

The Surgical Informatics Cookbook

Surgical Informatics, University of Edinburgh

2019-09-10

Contents

1	Introduction	5
1.1	How to contribute	5
1.2	Rules of posting	6
1.3	Indexing	6
2	Literature	9
3	Methods	11
4	Machine learning	13
4.1	Deep learning	13
5	Final Words	15
6	Plotting	17
6.1	GGHighlight Example	17

Chapter 1

Introduction

1.1 How to contribute

(Steps 1. to 5. only need to be done once - to set-up.)

1. Connect your RStudio and GitHub using these instructions, only up to “Create new project” is necessary here (the repository/project already exists): <https://www.datasurg.net/2015/07/13/rstudio-and-github/>
2. Get your GitHub account added to the surgicalinformatics organisation on GitHub (ask Riinu/Ewen): <https://github.com/SurgicalInformatics>
3. Open the terminal on argonaut, go to your home folder (just `cd` will default to taking you home). If the prompt says `username@argonaut:~$` that means you’re home - `~` means user’s home.
4. In the terminal, do `git clone git@github.com:SurgicalInformatics/cookbook.git`.
5. Look at the Files tab - a new folder called `cookbook` should have appeared at the bottom. Click on it and open the `cookbook.Rproj` file.
6. Add your thing by editing the appropriate `.Rmd` file - there’s one for each chapter. Use the Built tab to Built your changes into a book.
7. If anyone has pushed since you cloned/last pulled (hopefully they’ve not been working on the exact same chapter): click on More - Clean All. Then Pull from the Git tab. This only cleans the output files - html and PDF, it will not touch the changes you’ve made in the `.Rmd` file.
8. Then Build Book again - this will include the new changes you pulled as well as your changes.
9. Git tab - commit everything, push quickly before anyone else does or you’ll have to go back to step 7. You can check for new pushed commits here:

<https://github.com/SurgicalInformatics/cookbook/commits/master> Alternatively there's no harm in clicking the Pull button again - it should then say "Already up-to-date".

Pro tip: instead of clicking on every single file in the Git tab, go to the terminal, `cd cookbook` to go to the project folder if still home, and do `git add -A` which is the same thing. Still need to Commit though!

10. Have fun!

1.2 Rules of posting

Rules of how to post here.

1.3 Indexing

1.3.1 Index

Bold index headings:

`\index{linear regression@\textbf{linear regression}}` (ticks in .Rmd file are excluded when actually using)

Sub-entries of bold headings:

`\index{linear regression@\textbf{linear regression}!diagnostics}`

Stand-alone entries:

`\index{linear regression}`

1.3.2 Chapter and section references

You can label chapter and section titles using `{#label}` after them, e.g., we can reference Chapter `\@ref(intro)` (ticks in .Rmd are excluded when actually using). If you do not manually label them, there will be automatic labels anyway, e.g., Chapter `\@ref(methods)`.

1.3.3 Figure and table references

Figures and tables with captions will be placed in `figure` and `table` environments, respectively.

```
par(mar = c(4, 4, .1, .1))
plot(pressure, type = 'b', pch = 19)
```

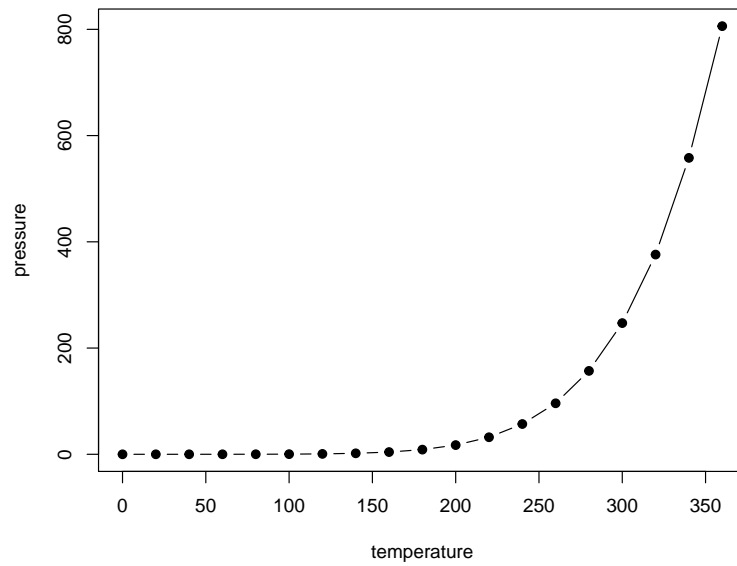


Figure 1.1: Here is a nice figure!

Reference a figure by its code chunk label with the `fig:` prefix, e.g., see Figure `\@ref(fig:nice-fig)`. Similarly, you can reference tables generated from `knitr::kable()`, e.g., see Table `\@ref(tab:nice-tab)`.

```
knitr::kable(  
  head(iris, 20), caption = 'Here is a nice table!',  
  booktabs = TRUE  
)
```

1.3.4 Citations

You can write citations, too. For example, we are using the **bookdown** package (Xie, 2018) in this sample book, which was built on top of R Markdown and **knitr** (Xie, 2015).

Table 1.1: Here is a nice table!

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.0	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa
4.6	3.4	1.4	0.3	setosa
5.0	3.4	1.5	0.2	setosa
4.4	2.9	1.4	0.2	setosa
4.9	3.1	1.5	0.1	setosa
5.4	3.7	1.5	0.2	setosa
4.8	3.4	1.6	0.2	setosa
4.8	3.0	1.4	0.1	setosa
4.3	3.0	1.1	0.1	setosa
5.8	4.0	1.2	0.2	setosa
5.7	4.4	1.5	0.4	setosa
5.4	3.9	1.3	0.4	setosa
5.1	3.5	1.4	0.3	setosa
5.7	3.8	1.7	0.3	setosa
5.1	3.8	1.5	0.3	setosa

Chapter 2

Literature

Here is a review of existing methods.

Chapter 3

Methods

We describe our methods in this chapter.

Chapter 4

Machine learning

4.1 Deep learning

4.1.1 Pulling images from REDCap directly to argodeep

4.1.1.1 Original file names

```
library(REDCapR)
uri = "https://redcap.cir.ed.ac.uk/api/"
token = "" # API token here
record_list = 1:318
field_list = c("photo", "photo_2", "photo_3", "photo_4")
event_list = c("wound_concerns_arm_2", "questionnaire_1_arm_2",
               "questionnaire_2_arm_2", "questionnaire_3_arm_2")
directory = "wound_raw" # destination directory must exist already

for(record in record_list){
  for(field in field_list){
    for(event in event_list){
      result =
        tryCatch({ # suppress breaking error when no image in slot
          redcap_download_file_oneshot(
            record      = record,
            field       = field,
            redcap_uri  = uri,
            token       = token,
            event       = event,
            overwrite   = TRUE,
```

```

        directory      = directory
    )
    }, error=function(e){})
}
}
}

```

4.1.1.2 Named from REDCap record ID and event

```

library(REDCapR)
uri = "https://redcap.cir.ed.ac.uk/api/"
token = "" # API token here
record_list = 1:318
field_list = c("photo", "photo_2", "photo_3", "photo_4")
event_list = c("wound_concerns_arm_2", "questionnaire_1_arm_2",
               "questionnaire_2_arm_2", "questionnaire_3_arm_2")
directory = "wound_named" # destination directory must exist already

for(record in record_list){
  for(field in field_list){
    for(event in event_list){
      file_name = paste0(record, "_", field, "_", event, ".jpg")
      result =
        tryCatch({
          redcap_download_file_oneshot(
            record      = record,
            field       = field,
            redcap_uri  = uri,
            token       = token,
            event       = event,
            overwrite   = TRUE,
            directory   = directory,
            file_name   = file_name
          )
        }, error=function(e){})
    }
  }
}

```

Chapter 5

Final Words

We have finished a nice book.

Chapter 6

Plotting

6.1 GGHhighlight Example

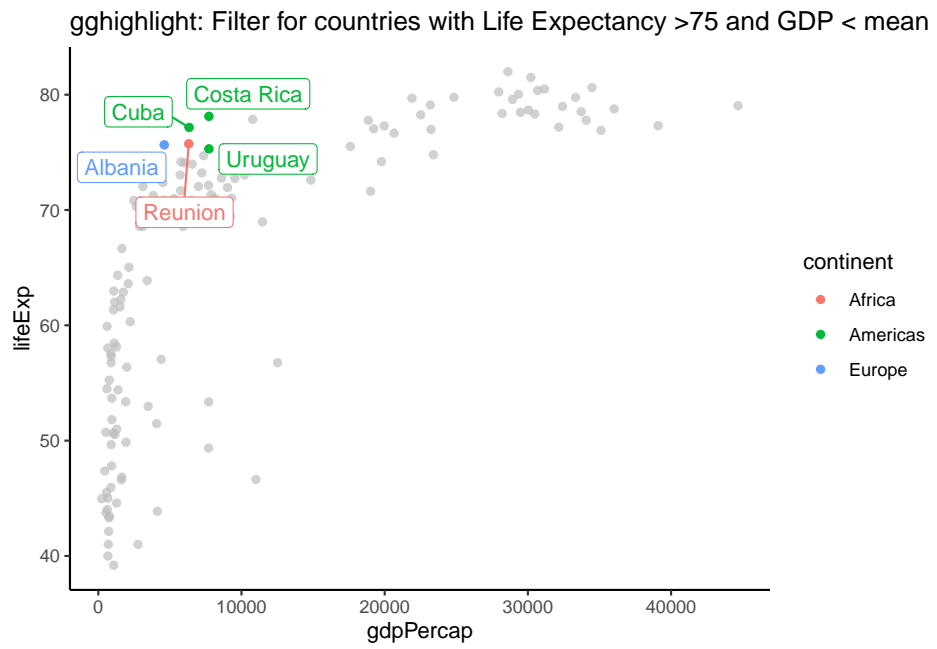
Plotting with gghighlight is pretty awesome allowing you to filter on any variable. It seems that gghighlight overwrites any 'colour' variable you put in the main aes. To get round this and have labels, save as a plot and add geom_label_repel separately.

```
library(gghighlight)
library(ggrepel)

mydata=gapminder

plot = mydata %>%
  filter(year == "2002") %>%
  ggplot(aes(x = gdpPercap, y = lifeExp, colour=continent)) +
  geom_point()+
  gghighlight(lifeExp > 75 & gdpPercap < mean(gdpPercap), label_key = country, use_direct_label = TRUE) +
  theme_classic()+
  labs(title= "gghighlight: Filter for countries with Life Expectancy >75 and GDP < mean" )

plot + geom_label_repel(aes(label= country), show.legend = FALSE) #only needed if you use use_direct_label = TRUE
```



Bibliography

Xie, Y. (2015). *Dynamic Documents with R and knitr*. Chapman and Hall/CRC, Boca Raton, Florida, 2nd edition. ISBN 978-1498716963.

Xie, Y. (2018). *bookdown: Authoring Books and Technical Documents with R Markdown*. R package version 0.7.