Writing R packages in Rstudio

https://stirlingcodingclub.github.io/SCC_R_package/notebook/Rpackage_notes.html

Brad Duthie

6 March 2019

Introduction: What is an R package?

- Bundles of code and data that can be written by anyone in the R community

Introduction: What is an R package?

- Bundles of code and data that can be written by anyone in the R community
- Held in online repositories
  - Comprehensive R Archive Network (CRAN) (`install.packages()`)
  - GitHub (`devtools::install_github()`)

Introduction: What is an R package?

- Bundles of code and data that can be written by anyone in the R community
- Held in online repositories
  - Comprehensive R Archive Network (CRAN) (`install.packages()`)
  - GitHub (`devtools::install_github()`)
- CRAN currently holds over 13000 R packages

Introduction: What is an R package?

- Bundles of code and data that can be written by anyone in the R community
- Held in online repositories
  - Comprehensive R Archive Network (CRAN) (`install.packages()`)
  - GitHub (`devtools::install_github()`)
- CRAN currently holds over 13000 R packages
- R packages are wide ranging
  - Massive multi-person development efforts for large scale coding projects
  - Small joke packages with minimal code
  - Personal packages customised for an individual user

Two packages that need to be installed

The `devtools` package:

> *Devtools makes package development a breeze: it works with R's existing conventions for code structure, adding efficient tools to support the cycle of package development. With devtools, developing a package becomes so easy that it will be your default layout whenever you're writing a significant amount of code.*

Two packages that need to be installed

The `devtools` package:

> *Devtools makes package development a breeze: it works with R's existing conventions for code structure, adding efficient tools to support the cycle of package development. With devtools, developing a package becomes so easy that it will be your default layout whenever you're writing a significant amount of code.*

The `roxygen2` package:

> *The goal of roxygen2 is to make documenting your code as easy as possible. R provides a standard way of documenting packages: you write .Rd files in the man/ directory.*

The most basic R package

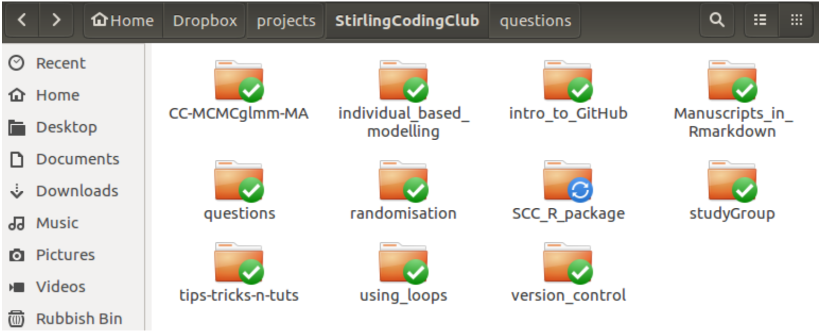
Goal: Create an R package that includes two functions.

```r
# Convert Fahrenheit to Celsius
F_to_C <- function(F_temp){
    C_temp <- (F_temp - 32) * 5/9;
    return(C_temp);
}

# Convert Celsius to Fahrenheit
C_to_F <- function(C_temp){
    F_temp <- (C_temp * 9/5) + 32;
    return(F_temp);
}
```

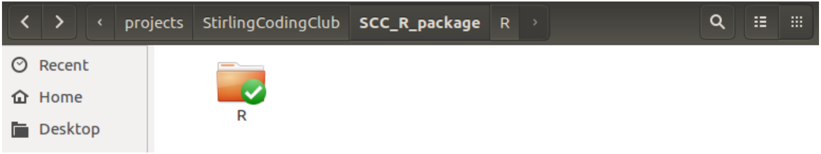
The most basic R package

First, create a new folder in your computer



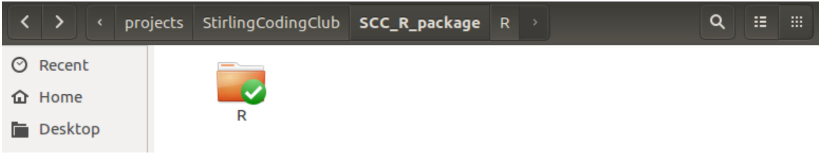
The most basic R package

Next, add a folder called 'R'

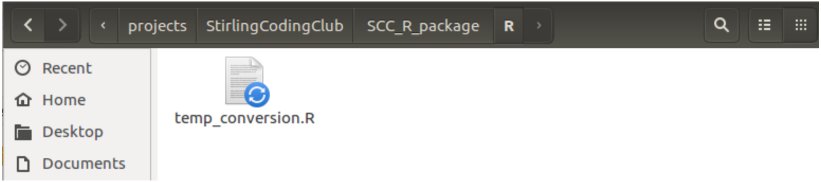


The most basic R package

Next, add a folder called 'R'



Then drop the R scripts into the 'R' folder



The most basic R package

Add a file called `DESCRIPTION` to the outer directory. This is a plain text file that can just be a few lines of code.

```
Package: SCCTempConverter
Type: Package
Title: Temperature Conversion Package for Demonstration
Version: 0.0.1.0
RoxygenNote: 6.1.0
```

It can also include a lot more information (e.g., authors, maintainers, extended description, website, etc.) if need be.

The most basic R package

This is now *already* an R package, which we can load.

```r
# Working dir should be SCC_R_package
library(devtools);
load_all(".");
```

The most basic R package

This is now *already* an R package, which we can load.

```
# Working dir should be SCC_R_package
library(devtools);
load_all(".");
```

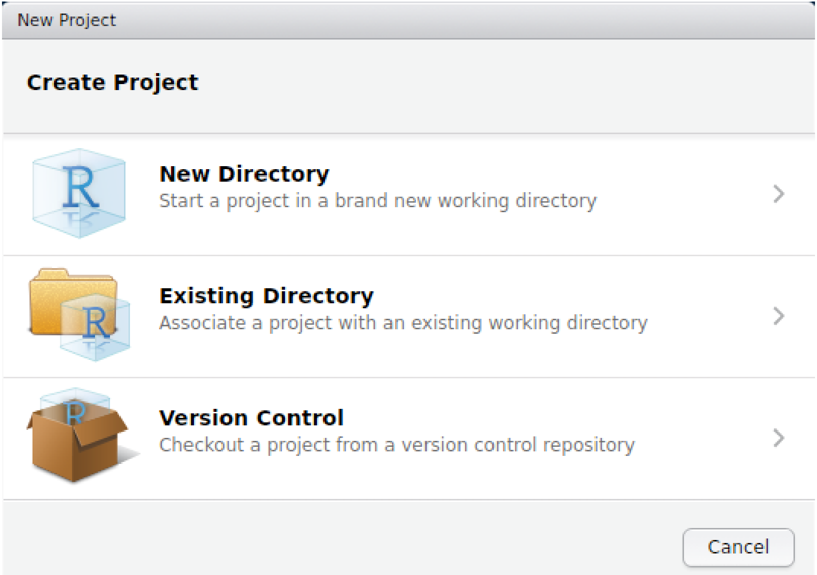
Once loaded, we can start using the R functions in the package.

```
C_to_F(20);
```

```
## [1] 68
```

Creating a new R project

To do this in Rstudio, go to `File > New Project...`



New Project

Create Project

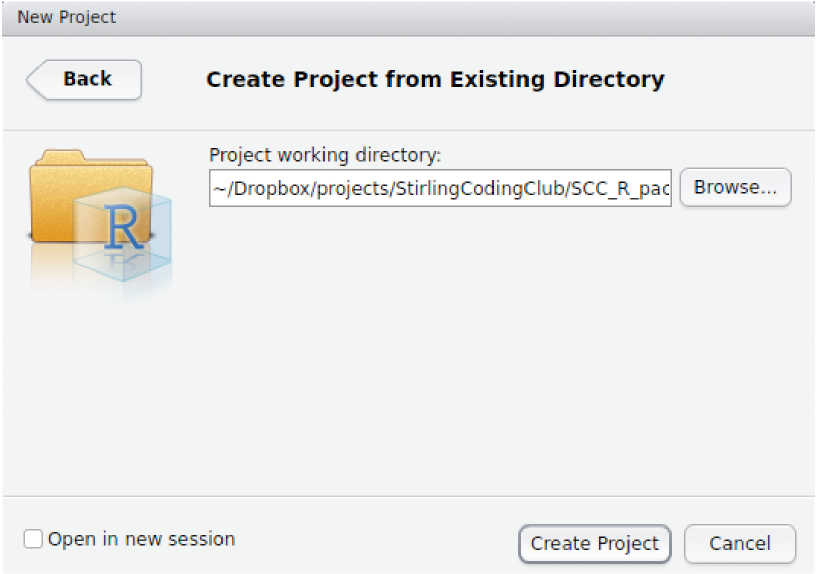
New Directory
Start a project in a brand new working directory

Existing Directory
Associate a project with an existing working directory

Version Control
Checkout a project from a version control repository

Creating a new R project

To do this in Rstudio, go to `File > New Project...`



New Project

Create Project from Existing Directory

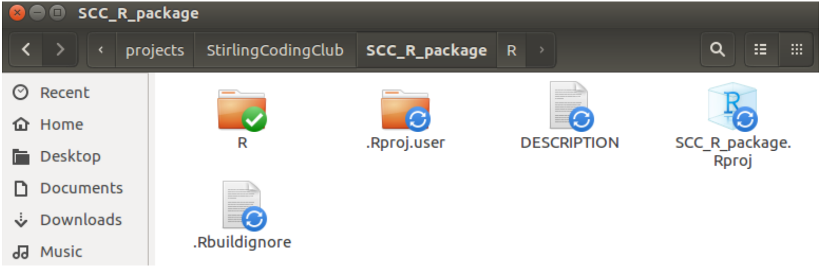
Project working directory:

~/Dropbox/projects/StirlingCodingClub/SCC_R_pac

□ Open in new session

Creating a new R project

To do this in Rstudio, go to `File > New Project...`



Adding some minimal documentation

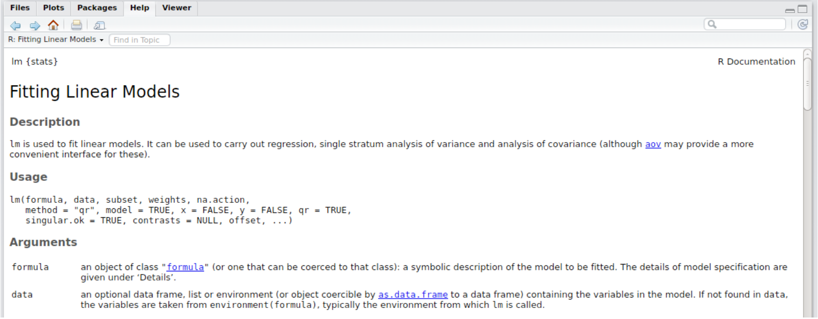
Documentation shows up in the ‘Help’ tab of RStudio when running the function `help`

```
help(lm);
```

Note that the code below does the same thing as the code above.

```
?lm
```

Adding some minimal documentation



lm {stats} R Documentation

Fitting Linear Models

Description

`lm` is used to fit linear models. It can be used to carry out regression, single stratum analysis of variance and analysis of covariance (although aov may provide a more convenient interface for these).

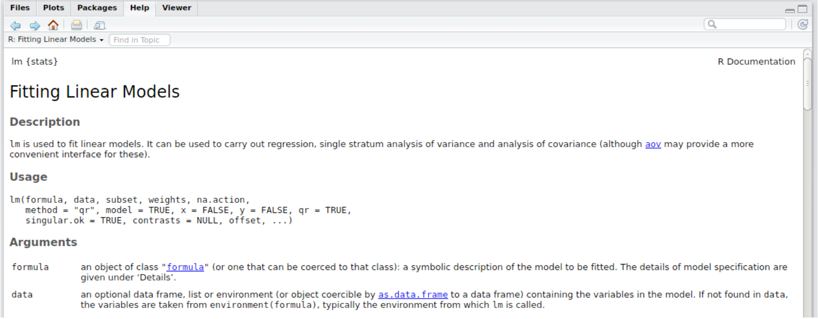
Usage

```
lm(formula, data, subset, weights, na.action,
   method = "qr", model = TRUE, x = FALSE, y = FALSE, qr = TRUE,
   singular.ok = TRUE, contrasts = NULL, offset, ...)
```

Arguments

`formula`	an object of class "`formula`" (or one that can be coerced to that class): a symbolic description of the model to be fitted. The details of model specification are given under 'Details'.
`data`	an optional data frame, list or environment (or object coercible by `as.data.frame` to a data frame) containing the variables in the model. If not found in data, the variables are taken from `environment(formula)`, typically the environment from which `lm` is called.

Adding some minimal documentation



lm {stats} R Documentation

Fitting Linear Models

Description

lm is used to fit linear models. It can be used to carry out regression, single stratum analysis of variance and analysis of covariance (although aov may provide a more convenient interface for these).

Usage

```
lm(formula, data, subset, weights, na.action,
   method = "qr", model = TRUE, x = FALSE, y = FALSE, qr = TRUE,
   singular.ok = TRUE, contrasts = NULL, offset, ...)
```

Arguments

formula	an object of class "formula" (or one that can be coerced to that class): a symbolic description of the model to be fitted. The details of model specification are given under 'Details'.
data	an optional data frame, list or environment (or object coercible by as.data.frame to a data frame) containing the variables in the model. If not found in data, the variables are taken from environment(formula), typically the environment from which lm is called.

Create help files in Rstudio using the roxygen2 package

```
install.packages("roxygen2");
library(roxygen2);
```

Adding some minimal documentation

```r
#' Fahrenheit conversion
#'
#' Convert degrees Fahrenheit temp to degrees Celsius
#' @param F_temp The temperature in degrees Fahrenheit
#' @return The temperature in degrees Celsius
#' @examples
#' temp1 <- F_to_C(50);
#' temp2 <- F_to_C( c(50, 63, 23) );
#' @export
F_to_C <- function(F_temp){
    C_temp <- (F_temp - 32) * 5/9;
    return(C_temp);
}
```

Adding some minimal documentation

If we load our package and type the following:

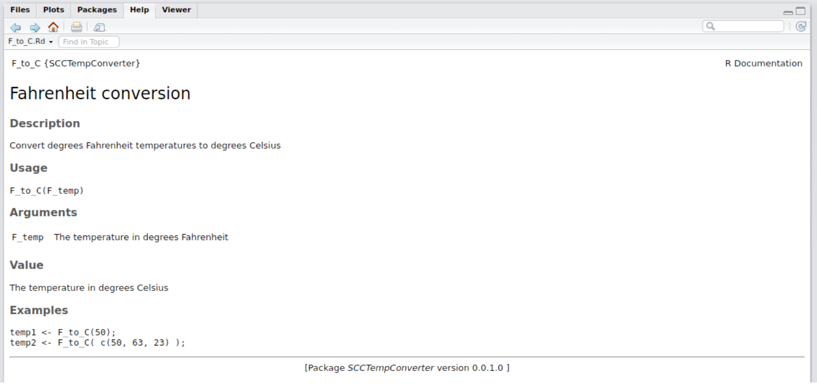
```
help(F_to_C); # ?F_to_C also works
```

Adding some minimal documentation

If we load our package and type the following:

```
help(F_to_C); # ?F_to_C also works
```

We get the below in RStudio



F_to_C {SCCTempConverter} R Documentation

Fahrenheit conversion

Description

Convert degrees Fahrenheit temperatures to degrees Celsius

Usage

```
F_to_C(F_temp)
```

Arguments

F_temp The temperature in degrees Fahrenheit

Value

The temperature in degrees Celsius

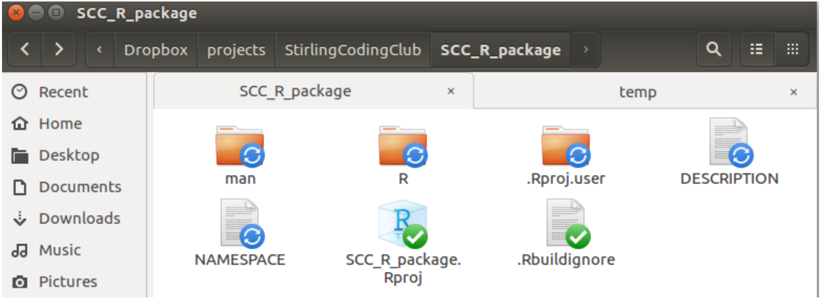
Examples

```
temp1 <- F_to_C(50);
temp2 <- F_to_C( c(50, 63, 23) );
```

[Package *SCCTempConverter* version 0.0.1.0]

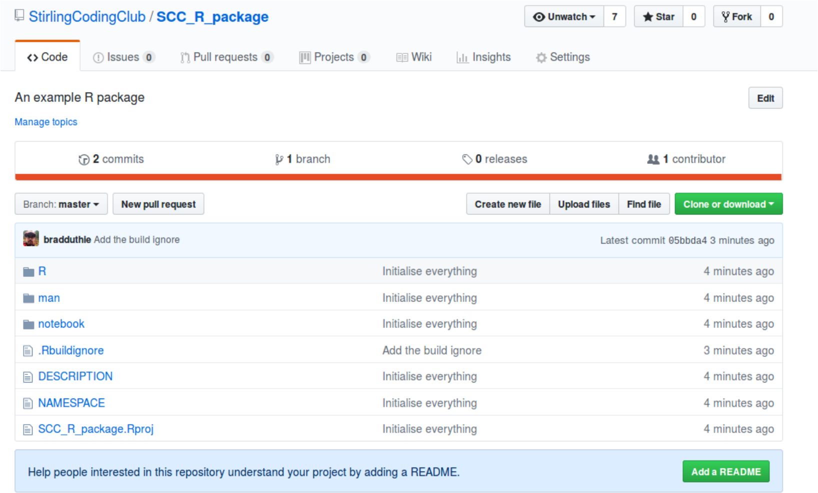
Adding some minimal documentation

Note that a new folder has been added to the directory



Upload the package to GitHub

See notes on version control for help



An example R package

2 commits | 1 branch | 0 releases | 1 contributor

braddthle Add the build ignore — Latest commit 05bbda4 3 minutes ago

R	Initialise everything	4 minutes ago
man	Initialise everything	4 minutes ago
notebook	Initialise everything	4 minutes ago
.Rbuildignore	Add the build ignore	3 minutes ago
DESCRIPTION	Initialise everything	4 minutes ago
NAMESPACE	Initialise everything	4 minutes ago
SCC_R_package.Rproj	Initialise everything	4 minutes ago

Help people interested in this repository understand your project by adding a README.

Upload the package to GitHub

Anyone can download it by using the install_github function in thedevtools

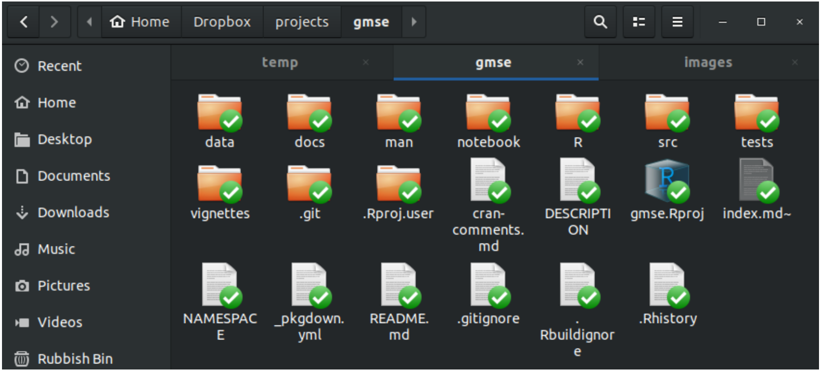
```r
library(devtools);
install_github("StirlingCodingClub/SCC_R_package");
```

Our R package is now installed. We can start using it by reading it in as a normal package.

```r
library(SCCTempConverter);
F_to_C(30);
```

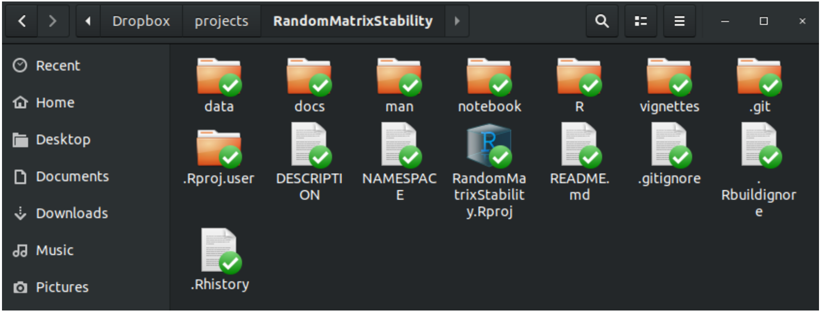
The R package design is very useful

A reasonably large R package directory



The R package design is very useful

A project for a manuscript in the R package style



Today's resources and additional help

Slides: https://stirlingcodingclub.github.io/SCC_R_package/notebook/Rpackage_slides.pdf

Notes: https://stirlingcodingclub.github.io/SCC_R_package/notebook/Rpackage_notes.html

From Karl Broman

- The minimal R package
- Building and installing an R package
- Writing documentation with Roxygen2

From RStudio

- R packages (free online book).
- pkgdown (automatically builds package website)