Tips n Tricks

Paula Andrea Martinez

2020-11-09

For beginners

*Manage your project structure

Tip - Create an RStudio Project

`File > New Project`

Tip - Create a folder structure for your project

You can click on `New Folder` and create your structure or you can do it from a script

```
dir.create("data")
```

Trick - Create a package that does that for you

tidyproject [1]

[1] Find it on GitHub orchid00/tidyproject

```
remotes::install_github("orchid00/tidyproject")
library(tidyproject)
# This function created my folder structure
# with data, scripts, plots, rmarkdown
#createStr()
```

Tip - Use here from the here package

It will save you headaches about paths you don't need to worry about

```
install.packages("here")
here("data","myfile.csv")
```

Trick - Open Recent

Please don't spend time navigating through your folder structure to find a recent project or file.

File > Recent Files File > Recent Projects

Also, *top right corner arrow down*

Trick - Find in Files

Shortcut `Cmd / Ctrl + Shift + F`

Tip - Add comments

Use a # at the beginning of the line or at the end of the line

Trick - Add comments

Shortcut `Cmd / Ctrl + Shift + C`

```
# comment
```

Tip - Add sections to your code

At the end of a section add 4 dashes - or 4 hashes #

```
# this is a section ----
```

Tip - Get data in

You can download files from the web directly into your project

```
?download.file
```

*Cheatsheets

Tip - used them when you are learning or to refresh

Top menu *Help* > *Cheatsheets*

*Markdown wise

Markdown is great! You can do websites, word docs, PDFs, books, etc.

Trick - Add a code chunk

Shortcut `Cmd / Ctrl + Alt + I`

Tip - Check code chunk options

I never remember so here is a cheat sheet[2]

Tip - Rmarkdown Theme Gallery

Use what is available [3] and make sure you have pandoc installed, otherwise you might see some errors.

For example, if you are on Ubuntu focal [4].

This page and PDF are using the Tufte handout style [5].

For HTML output, use `tufte_html` in the YAML metadata at the beginning of an R Markdown document (see an example below).

[2] `https://raw.githubusercontent.com/rstudio/cheatsheets/master/rmarkdown-2.0.pdf`

[3] `https://www.datadreaming.org/post/r-markdown-theme-gallery/`

[4] here is what I needed `https://packages.ubuntu.com/focal/pandoc`

[5] Tufte is a style that Edward Tufte uses in his books and handouts. Tufte's style is known for its extensive use of sidenotes, tight integration of graphics with text, and well-set typography. See Github repositories tufte-latex, tufte-css and its implementation into the **tufte** package

```
---
title: "An Example Using the Tufte Style"
author: "Paula Andrea Martinez"
output:
  tufte::tufte_handout: default
  tufte::tufte_html: default
---
```

INTERMEDIATE

Shortcuts all

Alt+Shift+K

**Manage your files*

Tip - check your files and folders

Lists your files and folders

```
library(here)
```

```
## here() starts at /home/paula/Documents/Projects/Rprojects/rladies/tips-n-tricks
```

```
dir(here(""))
```

```
## [1] "data"                "plots"               "rmarkdown"
## [4] "scripts"             "tips-n-tricks.Rproj"
```

Trick - get more information about files and folders

```
fs::dir_info(here(""))
```

```
## # A tibble: 5 x 18
##   path       type   size permissions modification_time   user  group device_id
##   <fs::path> <fct> <fs:> <fs::perms> <dttm>              <chr> <chr>     <dbl>
## 1 /home/pau~ dire~    4K rwxrwxr-x   2020-10-27 20:17:20 paula paula     66307
## 2 /home/pau~ dire~    4K rwxrwxr-x   2020-10-27 20:17:20 paula paula     66307
## 3 /home/pau~ dire~    4K rwxrwxr-x   2020-11-09 21:55:24 paula paula     66307
## 4 /home/pau~ dire~    4K rwxrwxr-x   2020-10-27 22:12:33 paula paula     66307
## 5 /home/pau~ file    205 rw-rw-r--   2020-11-09 21:46:10 paula paula     66307
## # ... with 10 more variables: hard_links <dbl>, special_device_id <dbl>,
## #   inode <dbl>, block_size <dbl>, blocks <dbl>, flags <int>, generation <dbl>,
## #   access_time <dttm>, change_time <dttm>, birth_time <dttm>
```

**Writing code*

Tip - Naming things

Please watch Naming things from Jenny Bryan [6]

[6] https://speakerdeck.com/jennybc/how-to-name-files

Tip - replacing NAs

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
```

```
## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.4     v dplyr   1.0.2
## v tidyr   1.1.1     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
msleep <- ggplot2::msleep
glimpse(msleep)
```

```
## Rows: 83
## Columns: 11
## $ name         <chr> "Cheetah", "Owl monkey", "Mountain beaver", "Greater s...
## $ genus        <chr> "Acinonyx", "Aotus", "Aplodontia", "Blarina", "Bos", "...
## $ vore         <chr> "carni", "omni", "herbi", "omni", "herbi", "herbi", "c...
## $ order        <chr> "Carnivora", "Primates", "Rodentia", "Soricomorpha", "...
## $ conservation <chr> "lc", NA, "nt", "lc", "domesticated", NA, "vu", NA, "d...
## $ sleep_total  <dbl> 12.1, 17.0, 14.4, 14.9, 4.0, 14.4, 8.7, 7.0, 10.1, 3.0...
## $ sleep_rem    <dbl> NA, 1.8, 2.4, 2.3, 0.7, 2.2, 1.4, NA, 2.9, NA, 0.6, 0....
## $ sleep_cycle  <dbl> NA, NA, NA, 0.1333333, 0.6666667, 0.7666667, 0.3833333...
## $ awake        <dbl> 11.9, 7.0, 9.6, 9.1, 20.0, 9.6, 15.3, 17.0, 13.9, 21.0...
## $ brainwt      <dbl> NA, 0.01550, NA, 0.00029, 0.42300, NA, NA, NA, 0.07000...
## $ bodywt       <dbl> 50.000, 0.480, 1.350, 0.019, 600.000, 3.850, 20.490, 0...
```

```
msleep_nona <- msleep %>%
  mutate(conservation = replace_na(conservation, "unknown"))
```

```
glimpse(msleep_nona)
```

```
## Rows: 83
## Columns: 11
## $ name         <chr> "Cheetah", "Owl monkey", "Mountain beaver", "Greater s...
## $ genus        <chr> "Acinonyx", "Aotus", "Aplodontia", "Blarina", "Bos", "...
```

```
## $ vore         <chr> "carni", "omni", "herbi", "omni", "herbi", "herbi", "c...
## $ order        <chr> "Carnivora", "Primates", "Rodentia", "Soricomorpha", "...
## $ conservation <chr> "lc", "unknown", "nt", "lc", "domesticated", "unknown"...
## $ sleep_total  <dbl> 12.1, 17.0, 14.4, 14.9, 4.0, 14.4, 8.7, 7.0, 10.1, 3.0...
## $ sleep_rem    <dbl> NA, 1.8, 2.4, 2.3, 0.7, 2.2, 1.4, NA, 2.9, NA, 0.6, 0....
## $ sleep_cycle  <dbl> NA, NA, NA, 0.1333333, 0.6666667, 0.7666667, 0.3833333...
## $ awake        <dbl> 11.9, 7.0, 9.6, 9.1, 20.0, 9.6, 15.3, 17.0, 13.9, 21.0...
## $ brainwt      <dbl> NA, 0.01550, NA, 0.00029, 0.42300, NA, NA, NA, 0.07000...
## $ bodywt       <dbl> 50.000, 0.480, 1.350, 0.019, 600.000, 3.850, 20.490, 0...
```

Tip - Selecting columns based on regex

```
msleep %>%
  select(matches("wt")) %>%
  glimpse
```

```
## Rows: 83
## Columns: 2
## $ brainwt <dbl> NA, 0.01550, NA, 0.00029, 0.42300, NA, NA, NA, 0.07000, 0.0...
## $ bodywt  <dbl> 50.000, 0.480, 1.350, 0.019, 600.000, 3.850, 20.490, 0.045,...
```

Tip - selecting by discrete number of categories

```
msleep %>%
  select_if(~n_distinct(.) < 6)
```

```
## # A tibble: 83 x 1
##    vore
##    <chr>
##  1 carni
##  2 omni
##  3 herbi
##  4 omni
##  5 herbi
##  6 herbi
##  7 carni
##  8 <NA>
##  9 carni
## 10 herbi
## # ... with 73 more rows
```

```
unique(msleep$vore)
```

```
## [1] "carni"   "omni"    "herbi"   NA        "insecti"
```

My favourite function count

```
msleep %>%
  count(vore)
```

```
## # A tibble: 5 x 2
##   vore        n
##   <chr>   <int>
## 1 carni      19
## 2 herbi      32
## 3 insecti     5
## 4 omni       20
## 5 <NA>        7
```

Tip - add count

```
msleep %>%
  select(name:vore) %>%
  add_count(vore)
```

```
## # A tibble: 83 x 4
##    name                        genus       vore      n
##    <chr>                       <chr>       <chr> <int>
##  1 Cheetah                     Acinonyx    carni    19
##  2 Owl monkey                  Aotus       omni     20
##  3 Mountain beaver             Aplodontia  herbi    32
##  4 Greater short-tailed shrew Blarina     omni     20
##  5 Cow                         Bos         herbi    32
##  6 Three-toed sloth            Bradypus    herbi    32
##  7 Northern fur seal           Callorhinus carni    19
##  8 Vesper mouse                Calomys     <NA>      7
##  9 Dog                         Canis       carni    19
## 10 Roe deer                    Capreolus   herbi    32
## # ... with 73 more rows
```

Tip - get rid of extra characters in column names

```
msleep_nona <- msleep_nona %>%
  select(1:4)

colnames(msleep_nona) <- c("Q1 Name", "Q2 sleeP total 1", "Q3 voore", "Q4 order")
colnames(msleep_nona)
```

```
## [1] "Q1 Name"          "Q2 sleeP total 1" "Q3 voore"         "Q4 order"
```

```
msleep_nona %>%
  select_all(~str_replace(., "Q[0-9]+ ", "")) %>%
  select_all(~str_replace_all(., " ", "_"))
```

```
## # A tibble: 83 x 4
##    Name                         sleeP_total_1 voore order
##    <chr>                        <chr>         <chr> <chr>
##  1 Cheetah                      Acinonyx      carni Carnivora
##  2 Owl monkey                   Aotus         omni  Primates
##  3 Mountain beaver              Aplodontia    herbi Rodentia
##  4 Greater short-tailed shrew Blarina       omni  Soricomorpha
##  5 Cow                          Bos           herbi Artiodactyla
##  6 Three-toed sloth             Bradypus      herbi Pilosa
##  7 Northern fur seal            Callorhinus   carni Carnivora
##  8 Vesper mouse                 Calomys       <NA>  Rodentia
##  9 Dog                          Canis         carni Carnivora
## 10 Roe deer                     Capreolus     herbi Artiodactyla
## # ... with 73 more rows
```

Trick use Janitor to clean names

```
library(janitor)
```

```
##
## Attaching package: 'janitor'
```

```
## The following objects are masked from 'package:stats':
##
##     chisq.test, fisher.test
```

```
msleep_nona %>%
  janitor::clean_names()
```

```
## # A tibble: 83 x 4
##    q1_name                      q2_slee_p_total_1 q3_voore q4_order
##    <chr>                        <chr>             <chr>    <chr>
##  1 Cheetah                      Acinonyx          carni    Carnivora
##  2 Owl monkey                   Aotus             omni     Primates
##  3 Mountain beaver              Aplodontia        herbi    Rodentia
##  4 Greater short-tailed shrew Blarina           omni     Soricomorpha
##  5 Cow                          Bos               herbi    Artiodactyla
##  6 Three-toed sloth             Bradypus          herbi    Pilosa
##  7 Northern fur seal            Callorhinus       carni    Carnivora
##  8 Vesper mouse                 Calomys           <NA>     Rodentia
##  9 Dog                          Canis             carni    Carnivora
## 10 Roe deer                     Capreolus         herbi    Artiodactyla
## # ... with 73 more rows
```

Tip - Random selection of rows

```
set.seed(123)

msleep %>%
  sample_frac(0.1)
```

```
## # A tibble: 8 x 11
##   name  genus vore  order conservation sleep_total sleep_rem sleep_cycle awake
##   <chr> <chr> <chr> <chr> <chr>              <dbl>     <dbl>       <dbl> <dbl>
## 1 Pilo~ Glob~ carni Ceta~ cd                   2.7       0.1      NA      21.4
## 2 Tree~ Tupa~ omni  Scan~ <NA>                 8.9       2.6       0.233  15.1
## 3 Tiger Pant~ carni Carn~ en                  15.8      NA        NA       8.2
## 4 Chin~ Chin~ herbi Rode~ domesticated        12.5       1.5       0.117  11.5
## 5 East~ Scal~ inse~ Sori~ lc                   8.4       2.1       0.167  15.6
## 6 Hous~ Mus   herbi Rode~ nt                  12.5       1.4       0.183  11.5
## 7 Chim~ Pan   omni  Prim~ <NA>                 9.7       1.4       1.42   14.3
## 8 Litt~ Myot~ inse~ Chir~ <NA>                19.9       2         0.2     4.1
## # ... with 2 more variables: brainwt <dbl>, bodywt <dbl>
```

Trick - separate one column into two

```
df <- data.frame(x = c("a:1", "a:2", "c:4", "d", NA))
(df)
```

```
##      x
## 1  a:1
## 2  a:2
## 3  c:4
## 4    d
## 5 <NA>
```

```
df %>% separate(x, c("key","value"), ":")
```

```
## Warning: Expected 2 pieces. Missing pieces filled with `NA` in 1 rows [4].
```

```
##    key value
## 1    a     1
## 2    a     2
## 3    c     4
## 4    d  <NA>
## 5 <NA>  <NA>
```

Trick near()

```
msleep %>%
  select(name, sleep_total)
```

```
## # A tibble: 83 x 2
##    name                          sleep_total
##    <chr>                               <dbl>
##  1 Cheetah                              12.1
##  2 Owl monkey                           17
##  3 Mountain beaver                      14.4
##  4 Greater short-tailed shrew           14.9
##  5 Cow                                   4
##  6 Three-toed sloth                     14.4
##  7 Northern fur seal                     8.7
##  8 Vesper mouse                          7
##  9 Dog                                  10.1
## 10 Roe deer                             3
## # ... with 73 more rows
```

```
sd(msleep$sleep_total)
```

```
## [1] 4.450357
```

```
msleep %>%
  select(name, sleep_total) %>%
  filter(near(sleep_total, 17, tol = sd(sleep_total)))
```

```
## # A tibble: 26 x 2
##    name                          sleep_total
##    <chr>                               <dbl>
##  1 Owl monkey                           17
##  2 Mountain beaver                      14.4
##  3 Greater short-tailed shrew           14.9
##  4 Three-toed sloth                     14.4
##  5 Long-nosed armadillo                 17.4
##  6 North American Opossum               18
##  7 Big brown bat                        19.7
##  8 Western american chipmunk            14.9
##  9 Thick-tailed opposum                 19.4
## 10 Mongolian gerbil                     14.2
## # ... with 16 more rows
```

Tip - Use %in% instead of or

```
y <- c("a", "a", "z", "y","b", "c")
```

```
y == "a" | y == "b" | y == "c"
```

```
## [1]  TRUE  TRUE FALSE FALSE  TRUE  TRUE
```

is the same as

```
y %in% c("a", "b", "c")
```

```
## [1]  TRUE  TRUE FALSE FALSE  TRUE  TRUE
```

```
msleep %>%
  select(order, name, sleep_total) %>%
  filter(order %in% c("Didelphimorphia", "Diprotodontia"))
```

```
## # A tibble: 4 x 3
##   order           name                   sleep_total
##   <chr>           <chr>                        <dbl>
## 1 Didelphimorphia North American Opossum        18
## 2 Didelphimorphia Thick-tailed opposum          19.4
## 3 Diprotodontia   Phalanger                     13.7
## 4 Diprotodontia   Potoroo                       11.1
```

Trick - Use %in% instead of or

```
msleep %>%
  select(order, name, sleep_total) %>%
  filter(order %in% str_subset(order, "Di"))
```

```
## # A tibble: 4 x 3
##   order           name                   sleep_total
##   <chr>           <chr>                        <dbl>
## 1 Didelphimorphia North American Opossum        18
## 2 Didelphimorphia Thick-tailed opposum          19.4
## 3 Diprotodontia   Phalanger                     13.7
## 4 Diprotodontia   Potoroo                       11.1
```

```
# add negative for second example
```

Tip - vars all_vars

```
msleep %>%
  select(name, sleep_total:sleep_cycle) %>%
  filter_at(vars(sleep_total, sleep_rem), all_vars(. > 5))
```

```
## # A tibble: 2 x 4
##   name                sleep_total sleep_rem sleep_cycle
##   <chr>                     <dbl>     <dbl>       <dbl>
## 1 Thick-tailed opposum        19.4       6.6          NA
## 2 Giant armadillo             18.1       6.1          NA
```

Advanced

*Writing code

Tip - Avoid dots in names

Believe me, or watch Jim Hester video [7]

[7] https://www.youtube.com/watch?v=IoWDQ6rx6yA

Working with BIG files

Use when data is larger than 1 Giga

Tip - Data table package

```
install.packages("data.table")
```

use the `fread()` function to read in big files

Tip - Use fread from data.table and filter with grep

This is amazing!

```
data <- data.table::fread("grep -w File ~/data/someHUGEfile.csv")
```

pre-filter with the grep command ;)

Trick - Use the pipe from base and filter

Trick - Compress files on the fly!

```
write.csv(data, gzfile("data/bigdata.gz"))
file <- read.csv(gzfile("data/bigdata.gz"))
```

Tip - Read or write big outputs

Functions in order of speed

```
read.csv()
readr::read_csv()
vroom::vroom_read()
```

Tip - Read or Write to compressed formats

Compress directly into *gz, 7z, zst*. We know that `write.csv` from base does a pretty good job for most smallish files. Try `readr::write_csv` and you will see the a 50% improvement. But, what surpasses all is `vroom::vroom_write` ~ 15 X faster than `write.csv`.

Tip - benchmark

When things are working the next step is optimise! You can check yourself with the function `bench_time` from the {bench} package.

*Workflows and Reproducibility

Tip - Use workflowr

Demonstration of a {workflowr} website [8] Milestones, versions, all in one place.

[8] https://www.youtube.com/watch?v=01wv94sZfvE

Tip - learn drake

If you are building analysis code that is likely to grow use drake The {drake} package records file inter dependencies in your analysis. When files are changed, {drake} only re-runs the parts that need to be re-run. This saves time and reduces errors ^[Learn from Matt Dray https://www.rostrum.blog/2019/07/23/can-drake-rap/ and from Amanda Dobbyn Rladies Chicago https://aedobbyn.github.io/nyc-fires/index.html#1].

Keep learning

Resources used to provide you with this collection of tips and tricks

- Suzan's tidyverse tricks https://suzan.rbind.io/categories/tutorial/
- Sean Lopp's posts https://rviews.rstudio.com/categories/tips-and-tricks/
- Sean Lopp's video NYC RStudio Conference https://www.youtube.com/watch?v=kuSQgswZdr8
- Jim Hester's video Pipe Connections https://www.youtube.com/watch?v=RYhwZW6ofbI
- tidyr reference docs https://tidyr.tidyverse.org/reference

Author: Paula Andrea Martinez 2020-11-09