




The rocky road to building my first R package

Lucy Morgan



What you need

□ You need a clear view of what you want your package to do

1. Is it just for your own use?
2. How might other people use it?
3. What inputs will it need?
4. And so on...

TIP 1

Most people come to write their first package when they have some code they want to share and make available. This can make building your first package stressful... perhaps start with a toy package!



Things you actually need

▣ There are two packages you'll need to install to create a basic Rpackage

1. devtools
2. roxygen2

For installation use

- ▣ `install.packages("devtools")`
- ▣ `devtools::install_github("klutometis/roxygen")`

And to load use `library("Package Name")`

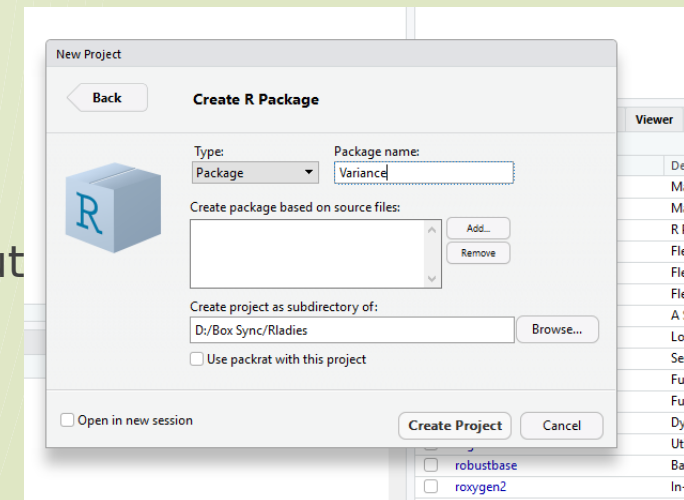
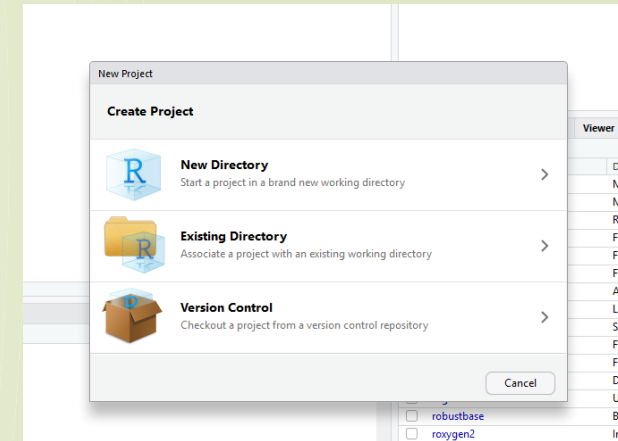
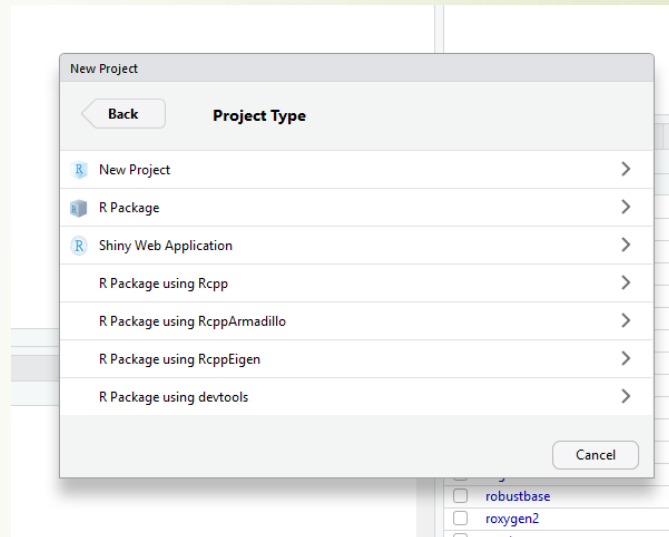
Create the package directory

File > New Project ... > New Directory

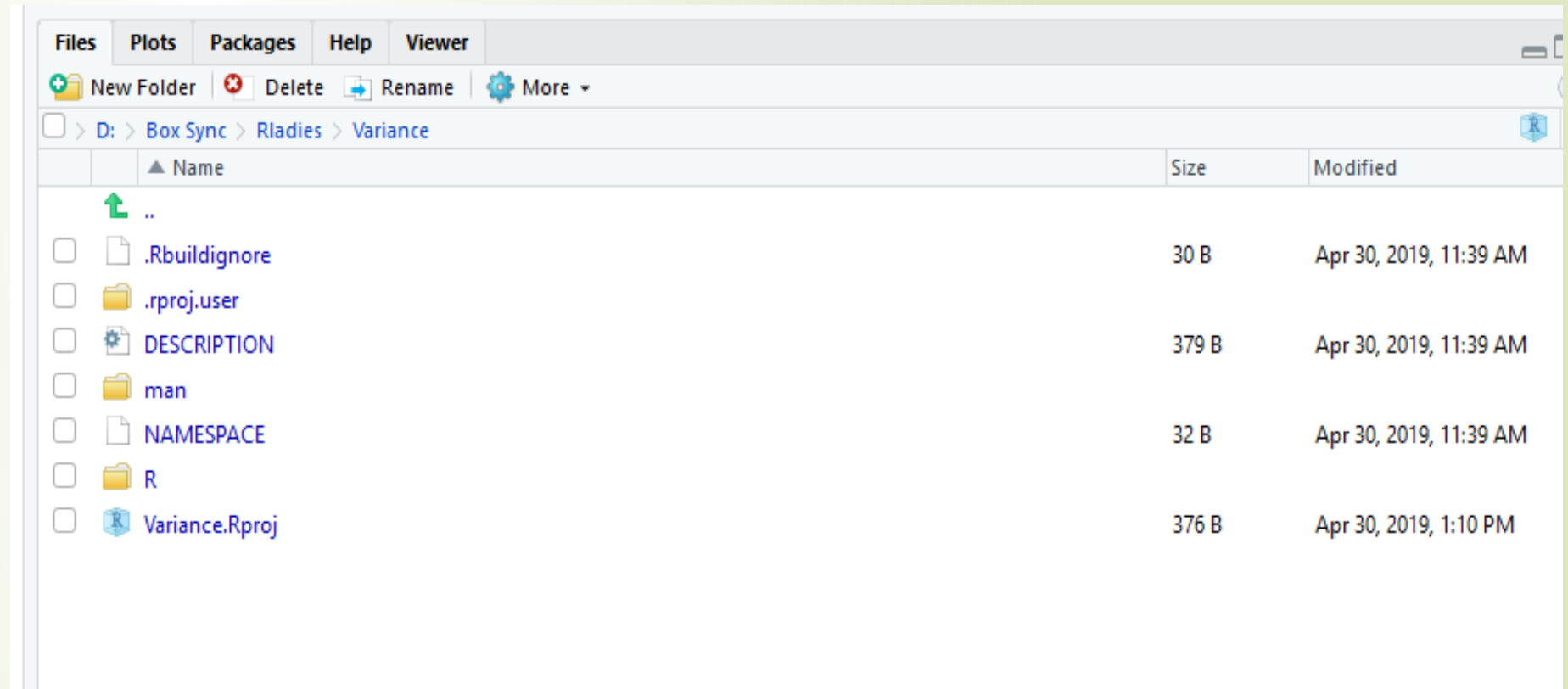
R Package

Name your package and where you want to put it. The source files are the .R files that contain your code.

OR use `package.skeleton("newpackagename")`



The package directory



	Name	Size	Modified
	..		
	.Rbuildignore	30 B	Apr 30, 2019, 11:39 AM
	.rproj.user		
	DESCRIPTION	379 B	Apr 30, 2019, 11:39 AM
	man		
	NAMESPACE	32 B	Apr 30, 2019, 11:39 AM
	R		
	Variance.Rproj	376 B	Apr 30, 2019, 1:10 PM

DESCRIPTION

```
DESCRIPTION
1 Package: Variance
2 Type: Package
3 Title: calculating the variance of an array
4 Version: 0.1.0
5 Author: Dr L. E. Morgan
6 Maintainer: Dr L. E. Morgan <l.e.morgan@lancaster.ac.uk>
7 Description: This package takes an array of data values and calculates the variance.
8 License: Unlimited
9 |
```

The description file holds important information about the package for example the version, the maintainer and the license.


This DESCRIPTION file can be found by a user by clicking on the package name.

It is displayed on the main package page along with the package manuals, the version and package description.

- [numbers](#)
- [optimParallel](#)
- [philentropy](#)
- [pillar](#)
- [pkgbuild](#)

R: Collection of functions for matrix calculations

Collection of functions for matrix calculations



Documentation for package 'matrixcalc' version 1.0-3

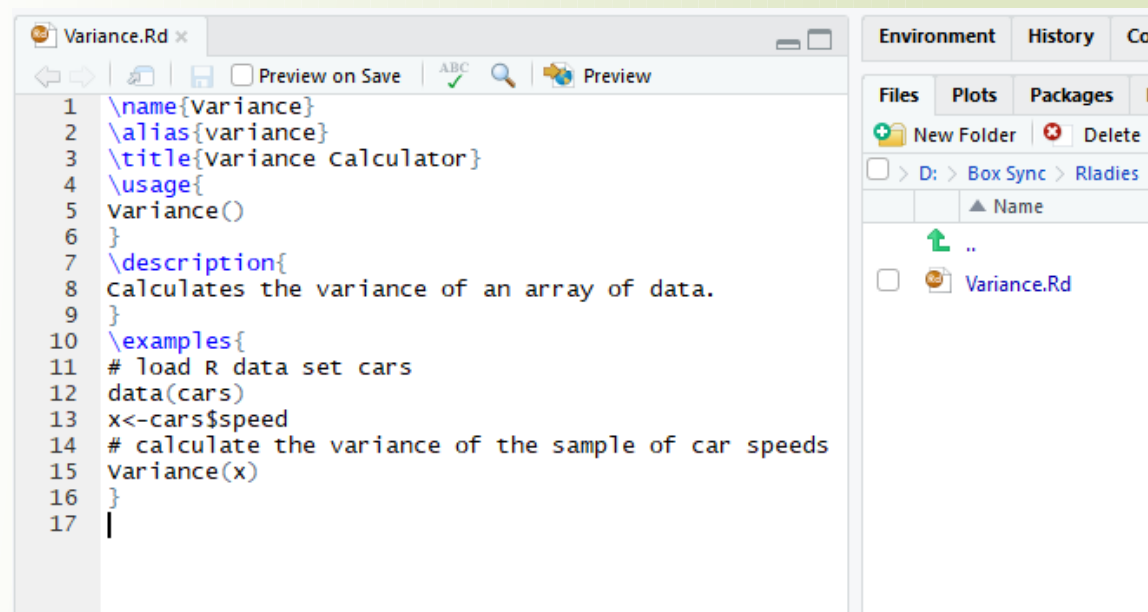
- [DESCRIPTION file.](#)

Help Pages

%s%	Direct sum of two arrays
commutation.matrix	Commutation matrix for r by c numeric matrices
creation.matrix	Creation Matrix

Manual

The manual contains information on the functionality of your package and also allows you to give example



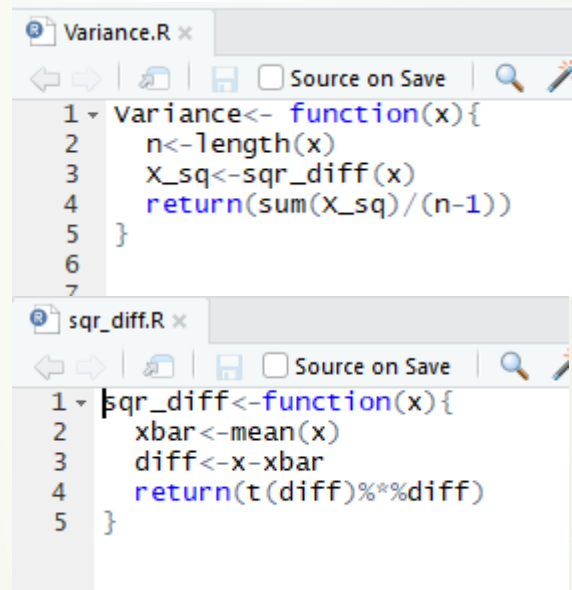
```
1 \name{variance}
2 \alias{variance}
3 \title{Variance Calculator}
4 \usage{
5 variance()
6 }
7 \description{
8 calculates the variance of an array of data.
9 }
10 \examples{
11 # load R data set cars
12 data(cars)
13 x<-cars$speed
14 # calculate the variance of the sample of car speeds
15 variance(x)
16 }
17 |
```

TIP 2

- Briefly filling in the package manual file before you start writing your code can help you plan what your code should do, and keep on track.

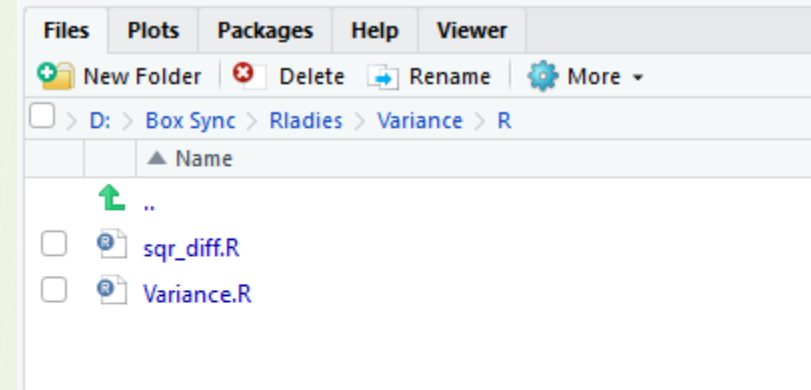
R

- ▮ The R directory of the package is where all R code should be put.



```
Variance.R x
1 Variance<- function(x){
2   n<-length(x)
3   X_sq<-sqr_diff(x)
4   return(sum(X_sq)/(n-1))
5 }
6
7

sqr_diff.R x
1 sqr_diff<-function(x){
2   xbar<-mean(x)
3   diff<-x-xbar
4   return(t(diff)%*%diff)
5 }
```

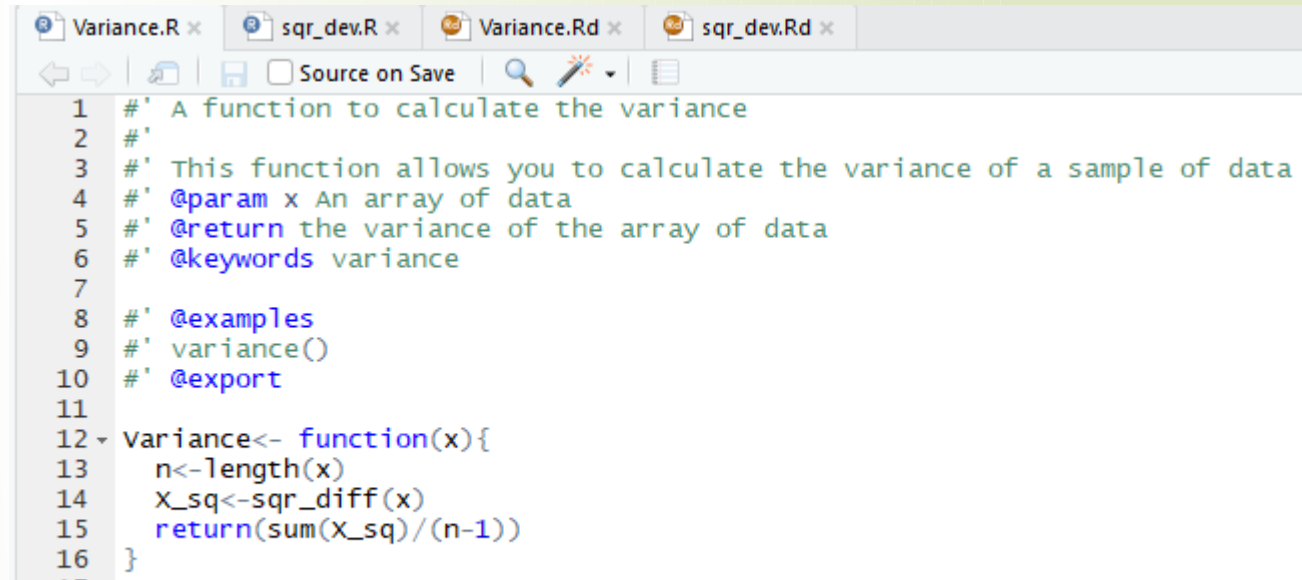


TIP 3

- ▮ Although this is a very simple function most functions are not and it helps to modularise your code into testable chunks.

roxygen

- Roxygen allows you to speed up your package building.
- Speeds up manual building and builds the namespace file.
- Using `#'` tells R it is an roxygen comment



```
Variance.R x  |  sq_dev.R x  |  Variance.Rd x  |  sq_dev.Rd x
Source on Save
1 #' A function to calculate the variance
2 #'
3 #' This function allows you to calculate the variance of a sample of data
4 #' @param x An array of data
5 #' @return the variance of the array of data
6 #' @keywords variance
7
8 #' @examples
9 #' variance()
10 #' @export
11
12 variance<- function(x){
13   n<-length(x)
14   X_sq<-sqr_diff(x)
15   return(sum(X_sq)/(n-1))
16 }
```

When you've added all your comments use the devtools function `document()` to build the namespace and manual files.

[document\(\)](#)

NOTE! Only use `@export` in files you want to be accessible to users

Install your package

- ▢ Set your working directory to the file which contains your package folder
- for me that is the folder that contains the folder “Variance”
- ▢ `Install(“Variance”)`
- ▢ Test your function/s `-> Variance(c(1,2,3,4))`
- ▢ Check your manual/s `-> ?Variance`

- ▢ For your package to appear in the package list you need to compile it
You can either use the build tool bar or `ctrl+shift+b`.

<input type="checkbox"/> <code>utils</code>	The R Utils Package	3.5.1
<input type="checkbox"/> <code>Variance</code>	Calculating the variance of an array	0.1.0
<input type="checkbox"/> <code>vcd</code>	Visualizing Categorical Data	1.4-4



Helpful Resources

I am definitely a novice when it comes to building packages but simple packages can save you a lot of time. Here are some sources that helped me put my first package together

- <https://hilaryparker.com/2014/04/29/writing-an-r-package-from-scratch/> <- a blog post (very simple and straight to the point). Small functions can save you time
- http://web.mit.edu/insong/www/pdf/rpackage_instructions.pdf <- detailed instructions
- <https://support.rstudio.com/hc/en-us/articles/200486488-Developing-Packages-with-RStudio> <- R support pages