A LyX/Sweave Tutorial

Jeff Laake

November 10, 2011

1 Introduction

The help files and tutorial that accompany LyX are very complete and this is not an attempt to replace those. My focus here is on the initial steps you would take to create a manuscript that uses Sweave to produce a reproducible research document. Some of the material I will cover is also covered in the LyX tutorial and I suggest that you read it as well. Also, a document written by Yihui Xie and others (http://www.lyx.org/trac/raw-attachment/ticket/7555/sweave.lyx) contains useful information about using Sweave with LyX. Also, get a good book on LaTeX. The book I have is A guide To LaTeX by Kopka and Daly. What I present here is more of a step-by-step approach of using LyX to create a Sweave document.

2 Creating a new document

As with any word processor, creating a new document only requires selecting File/New (Figure 1) which opens an empty document with a temporary name which you can change by using File/Save As. Each LaTeX document has a document class and the default class is article which is what you will normally use but there are many document classes that can be used (e.g., book). You can change the Document class by selecting Document/Settings from the LyX menu (Figure). To use Sweave, you need to add the Sweave module to the document as shown in Figure 2.

3 Adding Title, Author and Date

You could just begin typing text into your document but you most likely want to give it a title etc. The various structures in a LaTeX document are created with the button in the top left that will be displayed as Standard when you first open the document. These structures are called environments so I'll refer to this as the environment button. Standard represents the bulk of the document. To add a new environment to the document, like a title, you open the drop down

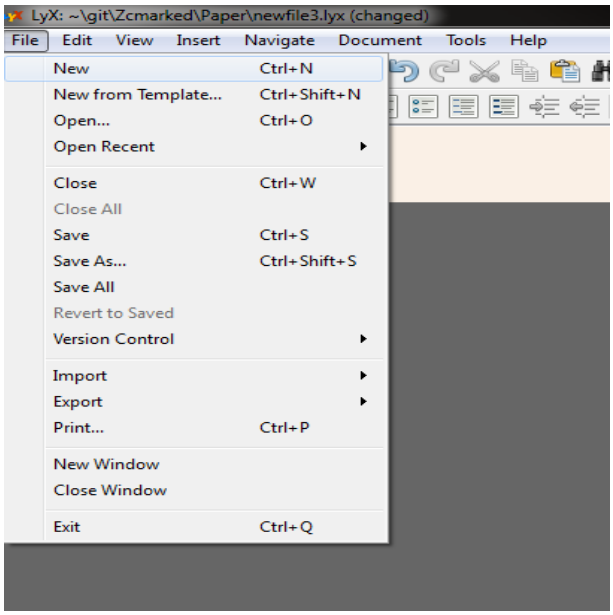


Figure 1: Creating a new empty document.

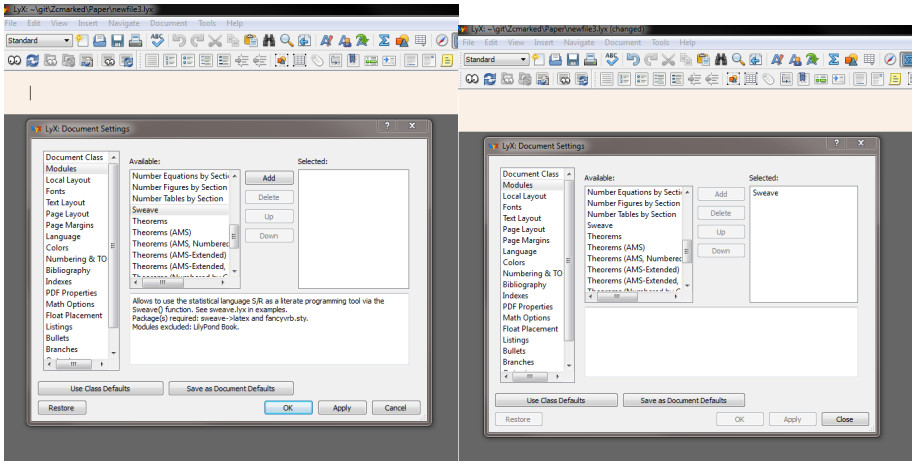


Figure 2: Adding Sweave module to the document with Document/Settings/Modules.

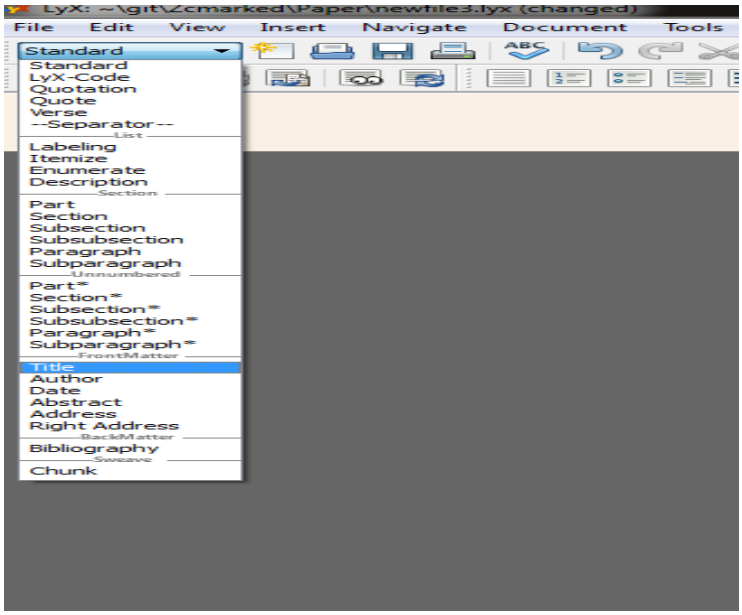


Figure 3: Adding a title to the document.

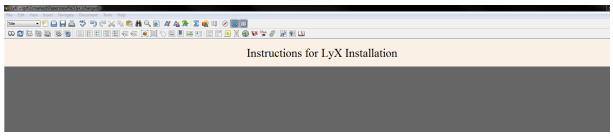


Figure 4: Entering your title.

box by clicking on the button and then selecting the environment, a title in this case (Figure 3).

This will change to the Title format for the document class and you can enter the title for your document (Figure 4). Make sure to hit enter after entering the text for your title. You will notice that the environment button will return to saying Standard but if you click anywhere on the title it will change to title. You can then do the same thing with Author. A date will automatically be entered and it will always use the current date. If you want to add a fixed date then you can use the Date environment in the same way as Title and Author. If you don't want a date to appear then add **\date{}** into your LaTeX preamble (Figure 5). Any other LaTeX that affects the entire document can be added to the preamble.

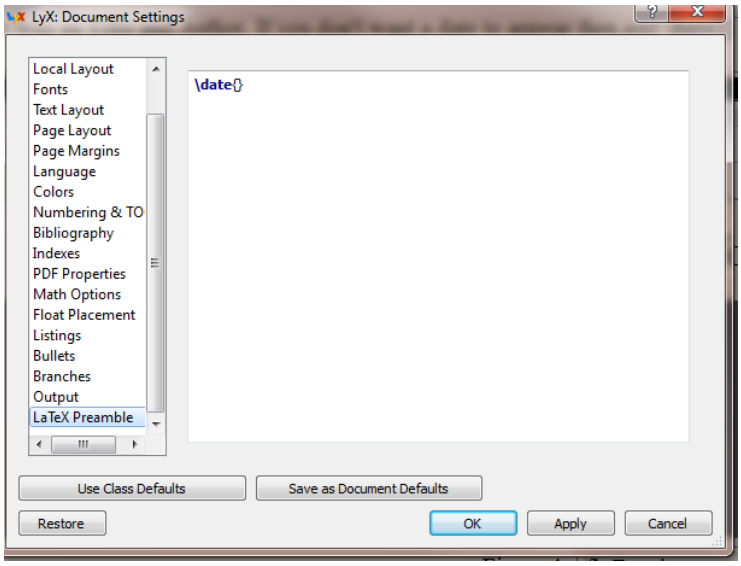


Figure 5: Adding LaTeX to the preamble.

4 Adding sections and text to your document

You can add numbered or unumbered sections to your document and within sections, sub-sections, sub-sub-sections etc. Again this is done with the environment button that was used to create title and author. Simply select Section for numbered sections and Section* for unnumbered sections (Figure 6). Then enter the text for your section header and press enter to return to standard entry. LyX/LaTeX takes care of the numbering for you.

After you have entered a section header you can enter whatever text you want. Below I pasted in the text from my LyX installation instructions (Figure 7) but you can also enter text manually. It was rather messy and I wanted to make it into an enumerated list so I did that by selecting the text and selecting the "enumerate" environment (Figure 8). It added numbers for each "paragraph" (separated by returns) in my text and then I cleaned up the text removing the numbers that were in the original text.

If you hit the enter key anywhere in the list it will start a new numbered entry at that point. The same thing occurs if you hit enter at the end of the list. To complete the list, simply change the environment back to Standard at the end of the list (Figure 9).

5 Entering R code

The bulk of your R code will be entered into "chunks" which is a Sweave construct. One way to create a chunk is select the chunk environment with the

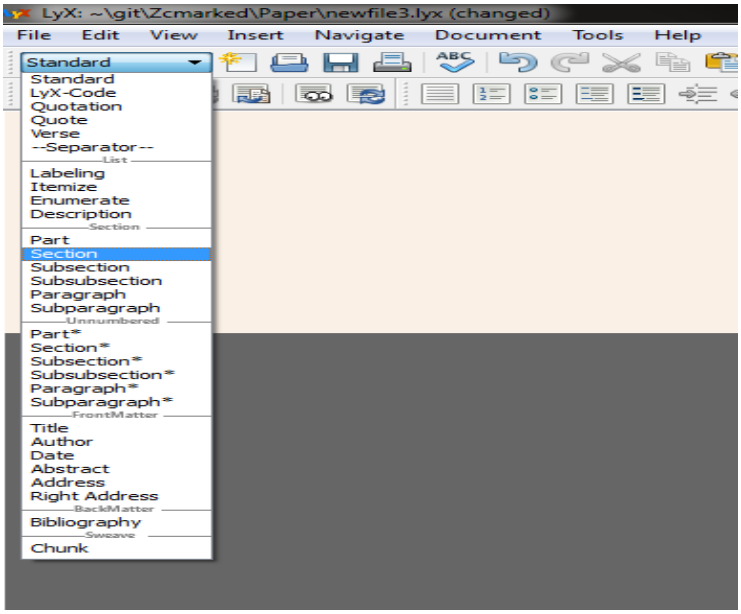
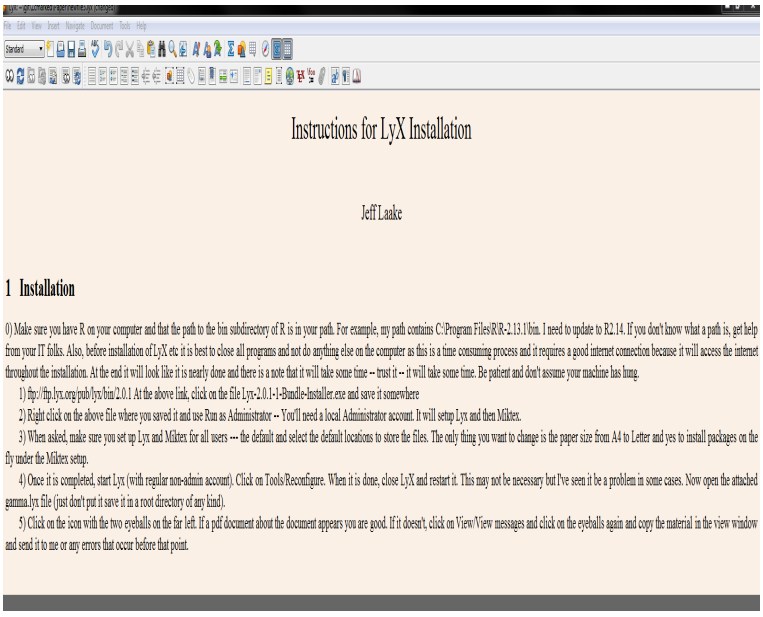


Figure 6: Adding a numbered section.



Instructions for LyX Installation

Jeff Laake

1 Installation

0) Make sure you have R on your computer and that the path to the bin subdirectory of R is in your path. For example, my path contains C:\Program Files\R\R-2.13.1\bin. I need to update to R2.14. If you don't know what a path is, get help from your IT folks. Also, before installation of LyX etc it is best to close all programs and not do anything else on the computer as this is a time consuming process and it requires a good internet connection because it will access the internet throughout the installation. At the end it will look like it is nearly done and there is a note that it will take some time -- trust it -- it will take some time. Be patient and don't assume your machine has hung.

1) ftp://ftp.lyx.org/pub/lyx/bin/2.0.1 At the above link, click on the file Lyx-2.0.1-1-Bundle-Installer.exe and save it somewhere

2) Right click on the above file where you saved it and use Run as Administrator -- You'll need a local Administrator account. It will setup Lyx and then Miktex.

3) When asked, make sure you set up Lyx and Miktex for all users -- the default and select the default locations to store the files. The only thing you want to change is the paper size from A4 to Letter and yes to install packages on the fly under the Miktex setup.

4) Once it is completed, start Lyx (with regular non-admin account). Click on Tools/Reconfigure. When it is done, close LyX and restart it. This may not be necessary but I've seen it be a problem in some cases. Now open the attached gamma.lyx file (just don't put it save it in a root directory of any kind).

5) Click on the icon with the two eyeballs on the far left. If a pdf document about the document appears you are good. If it doesn't, click on View/View messages and click on the eyeballs again and copy the material in the view window and send it to me or any errors that occur before that point.

Figure 7: Adding text.

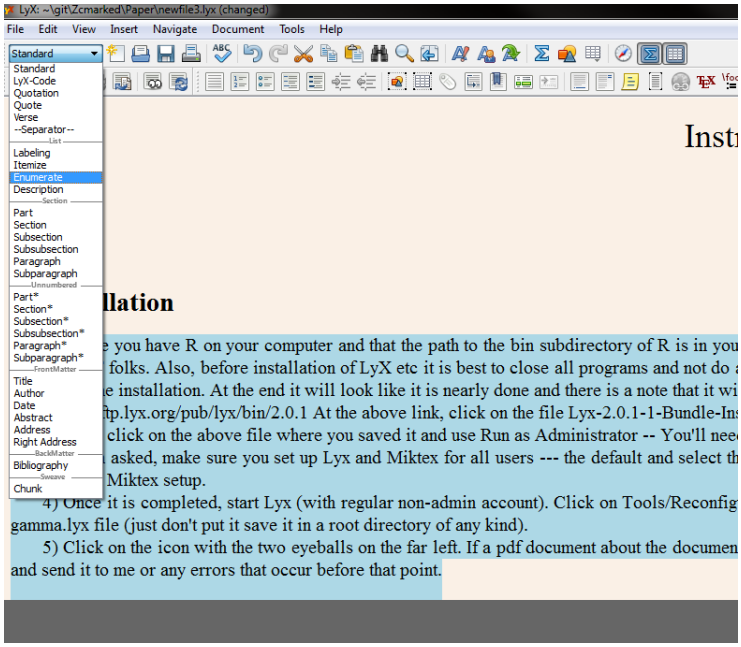


Figure 8: Enumerating a list.

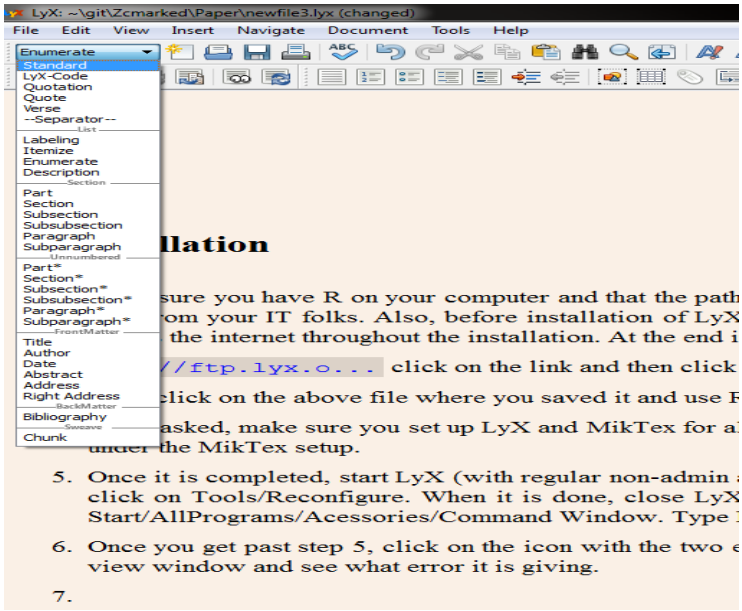


Figure 9: Ending a list.

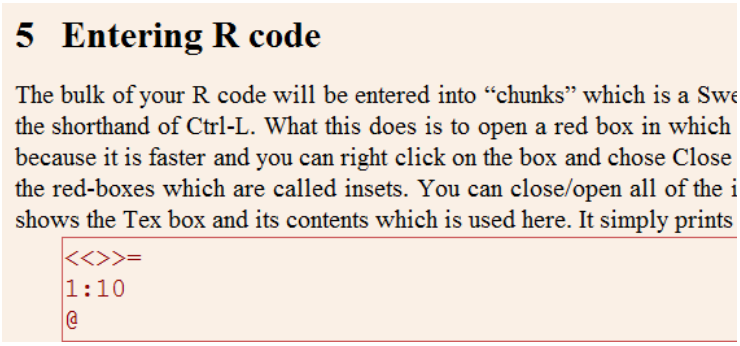


Figure 10: Entering R code.

environment button. I prefer instead to use Insert/Tex Code from the menu and I use the shorthand of Ctrl-L. What this does is to open a red box in which any LaTex code can be entered including a chunk. These are referred to as evil-red-text in LyX because Tex code can be a little daunting at first. I use this approach because it is faster and you can right click on the box and chose Close Inset which collapses the box so it is not in the way when you are editing the rest of the document. You can do the same thing for figures and other objects contained in the red-boxes which are called insets. You can close/open all of the insets from the View menu. When you open a Tex code box, it is empty and you need to enter the structure for the chunk and the code that you want. Figure 10 shows the Tex box and its contents which is used here. It simply prints the sequence 1:10 and the result which is the printed output for a vector containing the the numbers 1 to 10.

```
> 1:10

 [1]  1  2  3  4  5  6  7  8  9 10
```

The results of R code can be used anywhere in the document beyond the Tex box containing the R code. Thus one can simply put all of the R code in a .R file and use source("myfile.r") where myfile.r is the name of the file containing your R code. I typically put the bulk of my R code in an R package where it can be documented and maintained and then I only include code in the document that uses the packaged code. Here is an example in which I create a variable x and then show the sum of the elements of x in my document (Figure 11).

```
> x=1:10
```

Typically, you do not want to show the R code in your document unless you are writing a book about R or writing a tutorial like this one. You can prevent showing the code by using the option <<echo=F>>= when you define the chunk. Unless you are going to show some code and not others, you probably will want to set this behavior for all your chunks. You can do that by entering

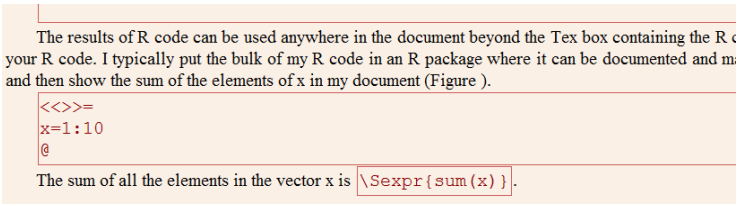


Figure 11: Using R code in an Sexpr to display results in the text.

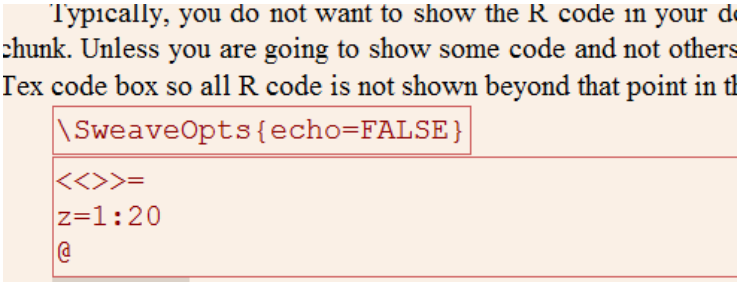


Figure 12: Tex boxes in this document which turn echo off so the contents are not shown in the pdf file.

SweaveOpts{}. For example, you can enter \SweaveOpts{echo=FALSE} into a Tex code box so all R code is not shown beyond that point in the output. See Figure 12 which shows the contents of this LyX file which has a chunk after the SweaveOpts is set but nothing is shown.

Even though the code is not shown, I can still use and display the results as in "the sum of the elements of z is" 210. The contents of the Tex code box for the sum of z is \Sexpr{sum(z)}. The various options for SweaveOpts and used in <<>>= are documented in the R help file for RWeaveLatex. One of the useful options is to give a name to your chunk as in <<mychunk,echo=FALSE>>. If you give it a name that name will appear in View Messages if there is an error.

6 Adding Figures

If you using Sweave, then many of your figures will be plots and other figures showing the results of your calculations. Rather than repeating it here, see the document (http://www.lyx.org/trac/raw-attachment/ticket/7555/sweave.lyx) or the sweave.lyx file that accompanies Lyx to see how to add R plots in figures. What I will show you is how to create a figure with a graphic or picture and how to add a label to the figure and use it to cross-reference in your document. In Figures 13 - 16 it shows how to add a Float/Figure, enter a Title, enter the graphic to be contained in the figure and add a label.

Once you have created the figure and given it a label, you can refer to the

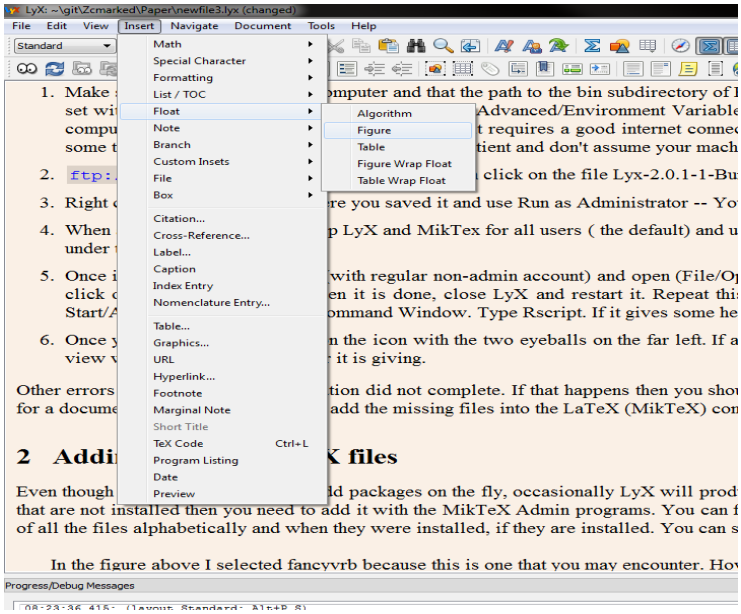


Figure 13: Adding a floating figure.

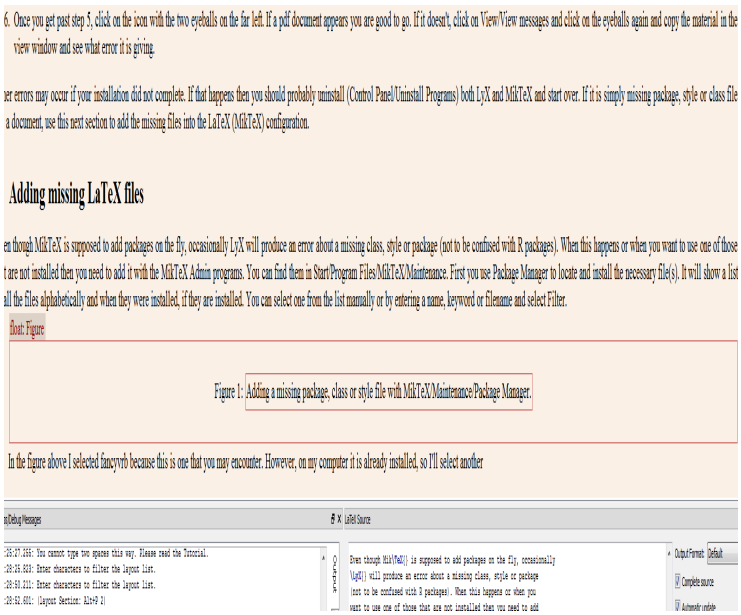


Figure 14: Adding the caption for the figure.

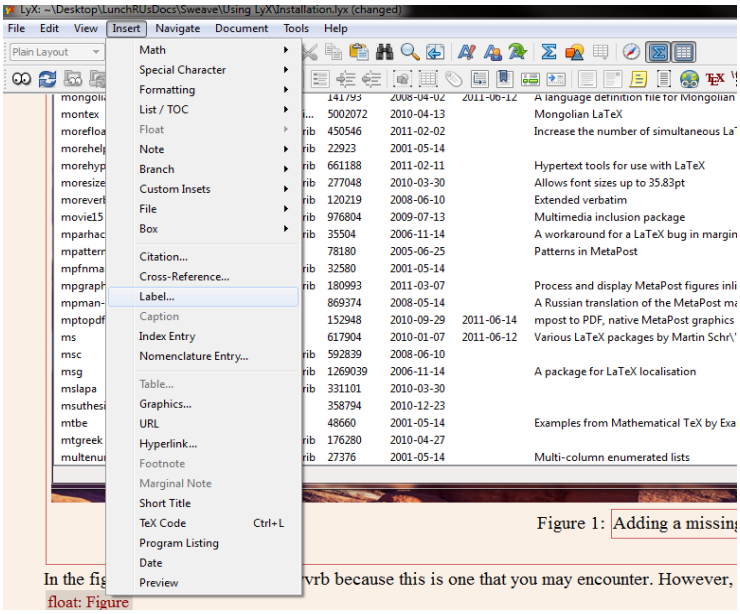


Figure 15: Adding a label to the figure for cross-referencing in the text.

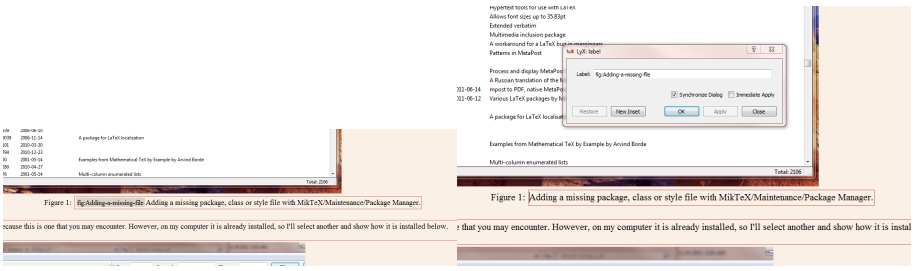


Figure 16: Entering the value for the label. Typically I use the default value that it offers which is the beginning portion of the caption as shown in the bottom portion of the figure. The result is shown in the top portion of the figure.

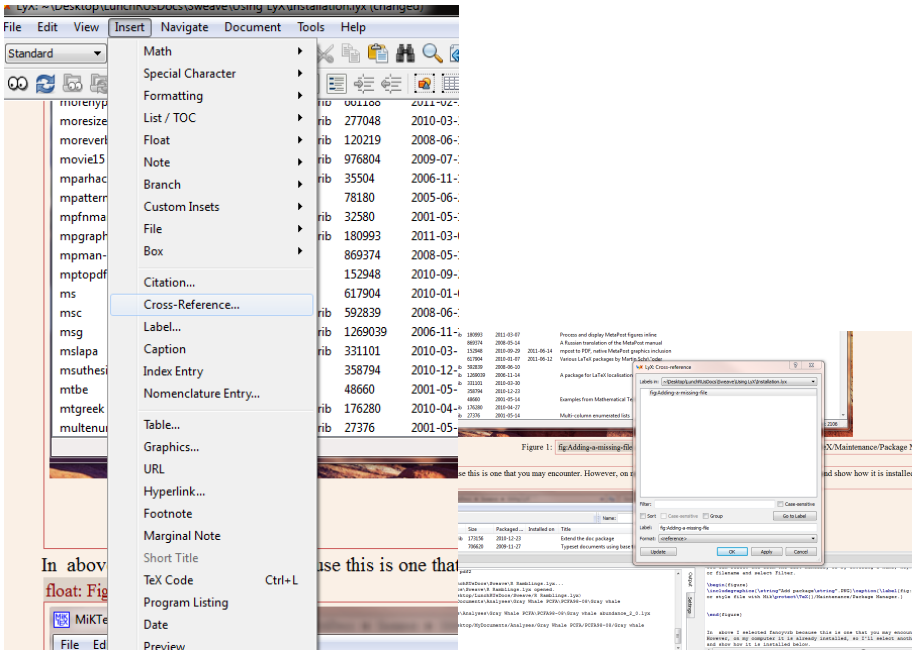


Figure 17: Using a label cross-reference in the document.

figure in the text by inserting a cross-reference which is a pointer to the label (Figure 17). What is substituted is the figure number so you will also want to add the word Figure or Fig. or whatever should accompany the number. If the figures are re-arranged then the numbers shift in the document automatically.

7 Adding a table

A table can easily be added with tools that are similar to what you would find in any word processor. If you select Insert/Table from the menu, a box will appear asking for the number of rows and columns that should be included in the table. There are many tools available for manipulating the table which will appear at the bottom of the screen if you click on the table icon at the top near the Sigma (summation sign) which displays the toolbar for equations. Once you create a table and are inside that environment then the table icons will work. If your focus is using R/Sweave, the you may want to consider using the R package xtable which will create the Tex code for the table with the contents of your dataframe, matrix, etc. Here is a simple example that loads the iris data and the xtable package. Then in a separate Tex code box uses the code to create the table with xtable. Figure 18shows the code used in this document.

```
> library(xtable)
> data(iris)
```

```
<<echo=TRUE>>=
library(xtable)
data(iris)
iris[1:3,]
@
The above shows the contents of the first 3 rows of iris as it would be printed in R. Below is the same contents in a nicely formatted table with a
<<results=tex>>=
print(xtable(iris[1:3,],caption="First 3 rows of the iris data"),caption.placement="top")
@
```

Figure 18: Tex code used to create the xtable iris data example. You must have the xtable R package installed to use this. The chunk containing the xtable must say results=tex so it knows to process the results with LaTeX.

```
> iris[1:3,]

  Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1          5.1         3.5          1.4         0.2  setosa
2          4.9         3.0          1.4         0.2  setosa
3          4.7         3.2          1.3         0.2  setosa
```

The above shows the contents of the first 3 rows of iris as it would be printed in R. Below is the same contents in a nicely formatted table with a caption.

Table 1: First 3 rows of the iris data

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.10	3.50	1.40	0.20	setosa
2	4.90	3.00	1.40	0.20	setosa
3	4.70	3.20	1.30	0.20	setosa

The R help files for xtable and print.xtable describe the various options for controlling the behavior of the table like caption.placement, etc.