A simple workflow for using R with Microsoft products

Marcus W. Beck

USEPA NHEERL Gulf Ecology Division, Gulf Breeze, FL
Email: beck.marcus@epa.gov, Phone: 850 934 2480

May 21, 2014

The problem...

- R is great and has an increasing user base
- RStudio is integrated with multiple document preparation systems
- Output documents are not in a format that facilitates collaboration with non R users, e.g., pdf, html
- Data coming to you may be in a proprietary format, e.g., xls spreadsheet

The solution?

- Solution one - Make liberal use of 'projects' within RStudio
- Solution two - Use `gdata` package to import excel data
- Solution three - Get pandoc to convert document formats - http://johnmacfarlane.net/pandoc/

The solution?

- Solution one - Make liberal use of 'projects' within RStudio
- Solution two - Use `gdata` package to import excel data
- Solution three - Get pandoc to convert document formats - http://johnmacfarlane.net/pandoc/

Not recommended for simple tasks unless you really, really love R

An example workflow

- I will present a workflow for integrating Microsoft products within RStudio as an approach to working with non R users
- Idea is to never leave the RStudio environment - dynamic documents!
- General workflow...

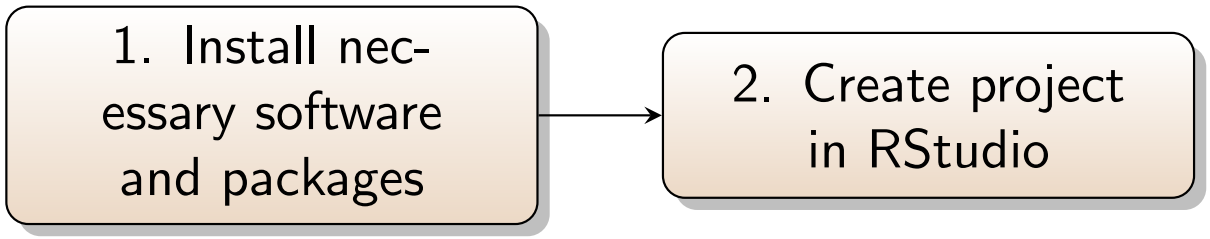
An example workflow

- I will present a workflow for integrating Microsoft products within RStudio as an approach to working with non R users
- Idea is to never leave the RStudio environment - dynamic documents!
- General workflow...

1. Install necessary software and packages

An example workflow

- I will present a workflow for integrating Microsoft products within RStudio as an approach to working with non R users
- Idea is to never leave the RStudio environment - dynamic documents!
- General workflow...

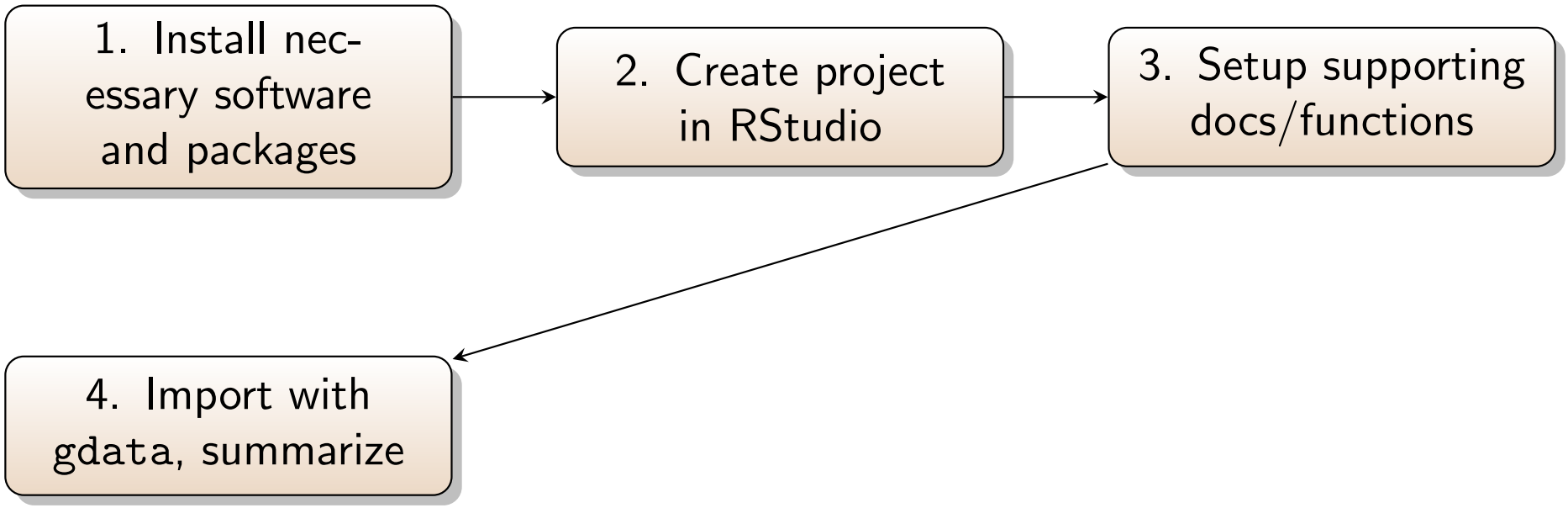


An example workflow

- I will present a workflow for integrating Microsoft products within RStudio as an approach to working with non R users
- Idea is to never leave the RStudio environment - dynamic documents!
- General workflow...

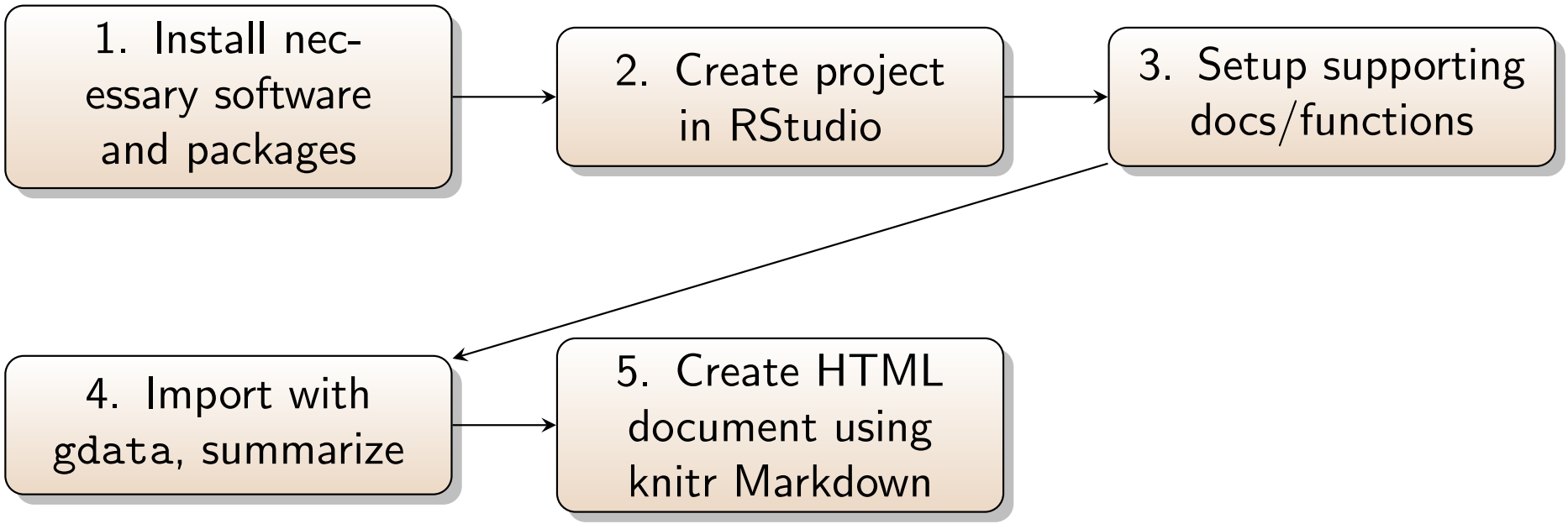
An example workflow

- I will present a workflow for integrating Microsoft products within RStudio as an approach to working with non R users
- Idea is to never leave the RStudio environment - dynamic documents!
- General workflow...



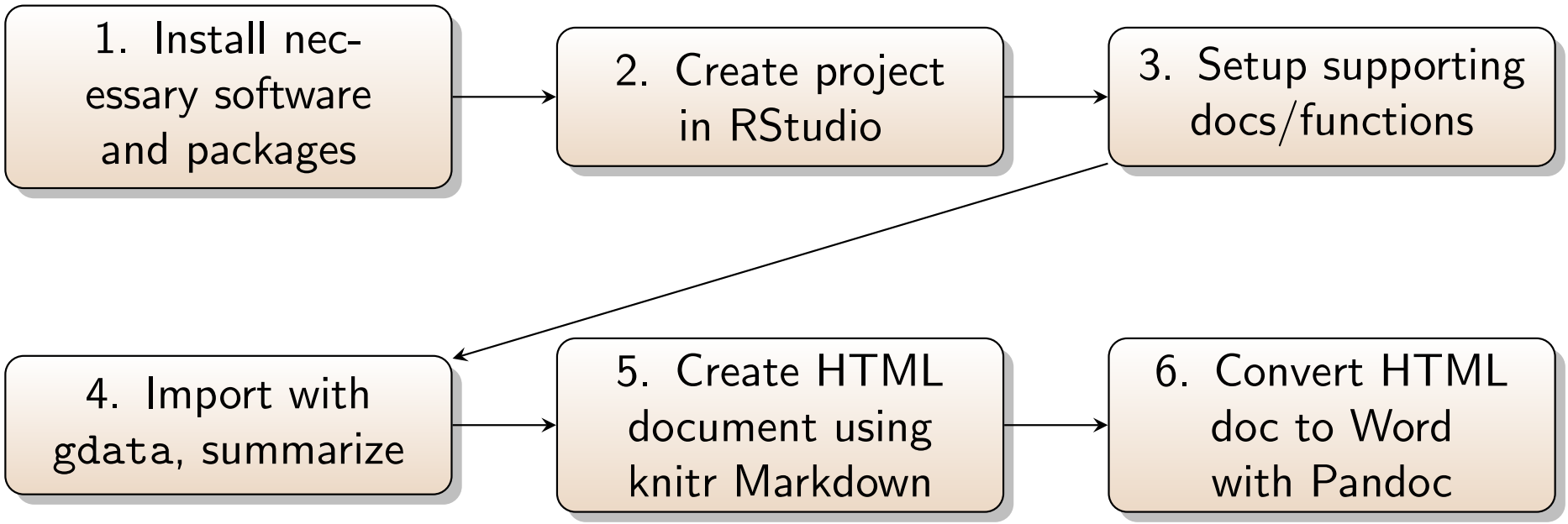
An example workflow

- I will present a workflow for integrating Microsoft products within RStudio as an approach to working with non R users
- Idea is to never leave the RStudio environment - dynamic documents!
- General workflow...



An example workflow

- I will present a workflow for integrating Microsoft products within RStudio as an approach to working with non R users
- Idea is to never leave the RStudio environment - dynamic documents!
- General workflow...



The example

You are sent an Excel file of data to summarize and report but you love R and want to do everything in RStudio...

SiteName	Year	Restoration	Reference	Observer.Names	Precipitation	Temperature
IGH	2005	3	3	Tyler_Amanda	0	48
Kelly	2005	4	2	Patrick_Chelsea	0	48
Carlton	2005	2	3	David_Megan	0	48
IGH	2006	9	6	Tyler_Amanda	0	52
Kelly	2006	9	1	David_Megan	0	52
Carlton	2006	7	3	Patrick_Chelsea	0	52
IGH	2007	12	7	David_Megan	12	41
Kelly	2007	2	18	Jeremy_Lucy	12	41
Carlton	2007	11	2	Patrick_Chelsea	12	41
IGH	2008	9	4	Tyler_Amanda	0	54
Kelly	2008	14	5	David_Megan	0	54
Carlton	2008	13	3	Patrick_Chelsea	0	54
IGH	2009	18	7	Patrick_Chelsea	0	55
Kelly	2009	16	5	David_Megan	0	55
Carlton	2009	20	1	Tyler_Amanda	0	55
IGH	2010	12	2	David_Megan	0	61
Kelly	2010	15	3	Patrick_Chelsea	0	61
Carlton	2010	24	4	Tyler_Amanda	0	61

Step 1

Install necessary software and Packages

Step 1

Install necessary software and Packages

- R and RStudio (can do with other R editors)
- Microsoft Office

Step 1

Install necessary software and Packages

- R and RStudio (can do with other R editors)
- Microsoft Office
- Strawberry Perl for using `gdata` package
- Pandoc

Step 1

Install necessary software and Packages

- R and RStudio (can do with other R editors)
- Microsoft Office
- Strawberry Perl for using `gdata` package
- Pandoc
- Packages: `gdata`, `knitr`, `utils`, `xtable`, others as needed...

Step 2

Create a project in RStudio

- Create a folder or use existing on local machine
- Add .Rprofile file to the folder for custom startup
- Move all data you are working with to the folder
- Literally create project in RStudio
- Set options within RStudio

Step 3

Setup supporting docs/functions, i.e., .Rprofile, functions, report, master

.Rprofile

```
# library path
.libPaths("C:\\Users\\mbeck\\R\\library")

# startup message
cat("My project...\n")

# packages to use
library(utils)  # for system commands
library(knitr)  # for markdown
library(gdata)  # for import xls
library(reshape2)  # data format conversion
library(xtable)  # easy tables
library(ggplot2)  # plotting

# perl path for gdata
prl_pth <- "C:/strawberry/perl/bin/perl.exe"

# functions to use
source("my_funcs.r")
```

Step 3

Setup supporting docs/functions, i.e., .Rprofile, functions, report, master

my_funcs.r

```
###### functions for creating report, created May 2014, M. Beck

###### processes data for creating output in report, 'dat_in' is input
###### data as data frame, output is data frame with converted variables
proc_fun <- function(dat_in) {

    # convert temp to C
    dat_in$Temperature <- round((dat_in$Temperature - 32) * 5/9)

    # convert data to long format
    dat_in <- melt(dat_in, measure.vars = c("Restoration", "Reference"))

    return(dat_in)

}

###### creates linear model for data, 'proc_dat' is processed data
###### returned from 'proc_fun', output is linear model object
mod_fun <- function(proc_in) lm(value ~ variable + Year, dat = proc_in)
```

Step 3

Setup supporting docs/functions, i.e., .Rprofile, functions, report, master

report.Rmd

````
======================
Here's a report I made for `r gsub('/|.xlsx','',name)`
----------------------

```{r echo=F, include=F}
# import data
url <- paste0('http://beckmw.files.wordpress.com/2014/05', name)
dat <- read.xls(xls = url, sheet = 'Sheet1', perl = prl_pth)

# process data for tables/figs
dat <- proc_fun(dat)

# model of data
mod <- mod_fun(dat)
```

### Model summary
```{r results='asis', echo=F}
print.xtable(xtable(mod, digits = 2), type = 'html')
```

### Figure of restoration and reference by year
```{r reg_fig, echo = F, fig.width = 5, fig.height = 3, dpi=200}
ggplot(dat, aes(x = Year, y = value, colour = variable)) +
  geom_point() +
  stat_smooth(method = 'lm')
```
````

Step 3

Setup supporting docs/functions, i.e., .Rprofile, functions, report, master

master.r

```
# file to process
name <- "/my_data.xlsx"

# rmd to html
knit2html("report.Rmd")

# pandoc conversion of html to word doc
system(paste0("pandoc -o report.docx report.html"))
```

Steps 4 - 6

After creating supporting documents in Project directory, final steps are completed by running 'master.r'

- Step 4 - xls file imported using `gdata` package, implemented in 'report.Rmd'
- Step 5 - HTML document created by converting 'report.Rmd' with `knit2html` in 'master.r'
- Step 6 - HTML document converted to Word with Pandoc by invoking system command

master.r

```
# file to process
name <- "/my_data.xlsx"

# rmd to html
knit2html("report.Rmd")

# pandoc conversion of html to word doc
system(paste0("pandoc -o report.docx report.html"))
```