

LyX and knitr

The `knitr` package allows one to embed R code within LyX and L^AT_EX documents. When a document is compiled into a PDF, LyX/L^AT_EX connects to R to run the R code and the code/output is automatically put into the PDF. In addition to this being a convenient way to use both LyX/L^AT_EX with R, it also provides an important component to the reproducibility of research (RR). For example, one can include the code for a data analysis described in a paper. This ensures that there would be no “copying and pasting errors” and also provide readers of the paper an immediate way to reproduce the research. RR continues to become more important and fortunately more tools are being developed to make it possible. Below are some discussions on the topic:

- AMSTAT News column on RR at <http://magazine.amstat.org/blog/2011/01/01/scipolicyjan11>.
- CRAN task view for RR and R at <http://cran.r-project.org/web/views/ReproducibleResearch.html>.
- Yihui Xie: Author of `knitr`
 - First and second editions of his *Dynamic Documents with R and knitr* book. Note that this book was typed in LyX.
 - Website for `knitr` at <http://yihui.name/knitr>

The Sweave environment is another way to include R code inside of LyX/L^AT_EX. This was developed prior to `knitr`, but it is more difficult to use.

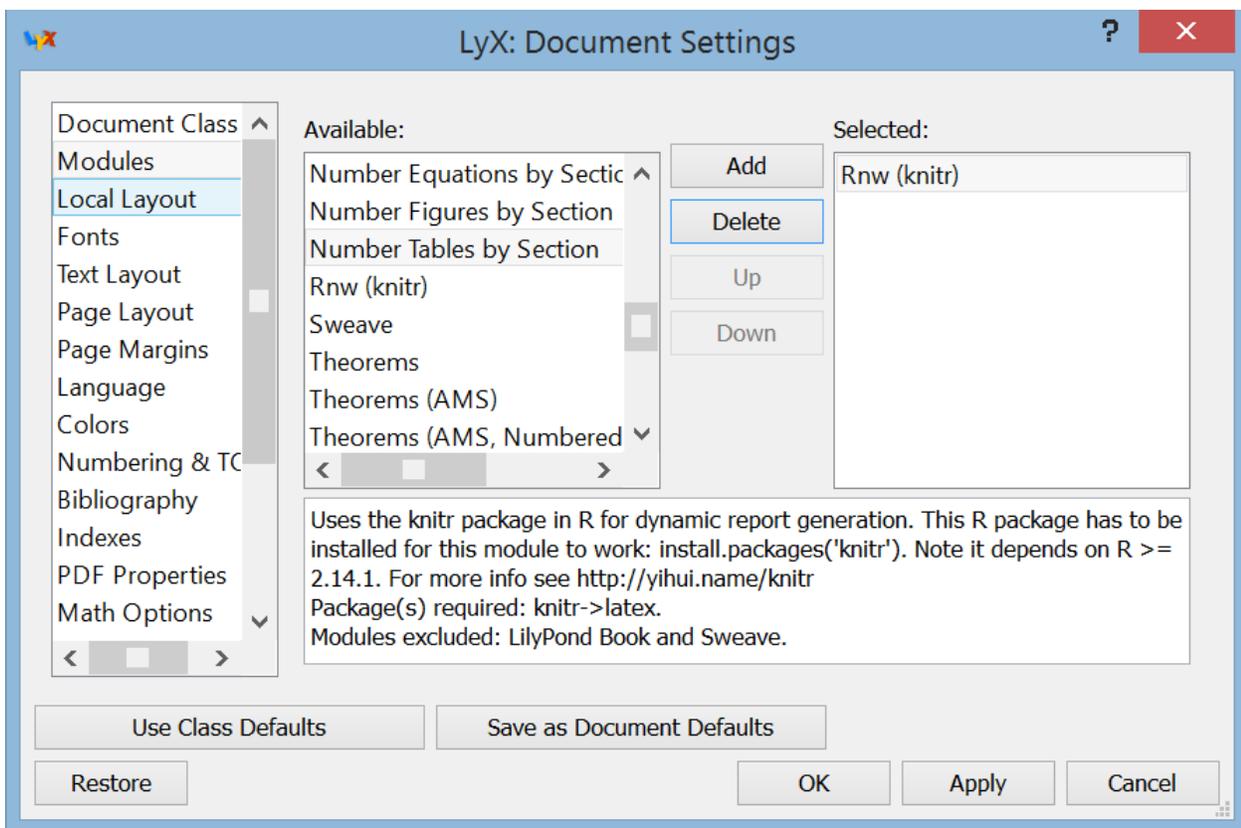
The purpose of this section is to examine the main components of `knitr` so that you will be able to complete the rest of the semester using LyX and `knitr` together for all assignments in our course! Also, a very important purpose is to give you the tools needed to complete all assignments in other R-based courses by using `knitr` and LyX together! The files used

here are `intro_example_cereal.lyx`, `intro_example_cereal.pdf`, `cereal.csv`, `ExternalCode.R`, `FirstBeamer-knitr.zip`, `JSM2015.zip`, and `RMarkdown.zip`.

Set-up

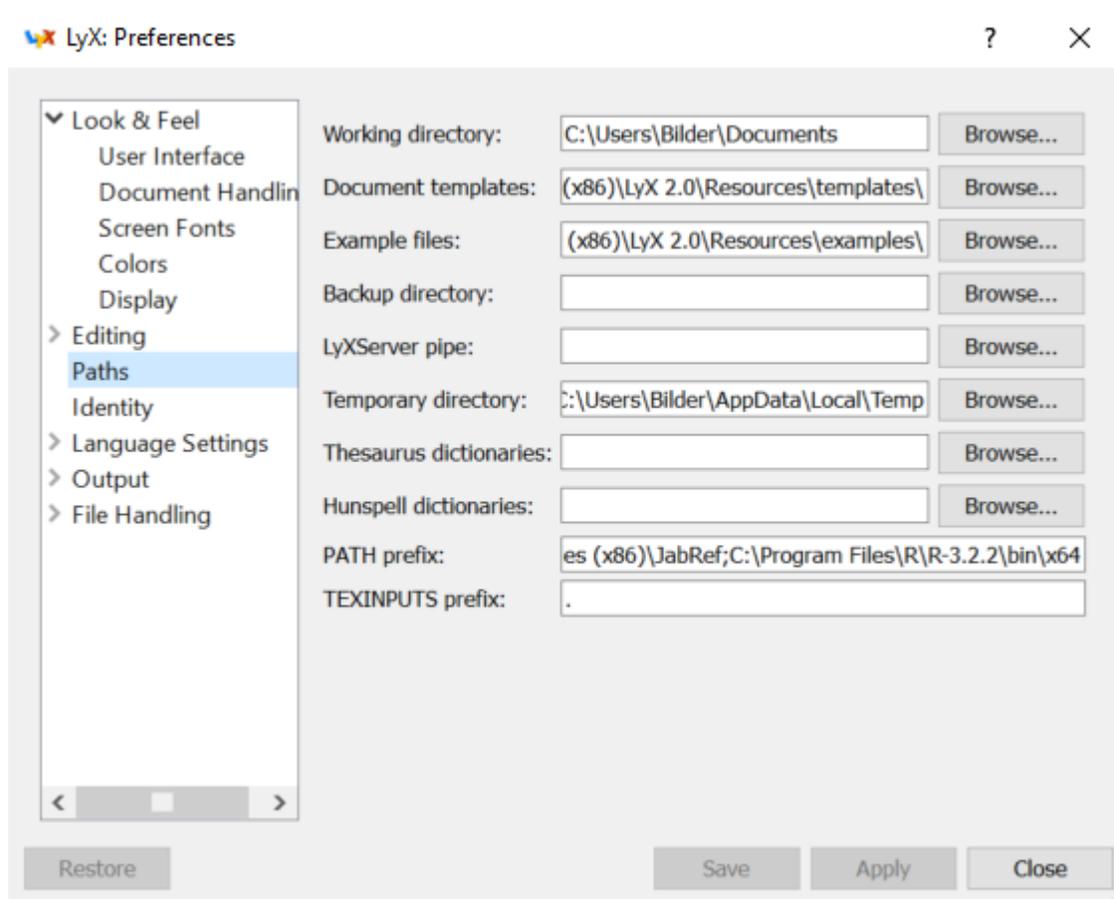
Below are the steps:

1. Install the `knitr` package in R. This will likely install a number of other packages too.
2. Select `DOCUMENT > SETTINGS > MODULES`. Then select the `Rnw (knitr)` AVAILABLE module and Add it to SELECTED. Select OK. Note that this step always has to be performed whenever you want to use `knitr` for the first time within a new document.



3. For LyX to know where `Rscript.exe` is located, select `TOOLS > PREFERENCES > PATHS` and add the location of `Rscript.exe` to the `PATH PREFIX` part. For example, I added `“;C:\Program Files\R\R-3.2.2\in\x64”` right after the location

for JabRef without any spaces. Make sure to include the semi-colon!



4. Select `TOOLS > RECONFIGURE` which tells LyX to find again all of the add-ons available with it.
5. Close LyX

When you re-open LyX, you should be able to use `knitr` with it. One way to determine if everything worked is to use my LyX example file that uses `knitr` and compile the document. If you open the LyX document and obtain a window that says there is a missing prerequisite for `knitr`, this means that you did not specify the path correctly.

Chunks

The `knitr` package uses “chunks” to run R code. Within a LyX document, a chunk is written inside a `LATEX` code box. The chunk syntax is:

```
<<Chunk name, options>>=
```

```
R code
```

```
@
```

Comments:

- Additional \LaTeX code can be included inside the \LaTeX code box but outside of the chunk. Therefore, the way `knitr` is used in \LaTeX is simply by using the exact same \LaTeX code that you would use in a \LaTeX program editor.
- The chunk name can include spaces unlike normal R code. However, Xie does mention that one should try to avoid spaces and periods on his website (<http://yihui.name/knitr/options>) to avoid possible problems with \LaTeX . Each chunk needs to have a unique name. They can also be left empty (no name), and `knitr` will generate a unique name for it during execution.
- Chapters 5-9 of Xie (2015) provide the main discussion for chunk options. The web page <http://yihui.name/knitr/options> also summarizes the available options.
- The equal sign at the end of the first line of a chunk must be included. You can interpret this similar to making an assignment (remember that `x<-2` can be written as `x=2` in R). Simply, we are creating a chunk with a particular name and putting a number of R instructions into it. Note that `<-` does not work at the end of the chunk line. Xie mentions that he prefers using `=` rather than `<-` in regular R coding, so this is probably the reason for the syntax. He may be the only advanced R user who prefers the equal sign!
- The R code is executed like normal within a chunk.
- `@` ends the chunk

Cereal data

I have created a separate document named `intro_example_cereal.lyx` that gives examples of how to use `knitr` with `LyX` in the context of my cereal data set (`cereal.csv`). All of my R code is also available in `cereal_knitr.R`. I have found it to be very useful to always create a separate R program that contains the code. This allows one to test the R code out before putting it into the `LyX` document. Unfortunately, when the R code in `LyX` has some type of error in it, `knitr` does a poor job with creating an error message that would help solve the problem.

Beamer

I have provided two examples of how Beamer and `knitr` work together. The first example corresponds to the first example given in the Beamer section. Rather than using program listing boxes for manually inserted R code/output, I use `knitr`. Below are a few notes about the `LyX` file:

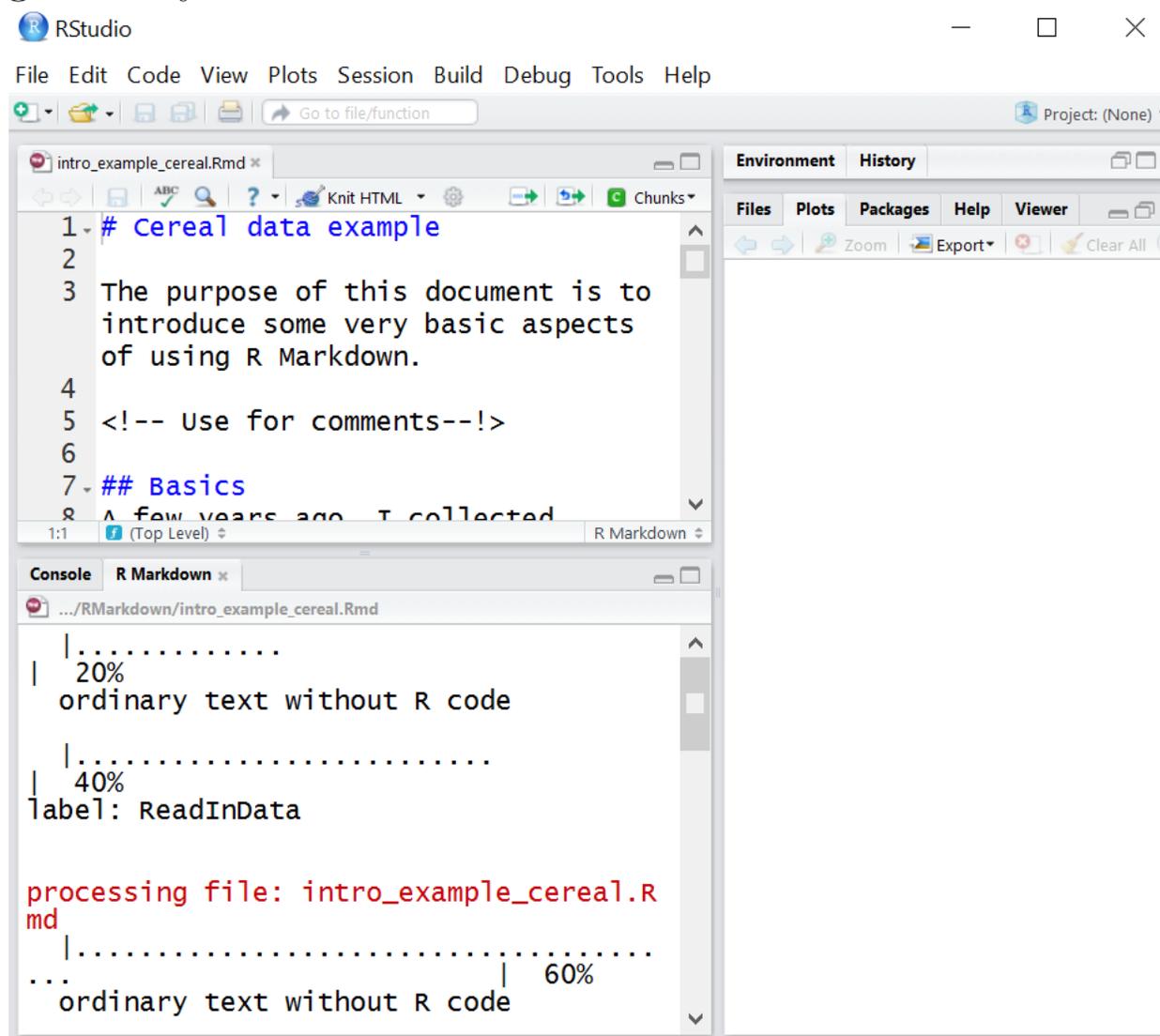
- Remember that an end-of-frame marker needs to be inserted between any two frames within the same section/sub-section.
- The `LATEX` code box for `knitr` needs to be put inside a fragile frame.
- It can be difficult to include all R code with output on a frame. Very often, you will need to compile the document multiple times while adjusting various settings and spacings to get everything to fit. I show how to use `\setlength\fbboxsep{-2mm}` on one frame to demonstrate one way to move code/output up a little to provide additional space (not necessarily needed for this particular case).

The second example comes from a JSM 2015 continuing education course that I gave based on my book. This was a full-day

short course where all 200+ slides were constructed using LyX and knitr. My course website contains the LyX files for only a portion of the short course. Again, you will see examples of how I deal with limited space on a frame. Also, you will see how I include R code/output within a bulleted list.

R Markdown

Markdown is an HTML-like language that allows you to create HTML documents. Combined with R and knitr, you can create nicely formatted, dynamic documents. This combination of Markdown and R is referred to as R Markdown. RStudio provides a good way to use it:



My intro_example_cereal.Rmd file was created in RStudio and

produced an `intro_example_cereal.html` file. To create the HTML file from `intro_example_cereal.Rmd`, select the KNIT HTML icon. PDFs and Word files can also be created with the addition of the Pandoc software program.

Recently, Xie gave a seminar at UNL to discuss his work with R Markdown. He alluded to that R Markdown may actually be a competitor to \LaTeX in the future.

knitr-like tool for use with SAS

StatRep is a package available from CTAN that is designed specifically for use with SAS code in \LaTeX . Overall, I have found it to be somewhat clunky in comparison to using `knitr` with \LaTeX and `LyX`. Also, this package cannot be as easily used with `LyX` as `knitr` can be. For more information regarding StatRep, please see <http://support.sas.com/rnd/app/papers/statrep/statrepmanual.pdf>.

There are ways to have SAS generate \LaTeX code that can be inserted into `LyX` through using the ODS. Carl Schwarz's "SAS - tricks and tips" document (<http://people.stat.sfu.ca/~cschwarz/Stat-650/Notes/PDF/ChapterSASTricks.pdf>) provides some nice advice regarding how to use it. Additional information is available at <http://support.sas.com/rnd/base/ods/odsmarkup/latex.html>.