", and the drop-down list, just above the output pane, will now be populated with more results.

- Depending on the template, a range of buttons / boxes appear above the output pane allowing you to e.g. **Download** the results, **Add to ebook**, **Make workflow** (functionality under development to support beta release of the LEAF workflow system), and run chains for **Extra iterations**.

- If applicable, an outputted dataset now appears in the list of datasets (see **Dataset > Choose**, via the top bar).



*The outputted dataset (which we earlier chose to call 'my_output') will now appear in the list of datasets (see **Dataset > Choose**) allowing us to investigate it further by matching it up with another template …*

*Stat-JR indicates it has finished running these executions, by being "Ready" again…*

Use default algorithm settings: Yes remove

Generate prediction dataset: No remove

Use default starting values: Yes remove

Name of output results: my_output remove

*Here you can add, to an **eBook**, the inputs you have just entered, the details of the template and dataset you have just chosen, and the outputs you would like to be displayed…*

*You can **Download** results, and run for **Extra iterations** …*

Current input string: {'D': 'Binomial', 'storeresid': 'Yes', 'nchains': '4', 'link': 'logit', 'defaultalg': 'Yes', 'iterations': '2500', 'outdata': 'my_output', 'seed': '1', 'defaultsv': 'Yes', 'Engine': 'eStat', 'L2ID': 'district', 'burnin': '1000', 'n': 'cons', 'thinning': '1', 'y': 'use', 'x': 'cons,age', 'makepred': 'No'}

Command: RunStatJR(template='2LevelMod', dataset='bang1', invars = {'L2ID': 'district', 'D': 'Binomial', 'storeresid': 'Yes', 'n': 'cons', 'link': 'logit', 'y': 'use', 'x': 'cons,age'}, estoptions = {'Engine': 'eStat', 'burnin': '1000', 'defaultsv': 'Yes', 'thinning': '1', 'nchains': '4', 'defaultalg': 'Yes', 'iterations': '2500', 'outdata': 'my_output', 'seed': '1', 'makepred': 'No'})

*This new button relates to functionality under development to support the beta release (in Stat-JR 1.0.4) of the workflow system…*

*The results (e.g. plots, model estimates, etc.) are added to the list of outputs; here we've chosen to display a summary table of results…*

Results

Parameters:

| parameter | mean | sd | ESS | variable |
|---|---|---|---|---|
| sigma2_u | 0.269406474316 | 0.089387391158 | 850 | |
| beta_0 | -0.539822657478 | 0.0856270762827 | 507 | cons |
| beta_1 | 0.00853908304127 | 0.00552257842808 | 2421 | age |
| tau_u | 4.14221258894 | 1.45517648317 | 692 | |
| u_0 | -0.4563146655 | 0.205573010541 | 2043 | district |
| u_1 | -0.0482519589551 | 0.348344553015 | 2358 | district |
| u_2 | 0.299541719112 | 0.485614002114 | 2318 | district |

This ends the quick-start guide to *TREE*; for a more detailed overview, see the *Beginner's Guide to Stat-JR's TREE interface* and also the *Advanced User's Guide to Stat-JR*, plus further information and guidance on the Stat-JR website, http://www.bristol.ac.uk/cmm/software/statjr/.