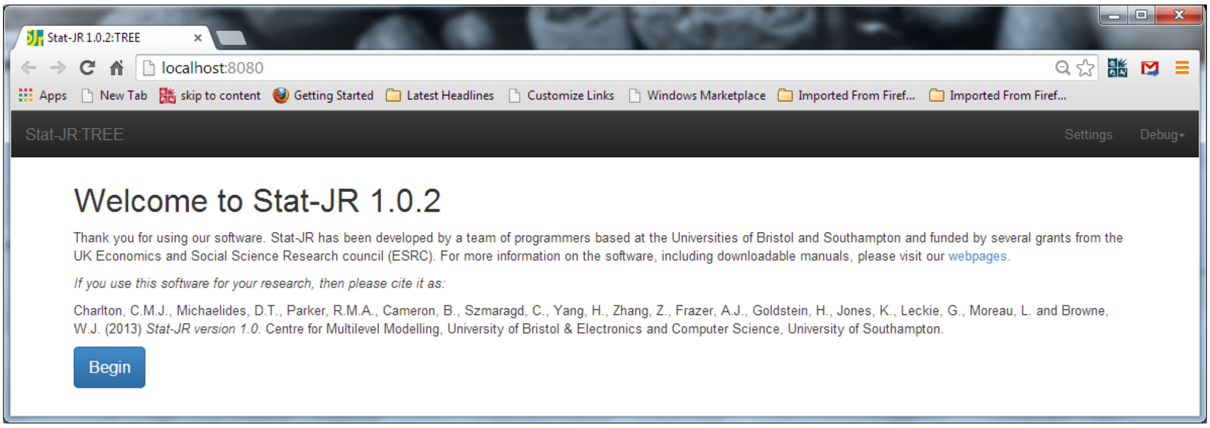
eBook Writing Workshop – Practical 2: Making an interactive eBook

Introduction

In this practical we will follow on from our work in practical 1 to now make eBooks that allow the reader to make interactive decisions and thus the eBook will look different each time it is run. We will demonstrate this in two ways. Firstly by starting afresh and constructing a new eBook that has some interactivity directly from the eBook writer in TREE. Second we will revisit the final eBook we created in the first practical and show how we can edit the files that make up this eBook to make that eBook interactive.

A New Interactive eBook

As in practical 1 we will begin by starting up the TREE interface which is achieved by double-clicking on **tree.cmd** in the base directory of the Stat-JR install. Note that if you already have TREE open then it is important to close it and start again as TREE is remembering all the operations you have done so far and we want to start afresh. The welcome screen will appear in the browser as before:



Welcome to Stat-JR 1.0.2

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.

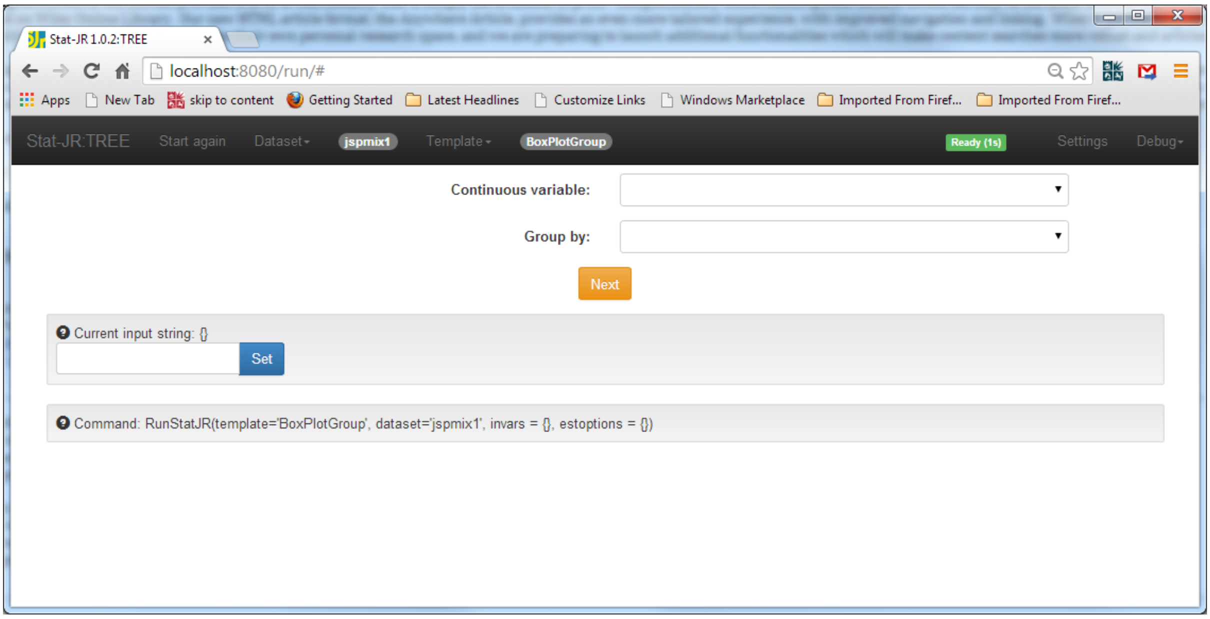
If you use this software for your research, then please cite it as:

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. (2013) *Stat-JR version 1.0.* Centre for Multilevel Modelling, University of Bristol & Electronics and Computer Science, University of Southampton.

We will click on **Begin** to bring up the main screen for Stat-JR and we will change our dataset as we did in practical 1 (i.e. via the **Choose** option on the **Dataset** pull-down list). I will use the **jspmix1** dataset as I did in practical 1 but please use your own dataset as you did before. We will dispense with summarising the dataset here and instead go straight into some preliminary graphing of the data.

In practical 1 we looked at adding a histogram to view the potential response variable and see if it is normally distributed. We will return to that eBook later so to add some variety we will here look at a second graphing template that does comparative boxplots, namely **BoxPlotGroup.** If you have a categorical predictor in your dataset then you could do something similar here, if you only have continuous predictors you might instead consider using the **XYPlot** template and plot the response against your predictors.

To access this template click on the **Template** pull-down list and select **Choose**. From the list of templates select **BoxPlotGroup** and click on **Use**. The main screen will now look as follows:



Continuous variable:

Group by:

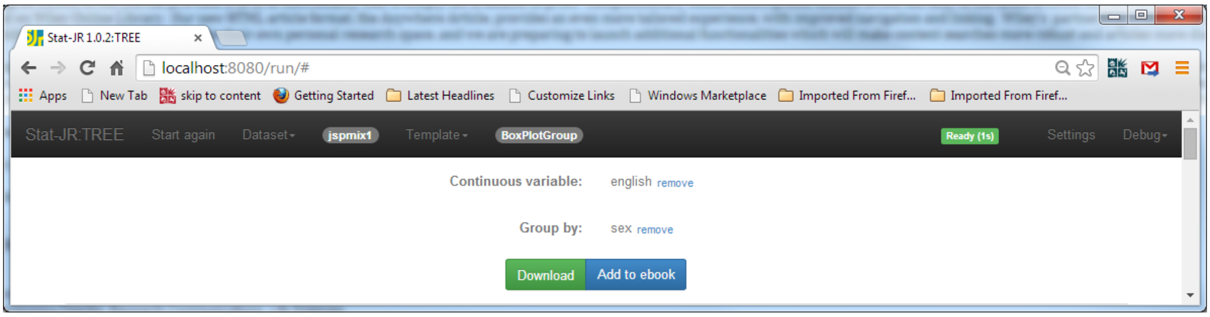
Next

Current input string: {}

Set

Command: RunStatJR(template='BoxPlotGroup', dataset='jspmix1', invars = {}, estoptions = {})

In practical 1 we fitted a model for **english** with a binary predictor, **sex** which we found to be significant. We could see what a boxplot of the English variable for each gender category looks like and whether this might indicate differences. We therefore fill in the inputs as follows and click on **Next** and **Run**:

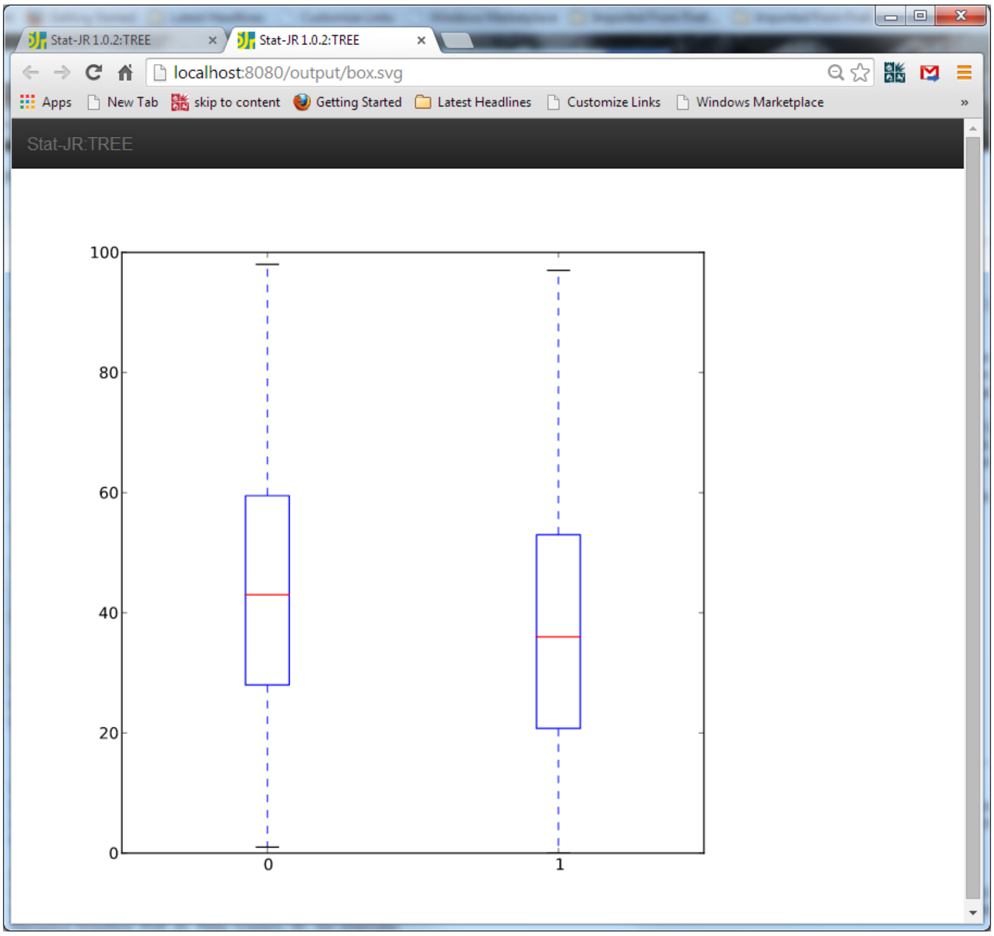


Continuous variable: english remove

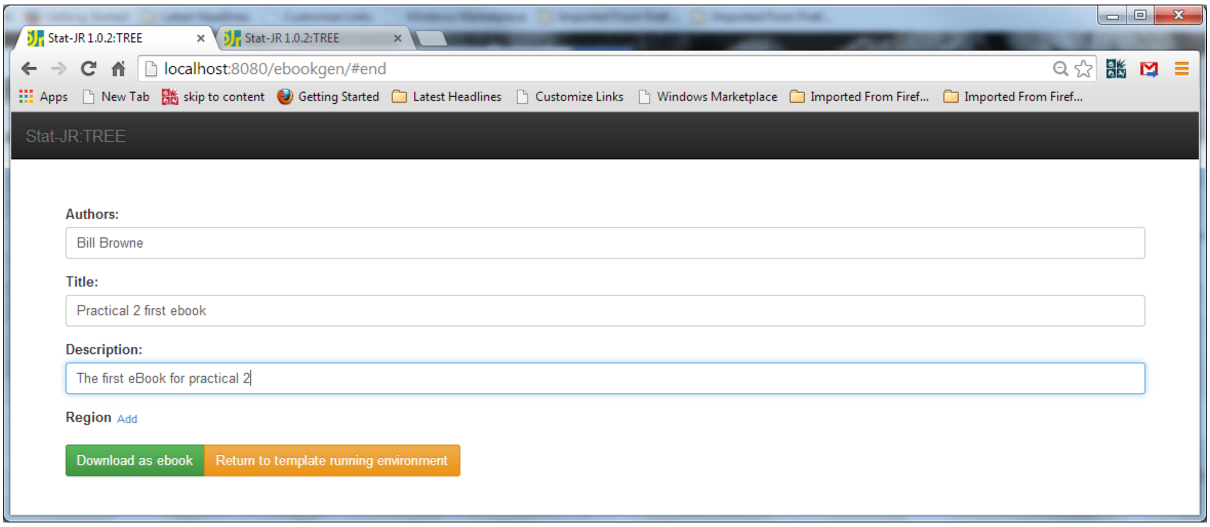
Group by: sex remove

Download Add to ebook

The boxplots produced can be selected from the output object list and popped out into a new tab by selecting **box.svg** from the pull-down list and clicking on **Popout**. The new tab will look thus:



Here we see the lower median value for sex category 1 (boys) compared to category 0 (girls) which we found to be significant in the first practical. There are several other categorical variables in the **jspmix1** dataset so we might like to give the eBook reader the chance to look at these. We could do this in TREE by producing lots of boxplots and putting each one into an eBook in a similar way to the logbook form used in practical 1, but the interactive alternative is to let the reader try this themselves in the eBook environment. We will therefore begin our eBook-writing here by including this boxplot in the first page of our new eBook. To start this off click on the **Add to ebook** button on the main screen of Stat-JR and fill in the details requested in the window.



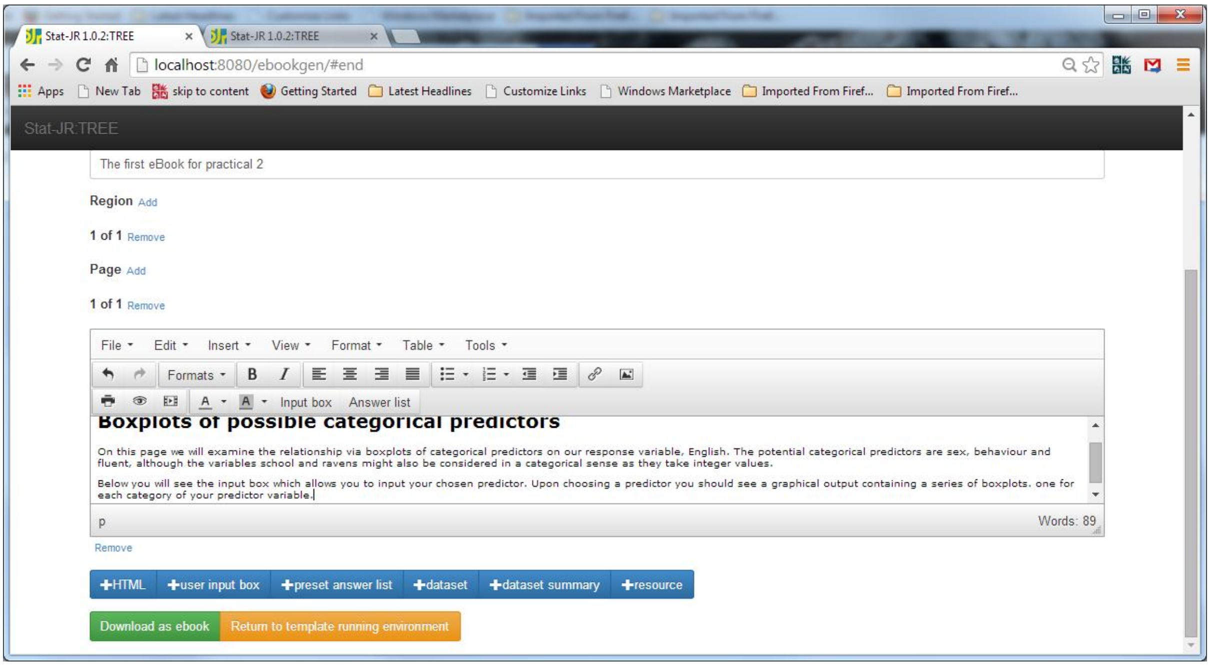
Authors: Bill Browne

Title: Practical 2 first ebook

Description: The first eBook for practical 2

Region

Next we **Add** a **Region** and a **Page** (by clicking on the **Add** buttons to the right of the names) as follows including some text in the first HTML input box:



The first eBook for practical 2

Region Add

1 of 1 Remove

Page Add

1 of 1 Remove

Boxplots of possible categorical predictors

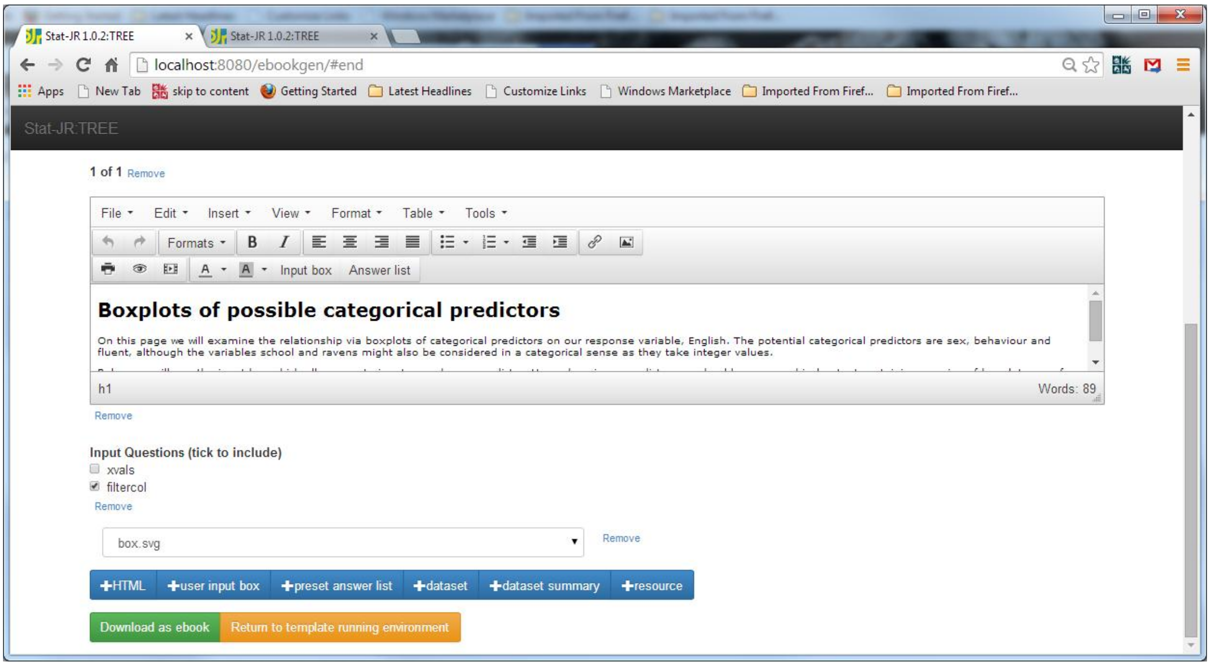
On this page we will examine the relationship via boxplots of categorical predictors on our response variable, English. The potential categorical predictors are sex, behaviour and fluent, although the variables school and ravens might also be considered in a categorical sense as they take integer values.

Below you will see the input box which allows you to input your chosen predictor. Upon choosing a predictor you should see a graphical output containing a series of boxplots. one for each category of your predictor variable.

p Words: 89

Remove

We then need to add a **user input box** so that the eBook reader can decide what to plot, along with the **resource** to be plotted. This is done as follows:



1 of 1 Remove

Boxplots of possible categorical predictors

On this page we will examine the relationship via boxplots of categorical predictors on our response variable, English. The potential categorical predictors are sex, behaviour and fluent, although the variables school and ravens might also be considered in a categorical sense as they take integer values.

h1 Words: 89

Remove

Input Questions (tick to include)

□ xvals

☑ filtercol

Remove

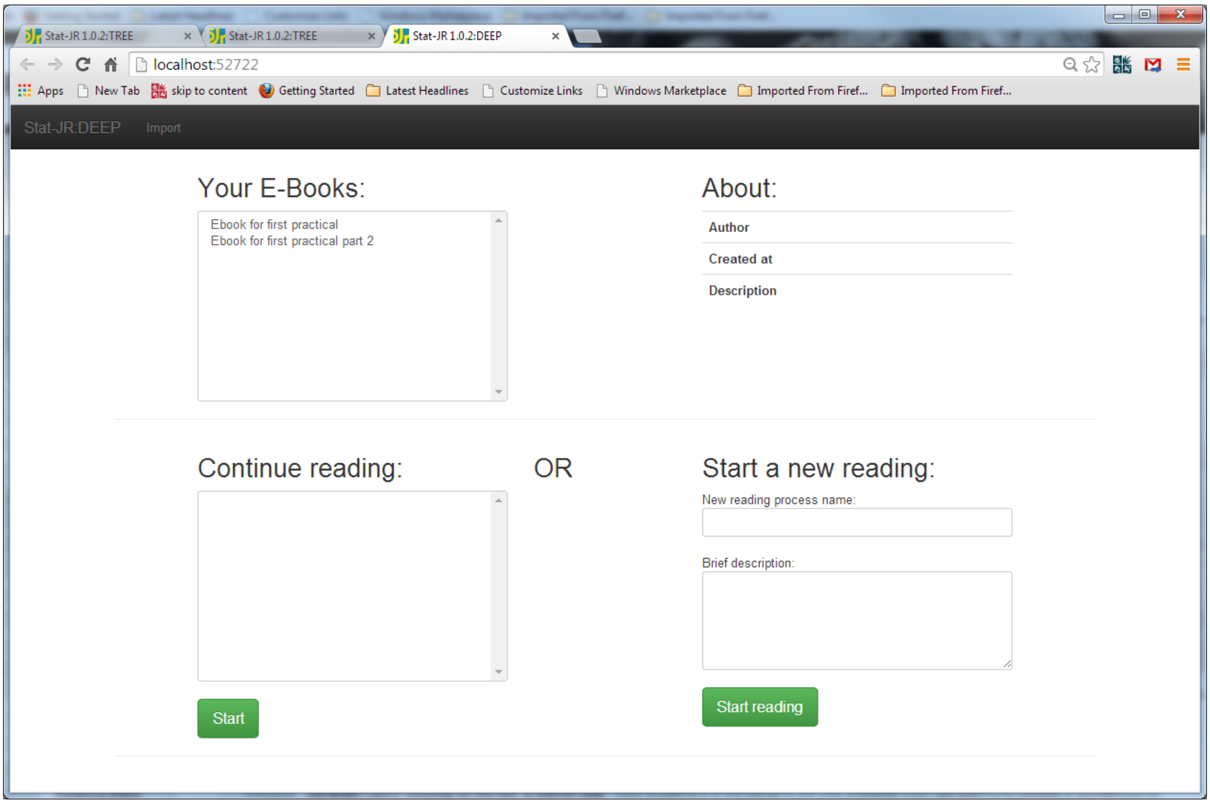
box.svg Remove

Here we put a tick next to **filtercol**, which happens to be the template input name for the categorical predictor; this means that the eBook user will be asked for this input when running the eBook. As we haven't put a tick next to **xvals**, which is the template input name for the variable to be plotted, this will be fixed at the value we choose in TREE, i.e. English.

For the purposes of this introduction we will end the eBook here and download it by clicking on the **Download as ebook** button. Here I will use the name **prac2.zip** for the file.

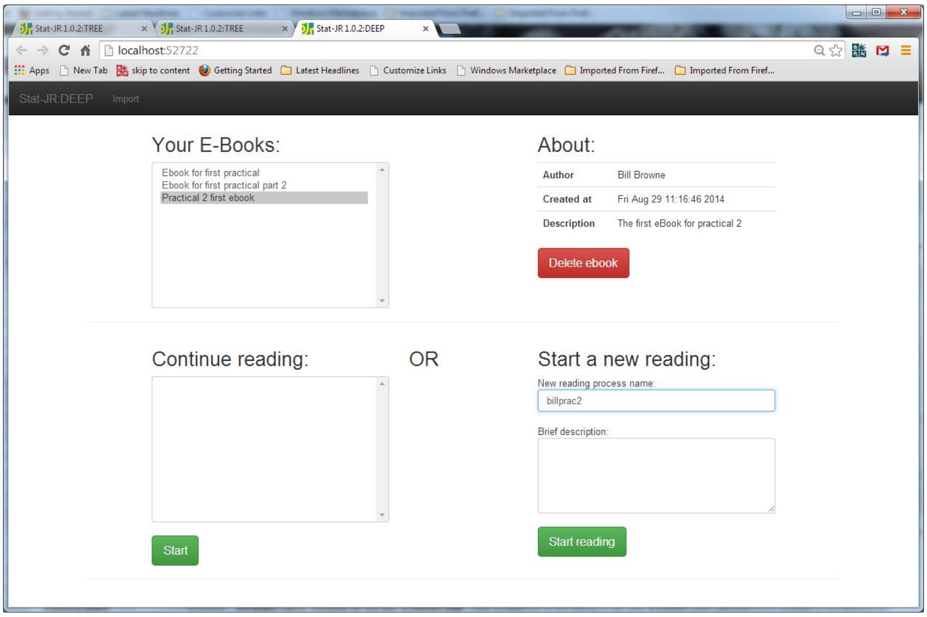
Reading in the Interactive eBook

We will next start up Stat-JR's DEEP interface so that we can look at the eBook we have constructed. This can be done by double-clicking on **deep.cmd** in the base directory of Stat-JR which will bring up another command window and the main DEEP window in the browser:



Ebook for first practical
Ebook for first practical part 2

Here I can see the ebooks that I viewed in the earlier practical listed under **Your E-Books** to the top left. We will need to import our new eBook so clicking on **Import**, via the black bar at the top, and then **Select an Ebook file** will allow us to pick **prac2.zip** from the list. We then continue uploading and our eBook list expands to include the new eBook. Next we click on the new eBook's name in the **Your E-Books** list and add a process name under **Start a new reading**, thus:



Your E-Books:

- Ebook for first practical
- Ebook for first practical part 2
- Practical 2 first ebook

About:

Author	Bill Browne
Created at	Fri Aug 29 11:16:46 2014
Description	The first eBook for practical 2

Continue reading:

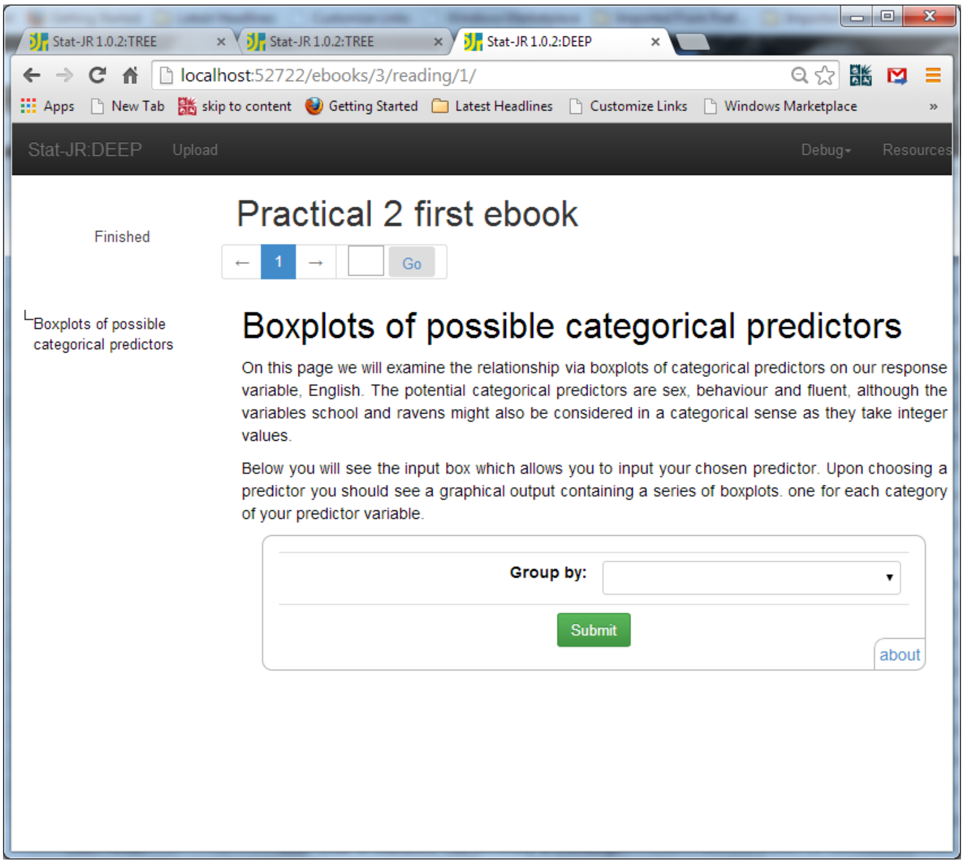
OR

Start a new reading:

New reading process name: billprac2

Brief description:

Finally we click on the **Start reading** button to start the eBook running and we are faced by the following one-page eBook:



Practical 2 first ebook

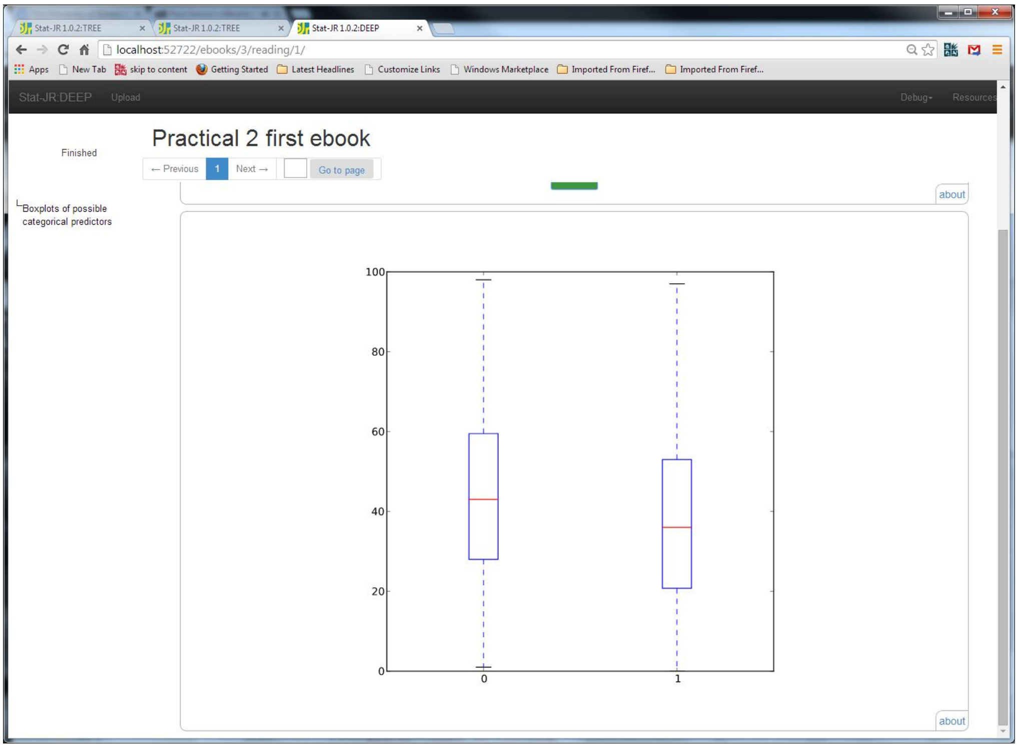
Boxplots of possible categorical predictors

On this page we will examine the relationship via boxplots of categorical predictors on our response variable, English. The potential categorical predictors are sex, behaviour and fluent, although the variables school and ravens might also be considered in a categorical sense as they take integer values.

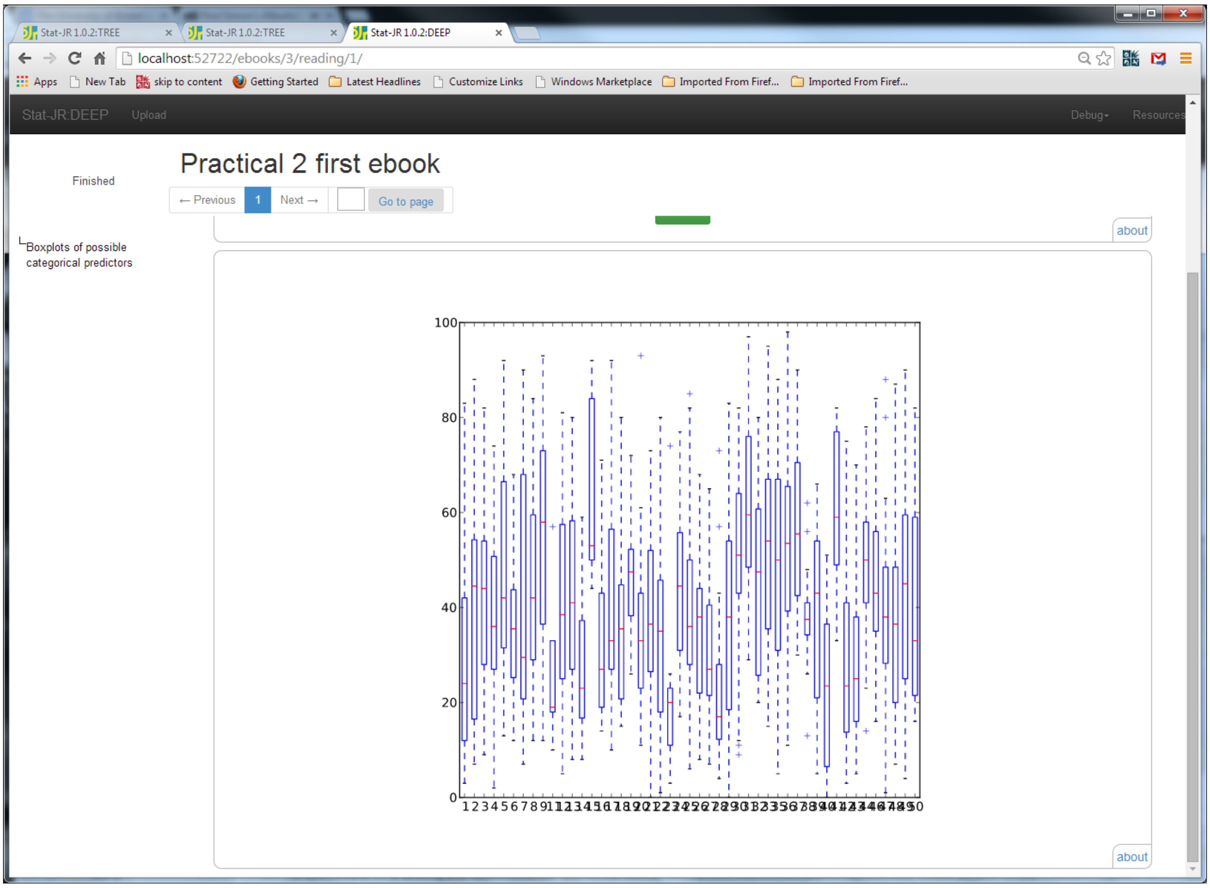
Below you will see the input box which allows you to input your chosen predictor. Upon choosing a predictor you should see a graphical output containing a series of boxplots. one for each category of your predictor variable.

Group by:

Here we see that we are asked for the input we ticked in TREE's eBook-writer. If we select **sex** from the pull-down list then click the **Submit** button we will see the same boxplot display that we saw within TREE:



Of course the eBook is now interactive so we can choose other options. For example if we choose **school**, and **Submit** again, we can get the rather busy boxplot displayed thus:



This should give you some idea of how we can bring interaction into an eBook. We could add further pages to this eBook, as we did in practical 1, but this one-page example should give the general idea of what is going on. We will therefore instead look under the bonnet at the underlying structure of the eBook so that we can see how the DEEP system knows when inputs are set etc. We will do this by first looking at this practical 2 eBook and then by returning to eBooks constructed in practical 1.

Structure of an eBook

An eBook is formed of a zipped folder containing several files. We will begin by looking a little at the file **prac2.zip** that we have constructed here in this practical. As it is a zip file a first step is to extract the files to a directory, say **prac2**, so that we can look at the file structure more closely. When you have done this you will find that the base directory contains one file **ebookdef.ttl** and a subdirectory **static**. The file **ebookdef.ttl** is the file that drives the ebook and contains many settings but is not meant to be modified by the eBook author and so we instead turn our attention to the **static** directory. Here you will find one file entitled **input0.py** and another subdirectory named **html**. If we had run several templates in our ebook there would in fact be several files in this directory each beginning with the string **input** and numbered (starting from 0) to correspond with the order that the templates are used in the eBook. We can open the file **input0.py** in a text editor (e.g. Wordpad) and we will find it contains a list of name-value pairs enclosed in curly brackets –in this case there is just one pair as we have set one input as having a fixed value. The text in **input0.py** for my eBook is as follows:

```
{'xvals': 'english'}
```

This is the input that we have fixed: i.e. the continuous variable we wish to plot in all our boxplots. As we are asking the user for the other input, namely the grouping variable, it does not appear in this file. If we were however to add it to this list (saving the file and zipping it up again), we would make our eBook static as all inputs would be pre-specified. We will revisit inputs files a little later when considering the eBook from practical 1, however in the meantime if we instead look at the subdirectory **html** we will find it contains one file called **ebookpages.html**.

If we open this file in a text editor we see that it contains most of the content seen on the screen when running the eBook, i.e. the content we typed-in using TREE's eBook-writer. If you are familiar with html then you will likely recognise some of the tags here:

```
<div id ="activityregion0" class="deep_activityregion">

        <div id="ebookpage1" class="deep_page">

                <h1>Boxplots of possible categorical predictors&nbsp;</h1>
<p>On this page we will examine the relationship via boxplots of categorical predictors on our
response variable, English. The potential categorical predictors are sex, behaviour and
fluent, although the variables school and ravens might also be considered in a categorical
sense as they take integer values.</p>
<p>Below you will see the input box which allows you to input your chosen predictor. Upon
choosing a predictor you should see a graphical output containing a series of boxplots. one
for each category of your predictor variable.</p>

                <div data-deep-id ="template0-inputq" class="deep_dynamic_output
deep_dynamic_hidden" data-deep-only="filtercol,_submit"></div>

                <div class="deep_textarea deep_dynamic_hidden">
                        <div data-deep-id="template0-box.svg" class="deep_dynamic_output
deep_dynamic_hidden"></div>
                </div>
        </div>
```

```
</div>
```

We described a little bit of the html syntax in the lecture preceding this practical and so will not reiterate it here. We will notice, however, that lots of the text is embedded within **div** tags. These tags separate out a division, or section, and are used by Stat-JR to identify both activity regions and pages within them, and also specify input and output objects via `div data-deep-id`. We are particularly interested in the line:

```
<div data-deep-id ="template0-inputq" class="deep_dynamic_output deep_dynamic_hidden" data-deep-only="filtercol,_submit"></div>
```

as this corresponds to the input question box for template0, and below we will need to add similar lines to our first eBook in order to tell DEEP where to place input questions. At this stage we will not alter the eBook from practical 2 but instead return to the final eBook from practical 1 (which in my example I stored as **jsp2.zip**).

Modifying an eBook

We will take our final eBook from practical one and unzip it. Here we again find the file **ebookdef.ttl** and a **static** directory but this time within the **static** directory there are two input files: **input0.py** that contains the inputs for the histogram and **input1.py** that contains the inputs for the model fitting. We will look at **input0.py** first:

```
{'vals': 'english', 'bins': '10'}
```

Here we see the two pre-specified inputs for the histogram, namely which variable to plot and how many bins. We will edit this so that the number of bins aren't specified (alternatively you could choose to not specify the variable (*vals*), or indeed delete both inputs – just leaving the curly brackets) and thus change the text as follows:

```
{'vals': 'english'}
```

Once you have done that, save the file.

We will next turn our attention to the file **input1.py**:

```
{'burnin': '500', 'defaultsv': 'Yes', 'thinning': '1', 'outdata': 'out', 'nchains': '3',
'defaultalg': 'Yes', 'iterations': '2000', 'y': 'english', 'x': 'cons,sex', 'seed': '1',
'makepred': 'Yes'}
```

Here we see that there are more inputs for the model fitting template than there were for the histogram template. Although most of the inputs here have names that are easy to match to the input questions in TREE, to find out precisely which input is which one could look at the template code for this template and we will discuss templates more in the next practical. For now we will simply remove the input for the predictor variables, **x**, thus leaving the file as follows:

```
{'burnin': '500', 'defaultsv': 'Yes', 'thinning': '1', 'outdata': 'out', 'nchains': '3',
'defaultalg': 'Yes', 'iterations': '2000', 'y': 'english', 'seed': '1', 'makepred': 'Yes'}
```

Once you've made this change, save the file.

Making these small changes and zipping up the eBook once more would be enough to make the eBook interactive if we had included the input boxes when using the eBook-writer in TREE; however, since we neglected to, we need to add them to the html file. Thus if we go into the **html** directory we can find the **ebookpages.html** file that contains the content of the eBook. In my example, opening this in a text editor I see the following:

```
 <div id ="activityregion0" class="deep_activityregion">


        <div id="ebookpage1" class="deep_page">

                <h1>Data Summary&nbsp;</h1>
<p>We will start our eBook by giving a summary of the dataset. <span style="color:
#ff0000;"><em>This text is red and italic.</em></span></p>

                <div data-deep-id ="template0-dataset-summary" class="deep_dynamic_output
deep_dynamic_hidden"></div>
        </div>

        <div id="ebookpage2" class="deep_page">

                <h1>Histogram</h1>
<p>Here we will firstly draw a histogram of the response variable, english.&nbsp;</p>
<p>Firstly you will see the template inputs and then the histogram itself</p>

                <div data-deep-id ="template0-input" class="deep_dynamic_output
deep_dynamic_hidden"></div>

                <div class="deep_textarea deep_dynamic_hidden">
                        <div data-deep-id="template0-histogram.svg" class="deep_dynamic_output
deep_dynamic_hidden"></div>
                </div>
        </div>
</div>
<div id ="activityregion1" class="deep_activityregion">

        <div id="ebookpage3" class="deep_page">

                <h1>Fitting a model to investigate gender differences</h1>
<p>In this activity region we have used the regression1 template to look at how sex affects
english. The inputs we use are as follows:</p>

                <div data-deep-id ="template1-input" class="deep_dynamic_output
deep_dynamic_hidden"></div>

                <h2>Equations&nbsp;</h2>
<p>Below are the equations for the model we are fitting&nbsp;</p>

                <div class="deep_textarea deep_dynamic_hidden">
                        <div data-deep-id="template1-equation.tex" class="deep_dynamic_output
deep_dynamic_hidden"></div>
                </div>
        </div>

        <div id="ebookpage4" class="deep_page">

                <h2>Algorithm</h2>
<p>This model is fitted using MCMC and the algorithm can be seen below:</p>

                <div class="deep_textarea deep_dynamic_hidden">
                        <div data-deep-id="template1-algorithm.tex" class="deep_dynamic_output
deep_dynamic_hidden"></div>
                </div>
        </div>

        <div id="ebookpage5" class="deep_page">

                <h2>Results</h2>
<p>Below are the model parameter estimates for this model</p>

                <div class="deep_textarea deep_dynamic_hidden">
                        <div data-deep-id="template1-ModelParameters"
class="deep_dynamic_output deep_dynamic_hidden"></div>
                </div>

                <p>and here you will see the model fit statistics</p>
                <div class="deep_textarea deep_dynamic_hidden">
                        <div data-deep-id="template1-ModelFit" class="deep_dynamic_output
deep_dynamic_hidden"></div>
                </div>
        </div>

        <div id="ebookpage6" class="deep_page">

                <h2>MCMC Diagnostics</h2>
<p>Below we can see diagnostics plots for both beta_0 and beta_1</p>

                <div class="deep_textarea deep_dynamic_hidden">
                        <div data-deep-id="template1-beta_0.svg" class="deep_dynamic_output
deep_dynamic_hidden"></div>
                </div>

                <div class="deep_textarea deep_dynamic_hidden">
                        <div data-deep-id="template1-beta_1.svg" class="deep_dynamic_output
deep_dynamic_hidden"></div>
                </div>
        </div>
</div>
```

If you look at the details of the above code you should see obvious patterns and by comparing what you see to the pages of the eBook in practical 1 (as rendered in the browser when reading it via the DEEP interface) it should become clearer as to what certain lines are doing. In the original eBook we included a box which writes-out the selected inputs, and these correspond to the lines:

```
<div data-deep-id ="template0-input" class="deep_dynamic_output deep_dynamic_hidden"></div>
```

and

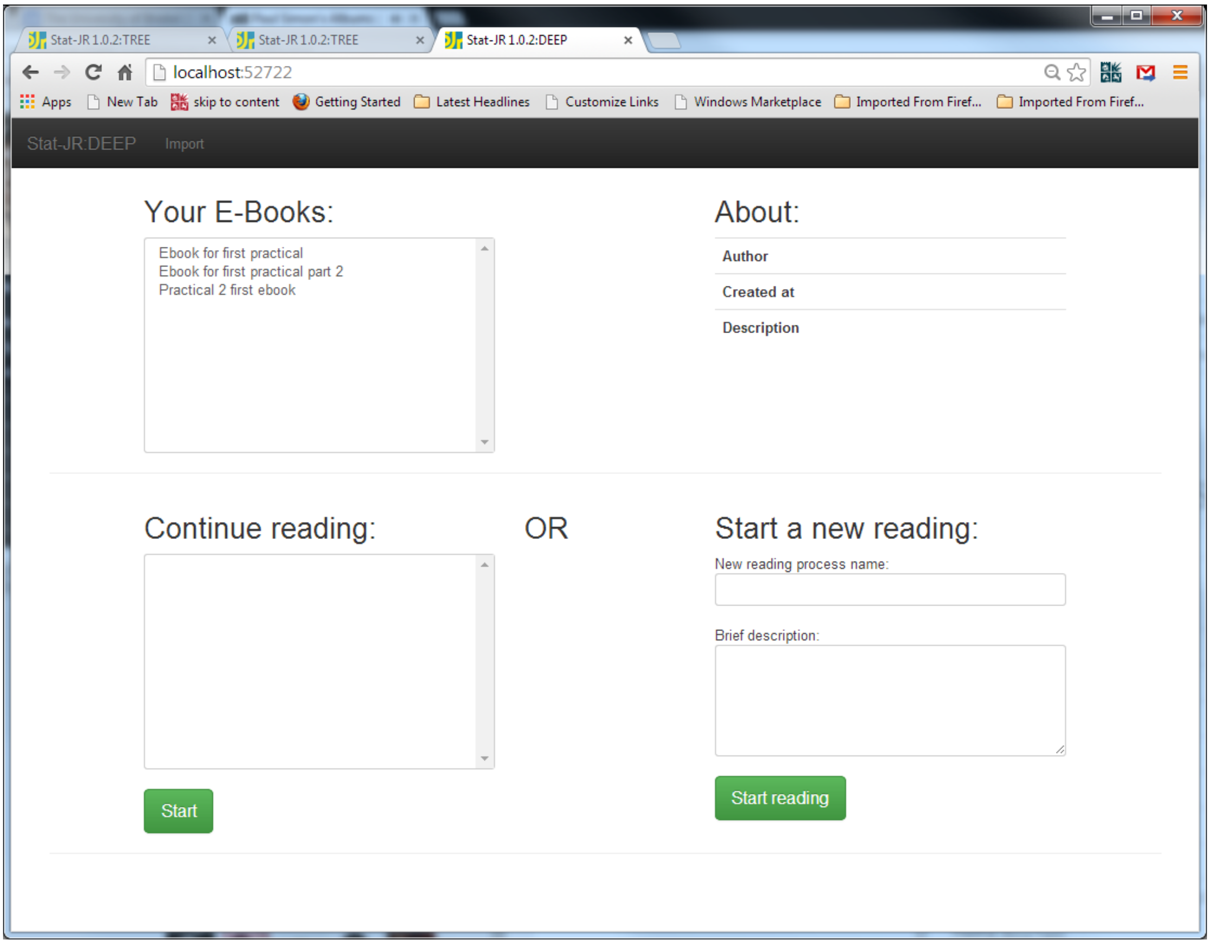
```
<div data-deep-id ="template1-input" class="deep_dynamic_output deep_dynamic_hidden"></div>
```

In fact to convert these lines so that we instead *ask* for the inputs, we can simply change the string **input** to **inputq** for each **data-deep-id** and then save the file (i.e. change **template0-input** to **template0-inputq** and **template1-input** to **template1-inputq**).

We next need to zip up the eBook. To do this ensuring we keep the same file structure we need to move up to the base directory containing the **ebookdef.ttl** file. We then highlight both it and the **static** directory and zip them up. This will create a zip file in this directory which I would suggest you name to something different e.g. in my case **jsp2new.zip** and move to another directory.

Examining our changed eBook

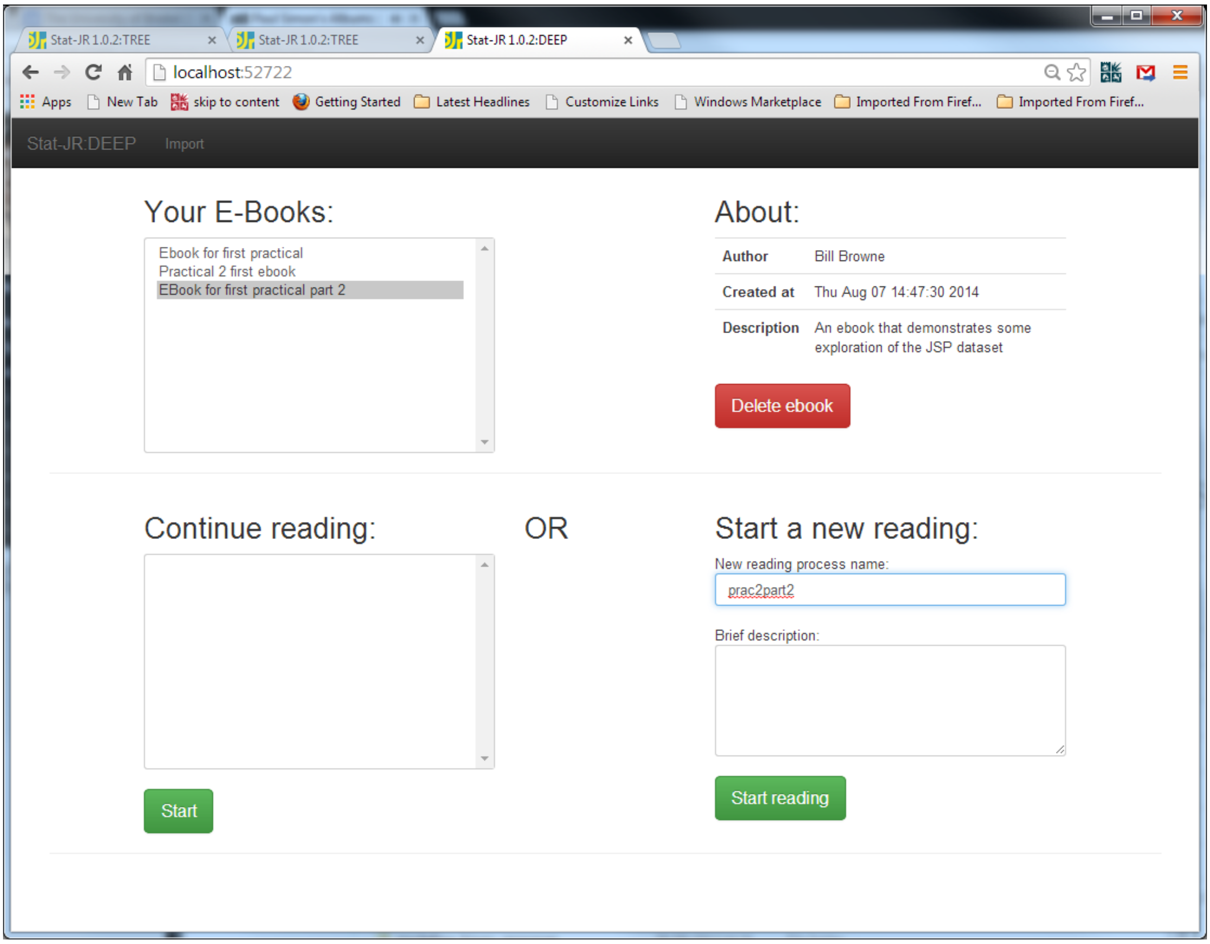
We will now return to DEEP and look at the eBook we have constructed. If you still have DEEP open from earlier in this practical then clicking on **Stat-JR:DEEP** in the header bar will bring you back to the main screen which will look thus:



Ebook for first practical
Ebook for first practical part 2
Practical 2 first ebook

You will see that we have previously imported the **Ebook for first practical part 2** that we have been modifying but this is the old zip file and so we will need to first delete this before importing our new modified eBook. Click on the eBook in the list and click on the **Delete eBook** button that appears and then the **OK** button.

The eBook will disappear from our list and we can them import the new zip file for the eBook as before and select it as below:



Your E-Books:

Ebook for first practical
Practical 2 first ebook
EBook for first practical part 2

About:

Author	Bill Browne
Created at	Thu Aug 07 14:47:30 2014
Description	An ebook that demonstrates some exploration of the JSP dataset

Continue reading:

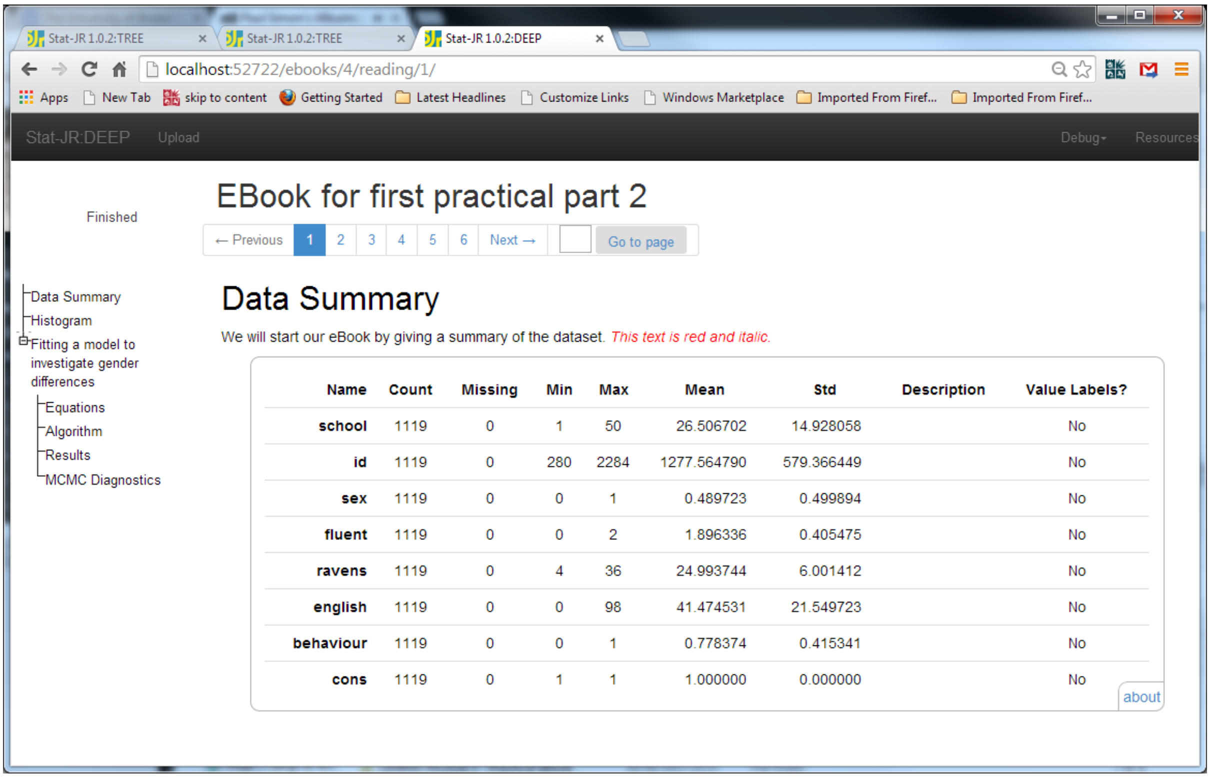
OR

Start a new reading:

New reading process name: prac2part2

Brief description:

After clicking on **Start reading** we are greeted with the initial page of the eBook which simply contains the data summary and hasn't changed:



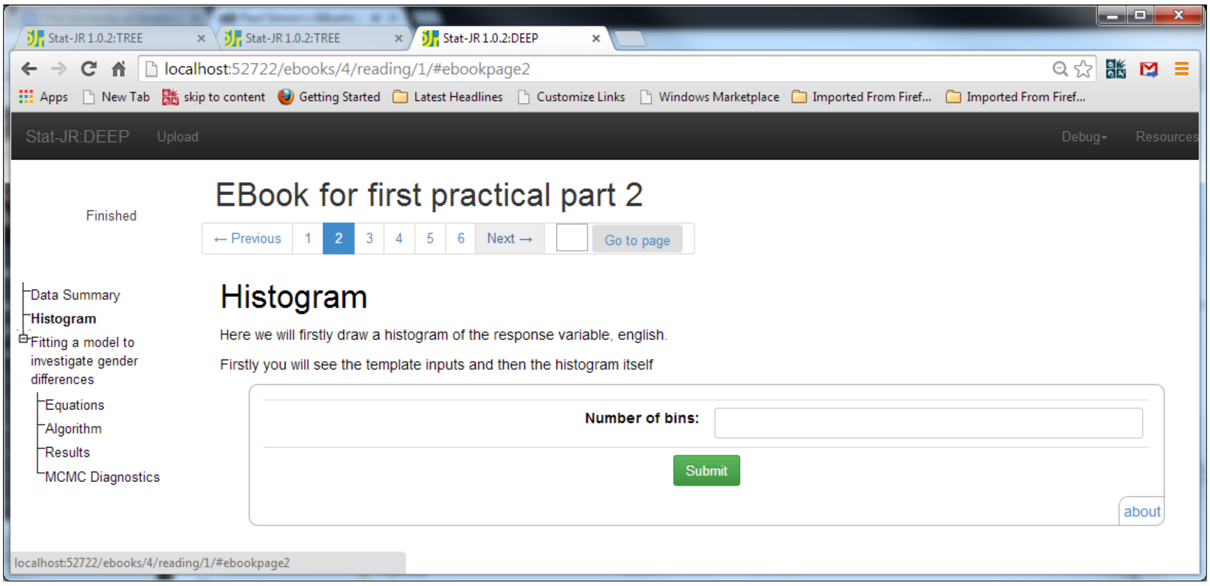
EBook for first practical part 2

Data Summary

We will start our eBook by giving a summary of the dataset. *This text is red and italic.*

Name	Count	Missing	Min	Max	Mean	Std	Description	Value Labels?
school	1119	0	1	50	26.506702	14.928058		No
id	1119	0	280	2284	1277.564790	579.366449		No
sex	1119	0	0	1	0.489723	0.499894		No
fluent	1119	0	0	2	1.896336	0.405475		No
ravens	1119	0	4	36	24.993744	6.001412		No
english	1119	0	0	98	41.474531	21.549723		No
behaviour	1119	0	0	1	0.778374	0.415341		No
cons	1119	0	1	1	1.000000	0.000000		No

Clicking on page 2 we now see the histogram screen and also the input box for the number of bins.



EBook for first practical part 2

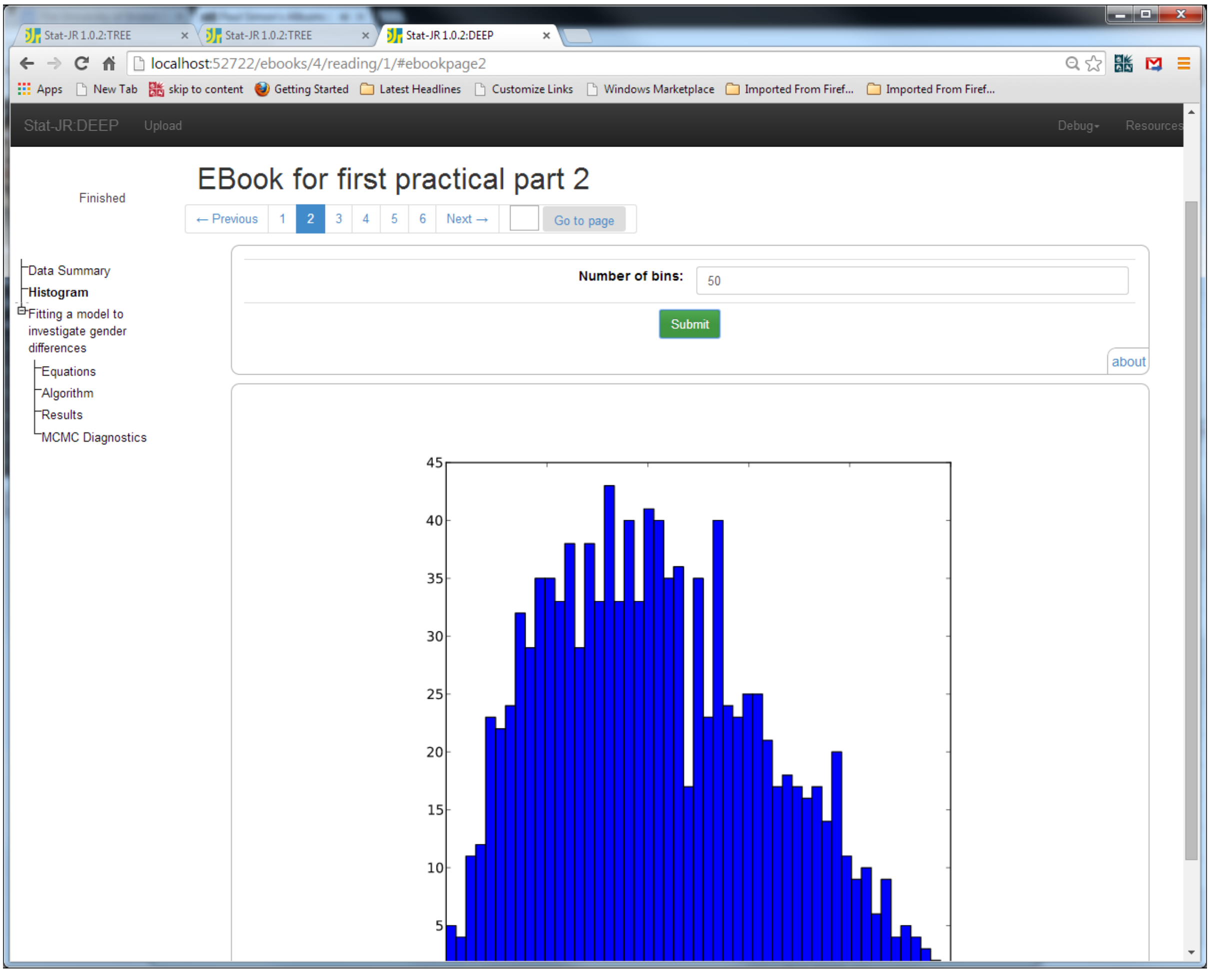
Histogram

Here we will firstly draw a histogram of the response variable, english.

Firstly you will see the template inputs and then the histogram itself

Number of bins:

The text above now isn't quite as intuitive as we don't see the template inputs, but we could have easily changed this text in the html file directly if we wished. If we input **50** for number of bins and click Submit we can see the resulting histogram:

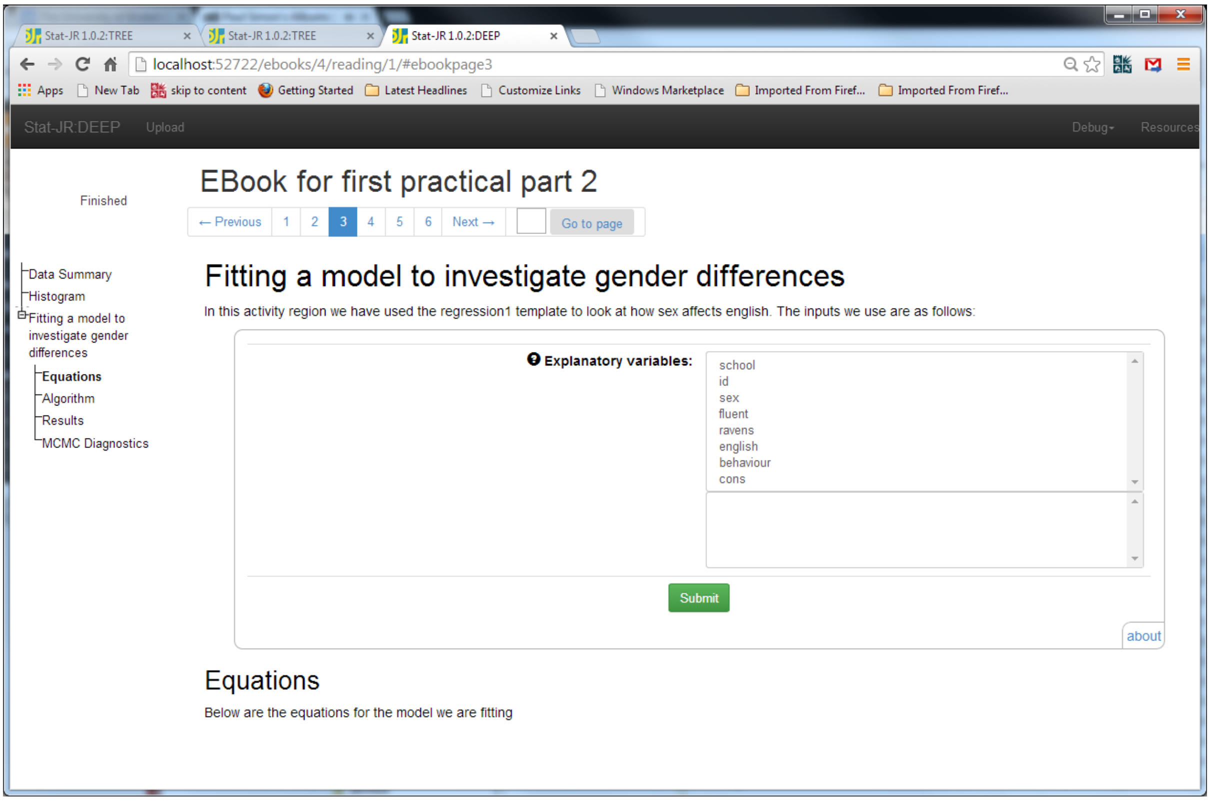


EBook for first practical part 2

Number of bins: 50

Demonstrating that we can now clearly change the number of bins interactively.

If we move on to the third page we see the following:



EBook for first practical part 2

Finished

← Previous 1 2 3 4 5 6 Next → Go to page

- Data Summary
- Histogram
- Fitting a model to investigate gender differences
  - Equations
  - Algorithm
  - Results
  - MCMC Diagnostics

Fitting a model to investigate gender differences

In this activity region we have used the regression1 template to look at how sex affects english. The inputs we use are as follows:

Explanatory variables:

school
id
sex
fluent
ravens
english
behaviour
cons

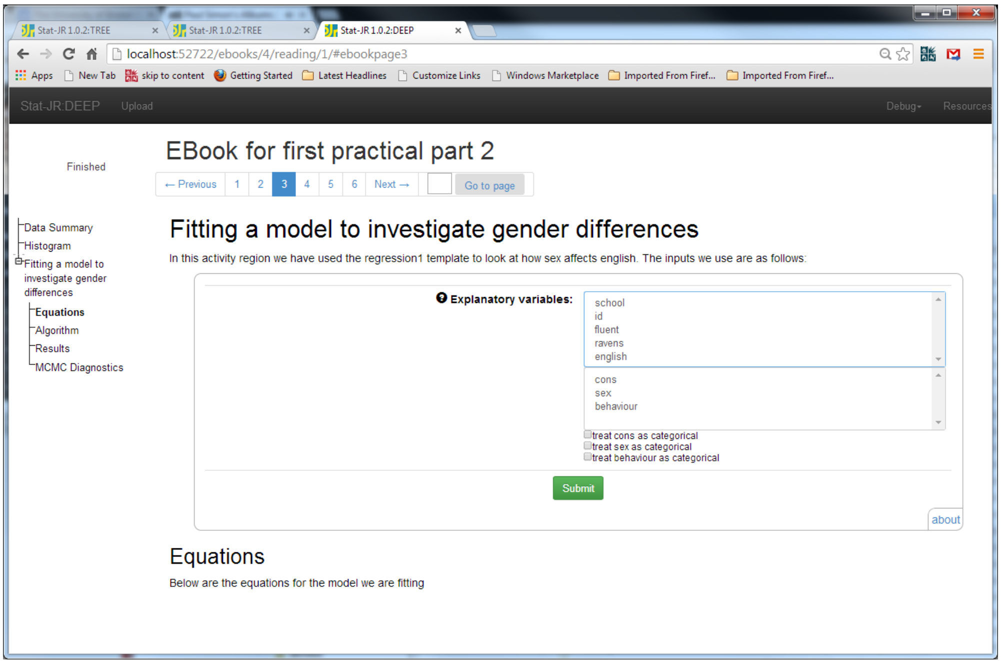
Submit

about

Equations

Below are the equations for the model we are fitting

Here once again we see the inputs that are still required, namely the explanatory (predictor) variables. Again some of the text is a little out of kilter as we do not yet see equations, as inputs have not been set. It is possible to hide this text until the equations are ready but we will not explain how to here. We will now choose some explanatory variables (remembering to indicate if any are categorical by ticking the relevant box); for my example I chose the following:



EBook for first practical part 2

Finished

← Previous 1 2 3 4 5 6 Next → Go to page

- Data Summary
- Histogram
- Fitting a model to investigate gender differences
  - Equations
  - Algorithm
  - Results
  - MCMC Diagnostics

Fitting a model to investigate gender differences

In this activity region we have used the regression1 template to look at how sex affects english. The inputs we use are as follows:

Explanatory variables:

school
id
fluent
ravens
english

cons
sex
behaviour

□ treat cons as categorical
□ treat sex as categorical
□ treat behaviour as categorical

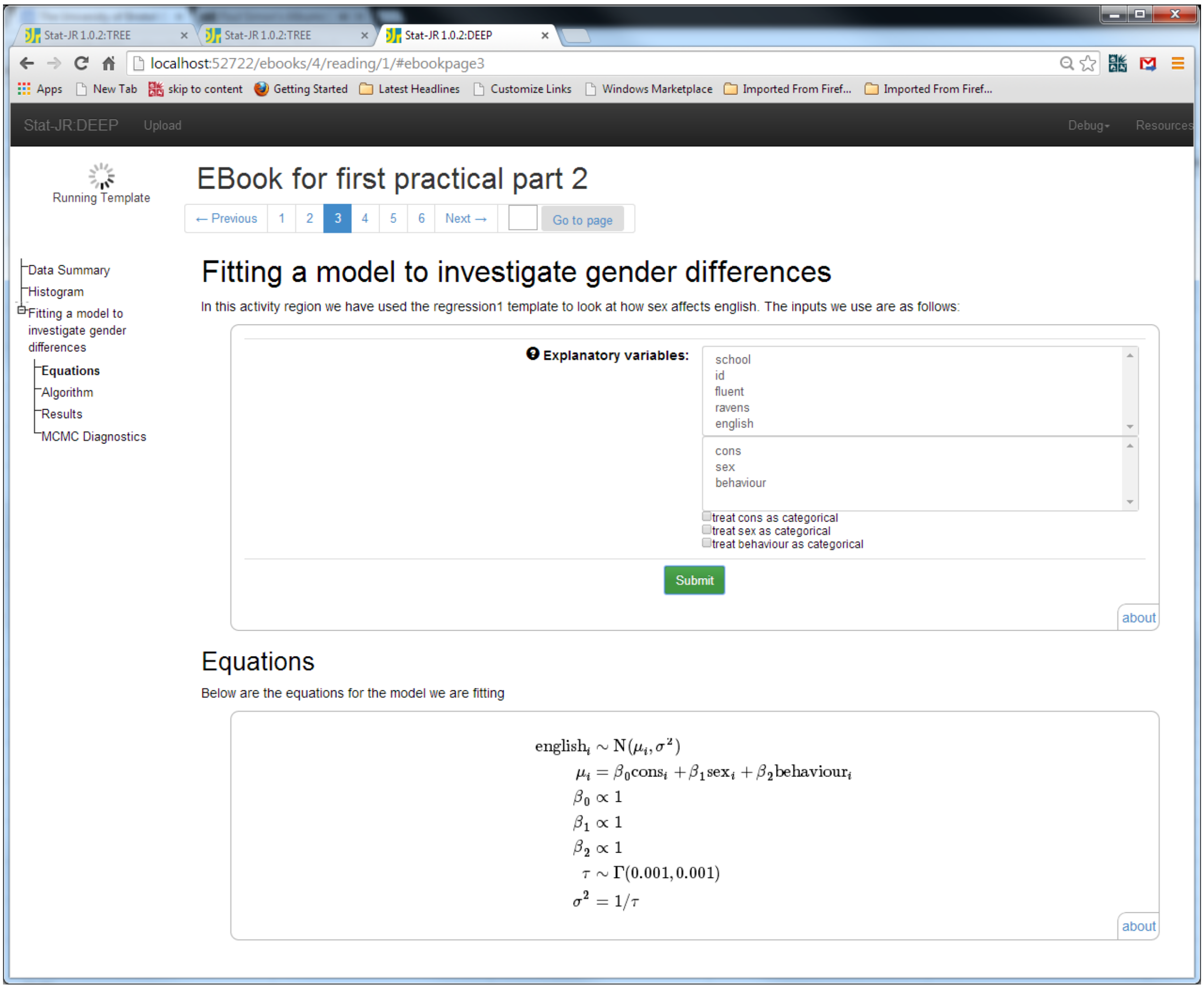
Submit

about

Equations

Below are the equations for the model we are fitting

Here I'm running a different model which also includes the second predictor **behaviour**. Clicking on the **Submit** button you will need to wait for the model to fit, but once it has done so the remaining pages will be filled with the output from this new model-fit. The first output to appear will be the equations on the same page:



EBook for first practical part 2

Fitting a model to investigate gender differences

In this activity region we have used the regression1 template to look at how sex affects english. The inputs we use are as follows:

Explanatory variables:

school
id
fluent
ravens
english

cons
sex
behaviour

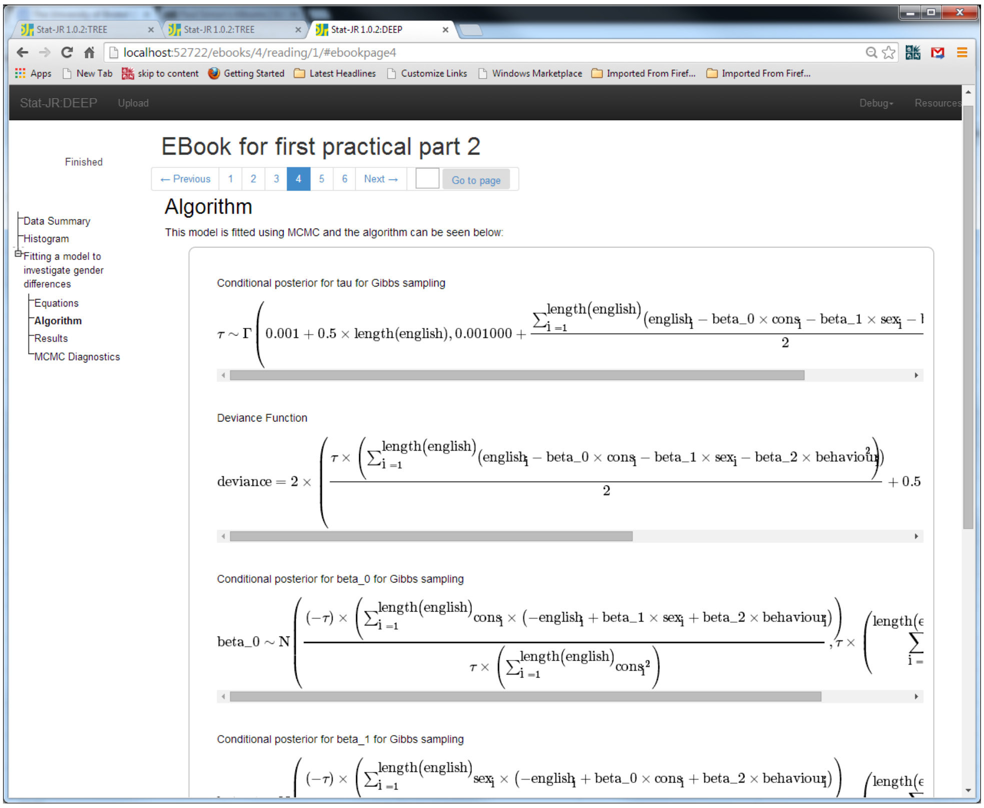
□ treat cons as categorical
□ treat sex as categorical
□ treat behaviour as categorical

Equations

Below are the equations for the model we are fitting

$$
\begin{aligned}
\mathrm{english}_i &\sim \mathrm{N}(\mu_i, \sigma^2) \\
\mu_i &= \beta_0 \mathrm{cons}_i + \beta_1 \mathrm{sex}_i + \beta_2 \mathrm{behaviour}_i \\
\beta_0 &\propto 1 \\
\beta_1 &\propto 1 \\
\beta_2 &\propto 1 \\
\tau &\sim \Gamma(0.001, 0.001) \\
\sigma^2 &= 1/\tau
\end{aligned}
$$

The equations will actually appear prior to the model-fit finishing, since they're not contingent on the results of the model-fit, only on the user's choice of explanatory variables. Once the status area in the top-left says it has finished we can move to page 4 we see the new algorithm (in fact this also appears prior to model fit):



EBook for first practical part 2

Algorithm

This model is fitted using MCMC and the algorithm can be seen below:

Conditional posterior for tau for Gibbs sampling

$$\tau \sim \Gamma\left(0.001 + 0.5 \times \text{length(english)}, 0.001000 + \frac{\sum_{i=1}^{\text{length(english)}} (\text{english}_i - \text{beta\_0} \times \text{cons}_i - \text{beta\_1} \times \text{sex}_i - \text{l}}{2}\right.$$

Deviance Function

$$\text{deviance} = 2 \times \left(\frac{\tau \times \left(\sum_{i=1}^{\text{length(english)}} (\text{english}_i - \text{beta\_0} \times \text{cons}_i - \text{beta\_1} \times \text{sex}_i - \text{beta\_2} \times \text{behaviour}_i)^2\right)}{2} + 0.5\right.$$

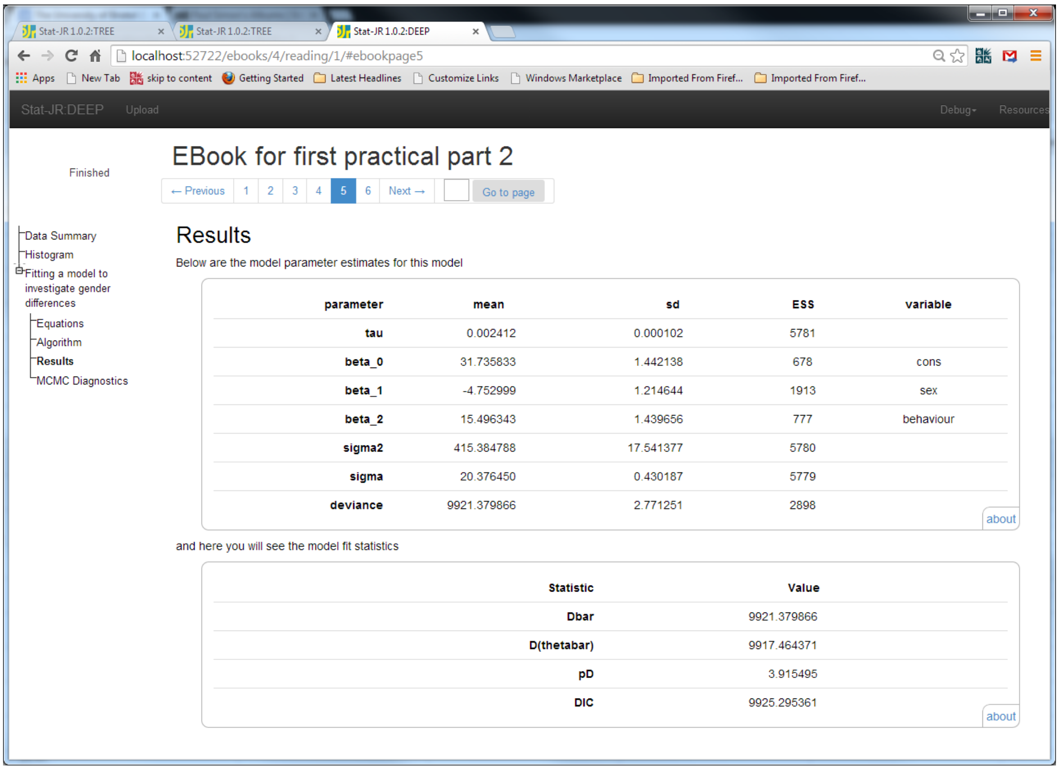
Conditional posterior for beta_0 for Gibbs sampling

$$\text{beta\_0} \sim \text{N}\left(\frac{(-\tau) \times \left(\sum_{i=1}^{\text{length(english)}} \text{cons}_i \times (-\text{english}_i + \text{beta\_1} \times \text{sex}_i + \text{beta\_2} \times \text{behaviour}_i)\right)}{\tau \times \left(\sum_{i=1}^{\text{length(english)}} \text{cons}_i^2\right)}, \tau \times \left(\sum_{i=}^{\text{length(e}}\right.\right.$$

Conditional posterior for beta_1 for Gibbs sampling

$$\left(\frac{(-\tau) \times \left(\sum_{i=1}^{\text{length(english)}} \text{sex}_i \times (-\text{english}_i + \text{beta\_0} \times \text{cons}_i + \text{beta\_2} \times \text{behaviour}_i)\right)}{}\right. \left(\text{length(e}\right.$$

Moving onto the next page (5) we get the new results and see that, in my example, the behaviour predictor (**beta_2**) has a strong positive effect. Indeed if I compare the DIC diagnostic (which is a measure of model fit for MCMC estimation) from the previous model I see it has reduced, indicating the current model is a better one:



EBook for first practical part 2

Results

Below are the model parameter estimates for this model

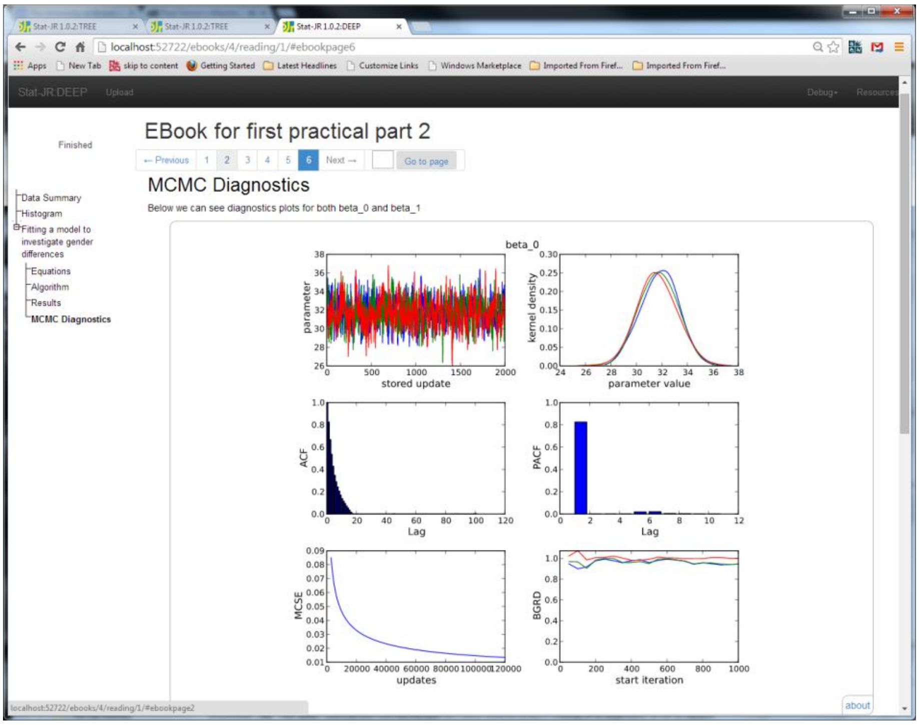
parameter	mean	sd	ESS	variable
tau	0.002412	0.000102	5781	
beta_0	31.735833	1.442138	678	cons
beta_1	-4.752999	1.214644	1913	sex
beta_2	15.496343	1.439656	777	behaviour
sigma2	415.384788	17.541377	5780	
sigma	20.376450	0.430187	5779	
deviance	9921.379866	2.771251	2898	

and here you will see the model fit statistics

Statistic	Value
Dbar	9921.379866
D(thetabar)	9917.464371
pD	3.915495
DIC	9925.295361

We can now use this eBook (pages 3-6) to try out several different models if we wish and find the best (in some sense) model for **english** from the available predictors.

Finally on page 6 we get the MCMC diagnostics for parameters **beta_0** and **beta_1**:



EBook for first practical part 2

MCMC Diagnostics

Below we can see diagnostics plots for both beta_0 and beta_1

Note that although we have 3 predictors now, we still only have diagnostics for the first two displayed in the eBook as that was what we explicitly included when first authoring it (although the other graphs are available via **Resources**, accessible either via the link in the top-right of the window, or by pressing **about** in the bottom-right of one of the dynamic objects). We can of course add more outputs via the file **ebookpages.html** and in fact if we had chosen a simpler model with just an intercept then only **beta_0**'s graph would appear here. This does throw up some issues as to how we write eBooks to cater for all possible (sensible) inputs. We will see that the present incarnation of DEEP is by no means perfect and one issue is the lack of an easy way to pass information from template to template within an eBook to make the eBook into a sequential story rather than a series of stand-alone sections. We will look at methods that deal with this issue in practical 3 however we will finish this practical with a short section on changing the look of an eBook.

Adding multiple columns to the boxplot eBook

When we use the eBook writer in TREE the eBooks that are produced have a standard format and are somewhat limited in terms of the features available within the html that we use. We have implemented some extra functionality in terms of the look and feel of the eBooks within the DEEP system and these are accessed with extra *div* tags. Here we will simply investigate one possible layout tag namely the *div class="deep-twocol"*. This class (as explained in the lecture) will allow us to display features in two columns as opposed to the current linear layout.

As a simple demonstration of this feature we will return to the eBook we constructed in the beginning of this practical that allowed us to view boxplots of our dataset. To modify this eBook we will firstly need to unzip the files stored within **prac2.zip** (although in fact you should have these files already unzipped from earlier)**.** If we do this we will observe the by now familiar eBook structure of a base directory containing the file **ebookdef.ttl** and the **static** directory which itself contains an inputs file (**input0.py**) and the **html** directory which contains the file **ebookpages.html** which we wish to modify.

In this case the file **ebookpages.html** looks as follows:

```
<div id ="activityregion0" class="deep_activityregion">

        <div id="ebookpage1" class="deep_page">

                <h1>Boxplots of possible categorical predictors&nbsp;</h1>
<p>On this page we will examine the relationship via boxplots of categorical predictors on our
response variable, English. The potential categorical predictors are sex, behaviour and
fluent, although the variables school and ravens might also be considered in a categorical
sense as they take integer values.</p>
<p>Below you will see the input box which allows you to input your chosen predictor. Upon
choosing a predictor you should see a graphical output containing a series of boxplots. one
for each category of your predictor variable.</p>

                <div data-deep-id ="template0-inputq" class="deep_dynamic_output
deep_dynamic_hidden" data-deep-only="filtercol,_submit"></div>

                <div class="deep_textarea deep_dynamic_hidden">
                        <div data-deep-id="template0-box.svg" class="deep_dynamic_output
deep_dynamic_hidden"></div>
                </div>
        </div>
</div>
```

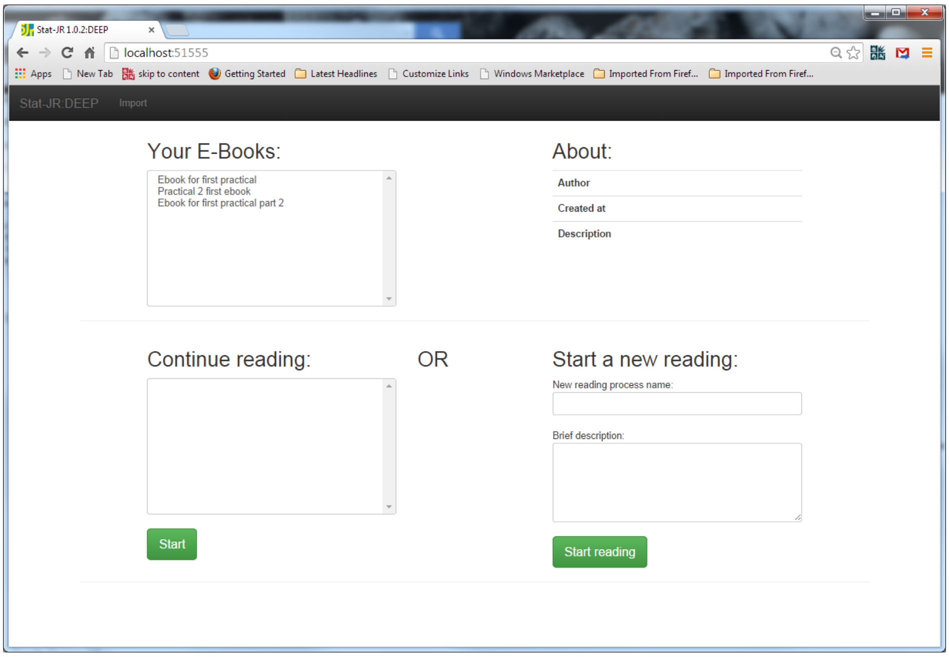
Here you see the one activity region containing one page that are specified by the first two **<div>** tags (matched with end **</div>** tags). Within this page we then see the heading within the pairing of **<h1>** and **</h1>** tags followed by the textual information split between two paragraphs within the pairings of **<p>** and **</p>** tags. Finally we then have the input question **div** and the output object **div**. What we'd like to do is have the output object i.e. the boxplot in a second column to the right of both the text and the questions. To do this we have to place the material that we wish to appear in each column within new **div** and **/div** tags with the whole section placed within a pairing of **<div class="deep-twocol">** and **</div>** lines. The modified code is below for inspection with the changes highlighted:

```
<div id ="activityregion0" class="deep_activityregion">

        <div id="ebookpage1" class="deep_page">

                <h1>Boxplots of possible categorical predictors&nbsp;</h1>
           <div class="deep-twocol">
              <div>
<p>On this page we will examine the relationship via boxplots of categorical predictors on our
response variable, English. The potential categorical predictors are sex, behaviour and
fluent, although the variables school and ravens might also be considered in a categorical
sense as they take integer values.</p>
<p>Below you will see the input box which allows you to input your chosen predictor. Upon
choosing a predictor you should see a graphical output containing a series of boxplots. one
for each category of your predictor variable.</p>

                        <div data-deep-id ="template0-inputq" class="deep_dynamic_output
deep_dynamic_hidden" data-deep-only="filtercol,_submit"></div>
              </div>
              <div>
                        <div class="deep_textarea deep_dynamic_hidden">
                              <div data-deep-id="template0-box.svg" class="deep_dynamic_output
deep_dynamic_hidden"></div>
                        </div>
              </div>
           </div>
        </div>
```

```
</div>
```

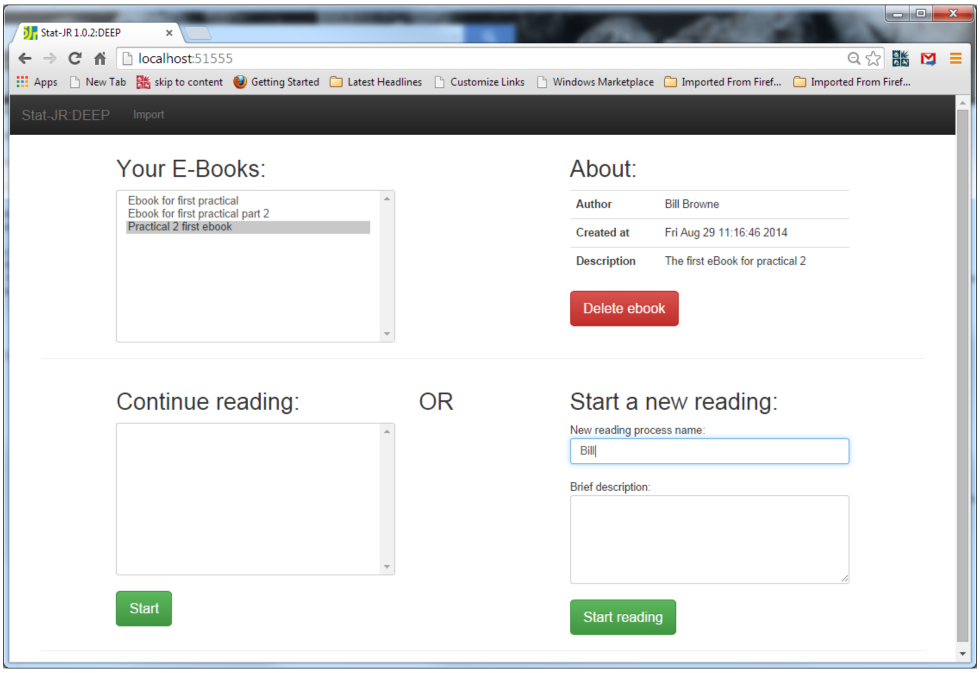
Once you have made such changes then the **ebookpages.html** file needs saving and the eBook needs zipping up which one does from the base directory.

If we return to the DEEP system main screen we will have several eBooks in our list:



Ebook for first practical
Practical 2 first ebook
Ebook for first practical part 2

We will again need to delete the existing version of the eBook from the list so click on **Practical 2 first ebook** from the list and click on the **Delete eBook** button. The eBook will vanish from the list and we next need to **Import** the new zip file we have created for this eBook which will then appear at the bottom of the list from where we highlight it and give a reading process name and click on **Start reading**:

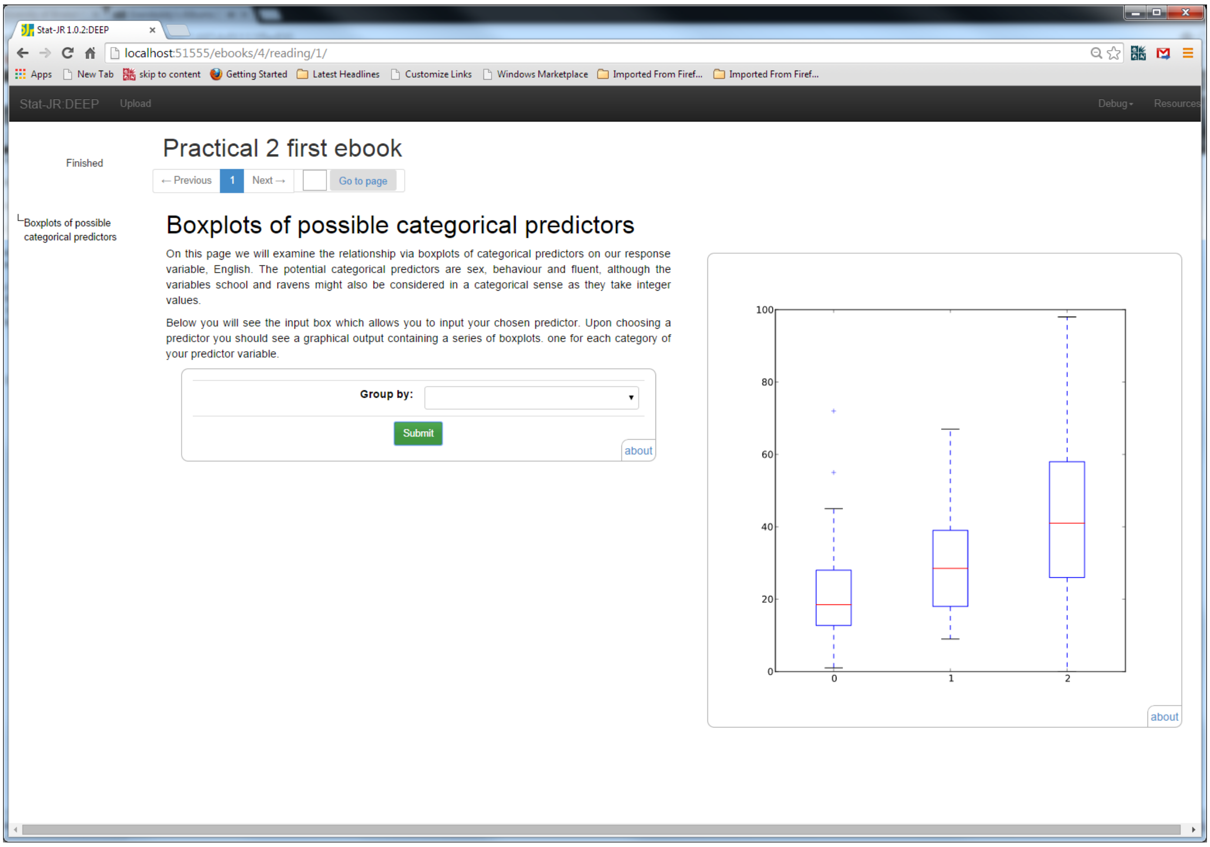


Ebook for first practical
Ebook for first practical part 2
Practical 2 first ebook

Author	Bill Browne
Created at	Fri Aug 29 11:16:46 2014
Description	The first eBook for practical 2

New reading process name: Bill

If we do this we will see the text and input question for the eBook is positioned to the left of the page. Choosing a categorical predictor (for example **fluent**) will then create the plot of the variable to the right of the screen thus:



Practical 2 first ebook

Boxplots of possible categorical predictors

On this page we will examine the relationship via boxplots of categorical predictors on our response variable, English. The potential categorical predictors are sex, behaviour and fluent, although the variables school and ravens might also be considered in a categorical sense as they take integer values.

Below you will see the input box which allows you to input your chosen predictor. Upon choosing a predictor you should see a graphical output containing a series of boxplots. one for each category of your predictor variable.

Group by:

We will finish this practical here but feel free to experiment with the two-columns environment in other ways. We have now progressed from purely static eBooks to interactive eBooks and in the next session and practical we will go one step further to look at how we can improve interactivity via super-templates in TREE and also the ability to construct html output objects.