Biodiversity data analysis workshop – Day 2

Dominic Henry & Lizanne Roxburgh
Conservation Science Unit

dominich@ewt.org.za
lizanner@ewt.org.za

ENDANGERED WILDLIFE TRUST
Protecting forever, together.

science & technology
Department:
Science and Technology
REPUBLIC OF SOUTH AFRICA

NRF National Research Foundation

SANBI
Biodiversity for Life
South African National Biodiversity Institute

31st October 2019

Workshop goals

ENDANGERED WILDLIFE TRUST

- Introduce foundational concepts
- Provide insight into the potential applications of R
- Overcome the first hurdles of using R
- Provide a solid foundation for data wrangling
- Introduce you to the R community
- Start you on your R journey & learn by doing

Programme – Day 2

ENDANGERED WILDLIFE TRUST

08h30 – 10h00	Session 1 – Introduction to R, RStudio, basics of programming
10h00 – 10h30	Tea break
10h30 – 12h15	Session 2 – Data wrangling with the tidyverse
12h15 – 13h30	Lunch
13h30 – 15h00	Session 3 – Data visualisation using ggplot2
15h00 – 15h30	Tea break
15h30 – 17h00	Session 4 – Handling spatial data in R

What is R?

ENDANGERED WILDLIFE TRUST

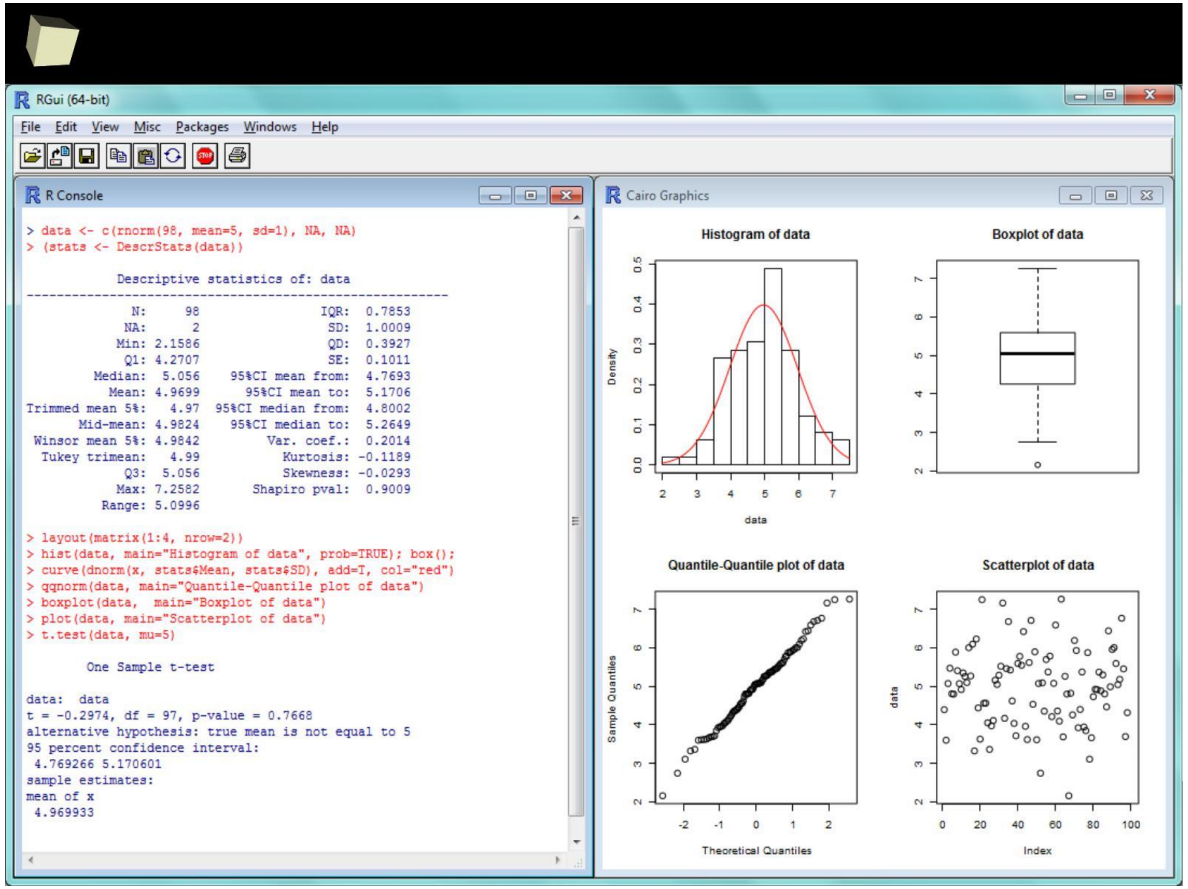
- A high-level functional programming language
- Open source (free!) software environment
- A diverse community
- Driven by the *R Development Core Team*

www.r-project.org

What can R be used for?

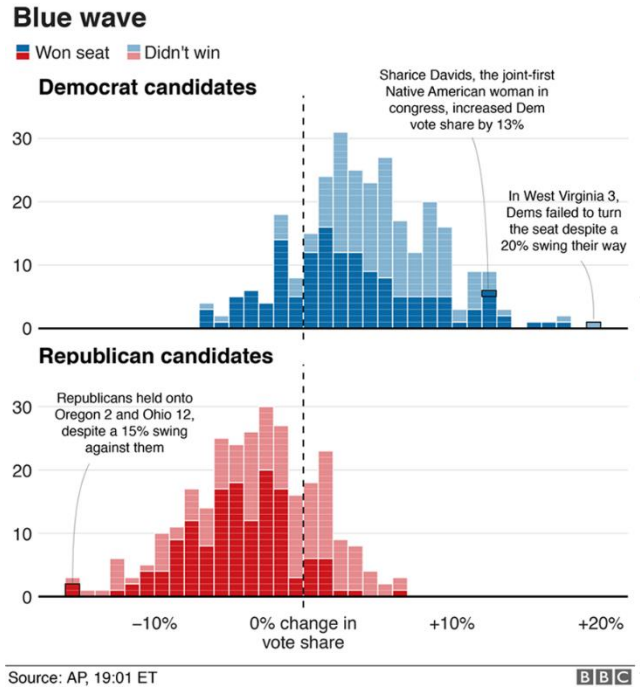
ENDANGERED WILDLIFE TRUST

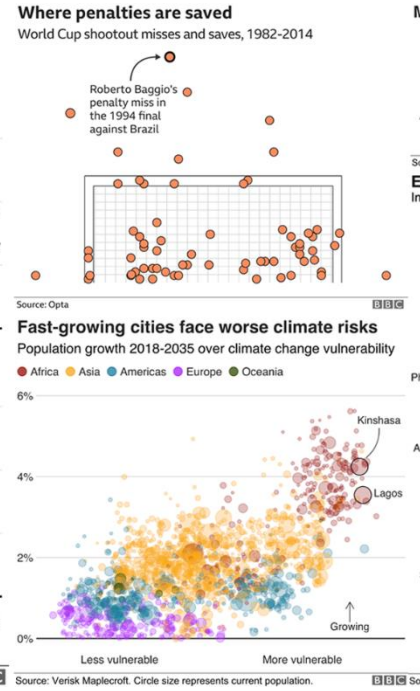
- Statistical programming



What can R be used for?

- Statistical programming
- Data wrangling and visualisation





What can R be used for?

ENDANGERED WILDLIFE TRUST

- Statistical programming
- Data wrangling and visualisation
- Web apps

What can R be used for?

ENDANGERED WILDLIFE TRUST

- Statistical programming
- Data wrangling and visualisation
- Web apps
- Spatial analysis

What can R be used for?

ENDANGERED WILDLIFE TRUST

- Statistical programming
- Data wrangling and visualisation
- Web apps
- Spatial analysis
- Data science

What can R be used for?

ENDANGERED WILDLIFE TRUST

- Statistical programming
- Data wrangling and visualisation
- Web apps
- Spatial analysis
- Data science
- Almost anything...

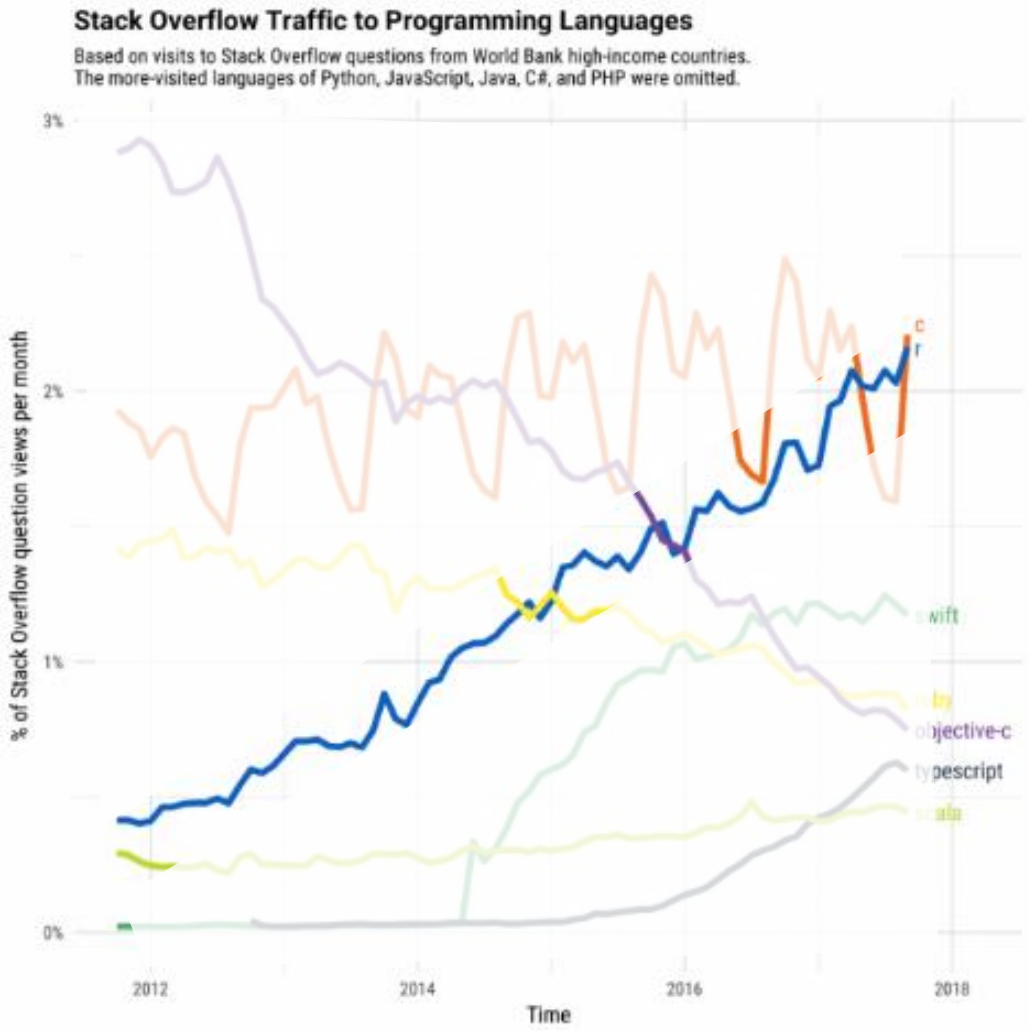
Why should we use R?

ENDANGERED WILDLIFE TRUST

- Repetitive manual tasks are boring, time-consuming and prone to errors
- Allows for reproducible and transferable research, processes and analysis
- Free with incredible user support
- Unlimited (almost) capabilities
- It is a valuable career skill
- It will save you time!

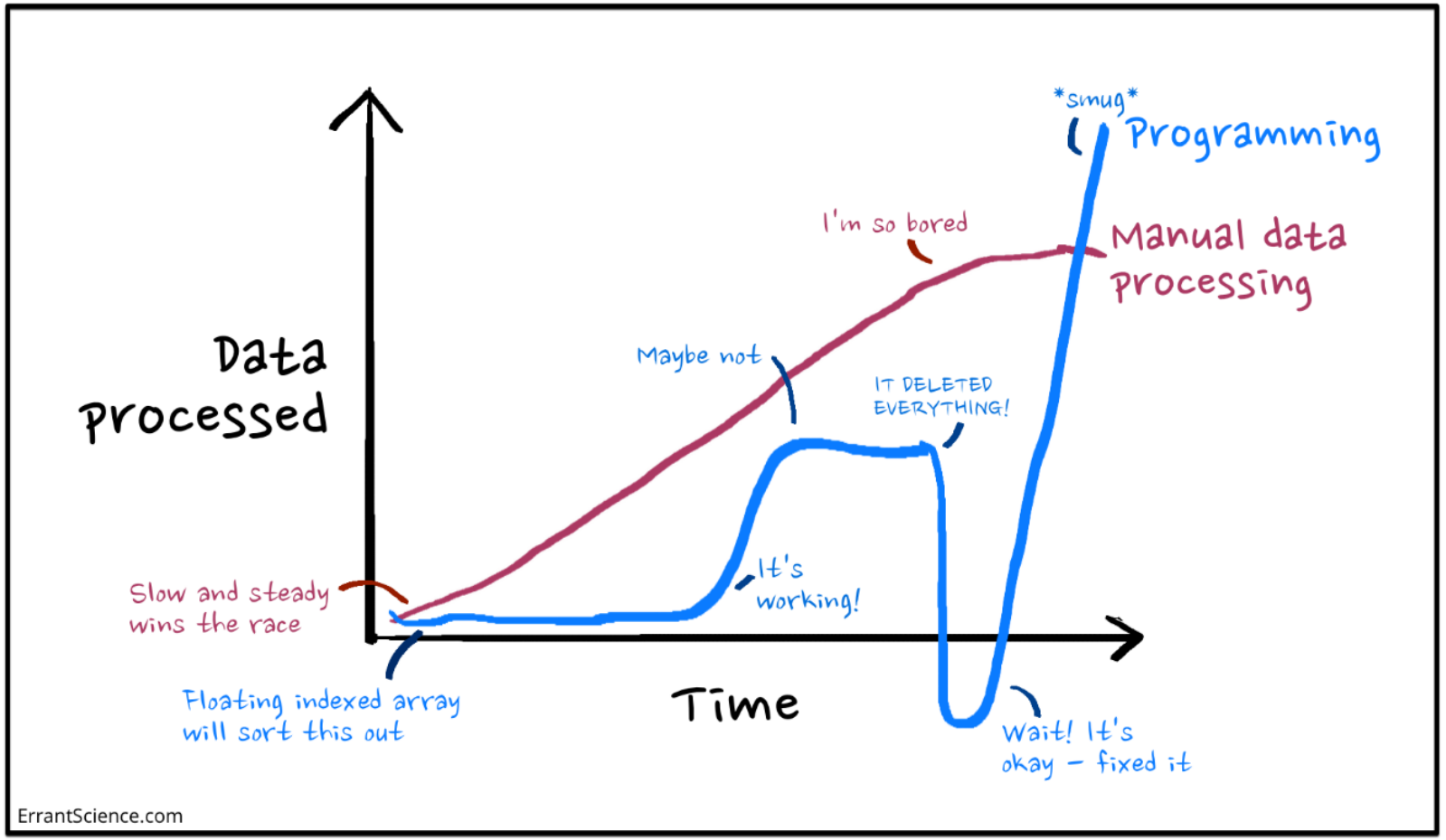
Why should we use R?

ENDANGERED WILDLIFE TRUST



Is it worth the effort?

ENDANGERED WILDLIFE TRUST



Programming vs manual data processing

What is RStudio?

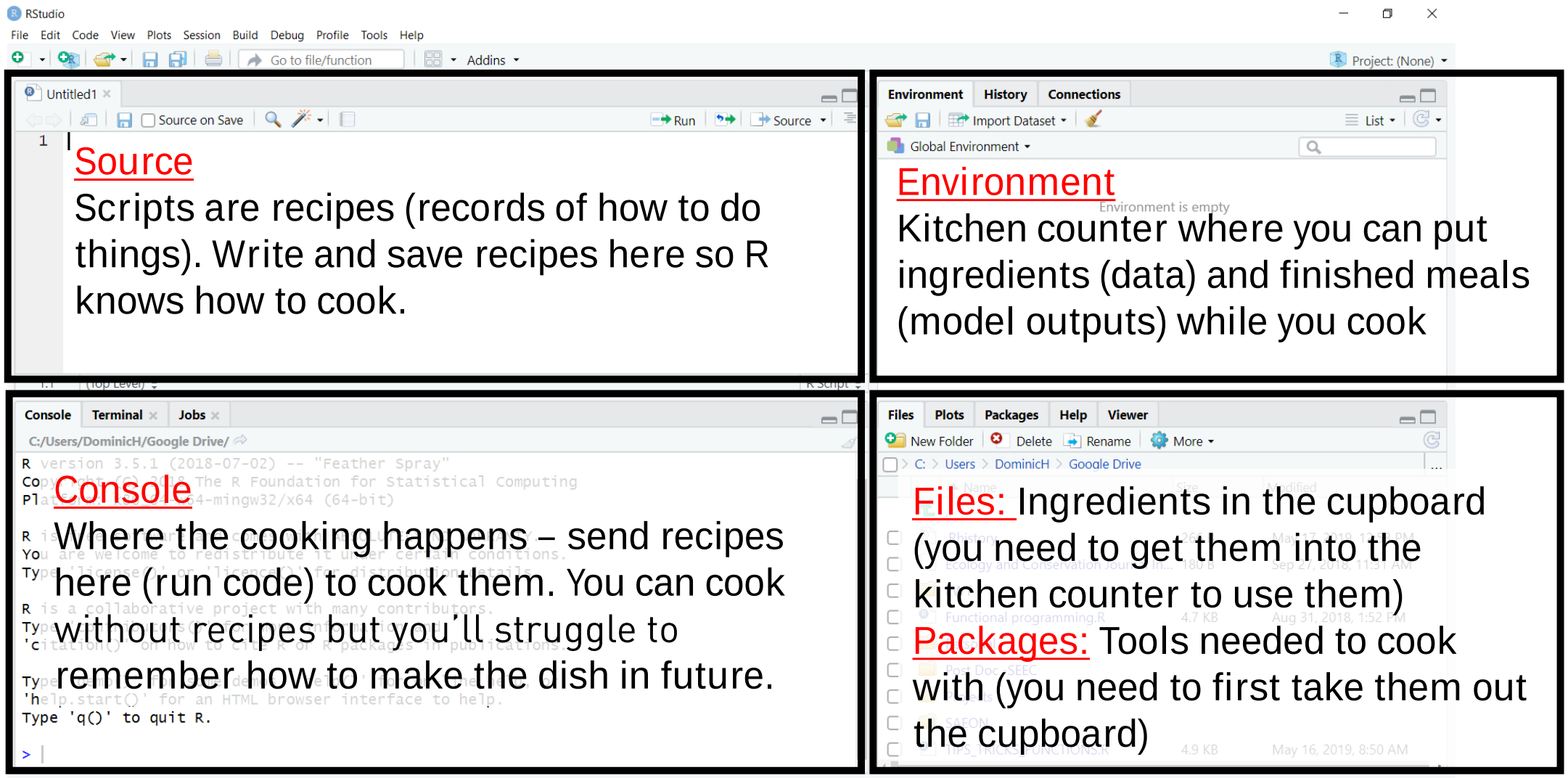
ENDANGERED WILDLIFE TRUST

- Integrated development environment (IDE) for R
- Most popular IDE, free and open-source
- Enhances extends the core capabilities of R
- Provides an easy way to write & save scripts, interact with objects, visualise results and publish end products

RStudio®

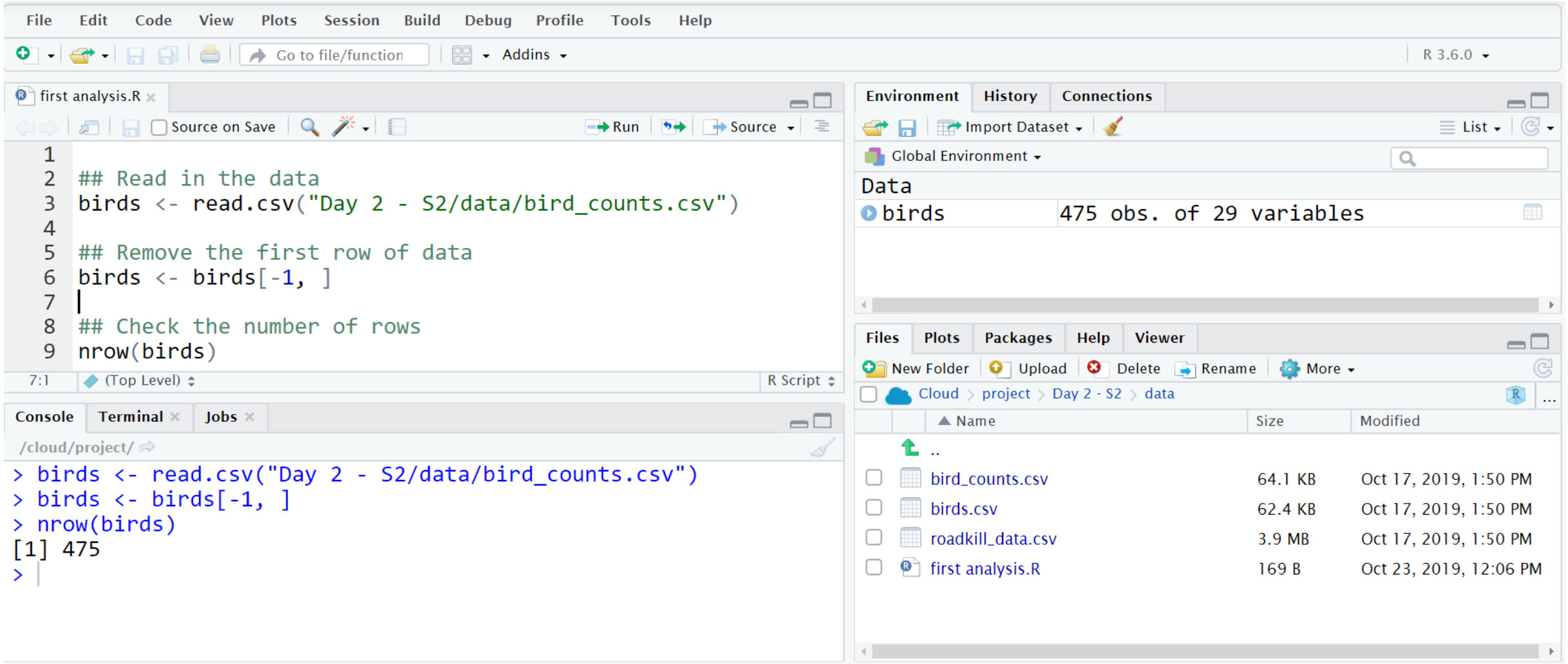
Layout of RStudio

ENDANGERED WILDLIFE TRUST



Layout of RStudio

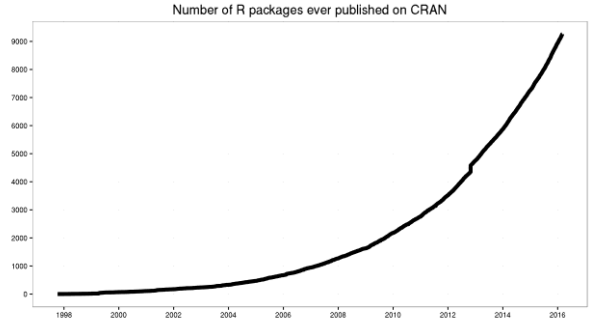
ENDANGERED WILDLIFE TRUST



R packages

ENDANGERED WILDLIFE TRUST

- Base R
- The building blocks of the R user experience
- Consists of functions & objects developed by members of the R community
- Completely free to use
- Download and install in a single line: `install.packages("tidyverse")`
- Load before starting each session: `library(tidyverse)`



R package repositories

ENDANGERED WILDLIFE TRUST

CRAN

https://cran.r-project.org/

GitHub

https://github.com/

R package vignettes

ENDANGERED WILDLIFE TRUST

- An in-depth guide to a package
- Can describe the problem that a package is designed to solve, and then show the reader how to solve it
- A vignette divides functions into useful categories, and demonstrate how to coordinate multiple functions to solve problems

sf package vignette

ENDANGERED WILDLIFE TRUST

1. Simple Features for R

Simple features or *simple feature access* refers to a formal standard (ISO 19125-1:2004) that describes how objects in the real world can be represented in computers, with emphasis on the *spatial* geometry of these objects. It also describes how such objects can be stored in and retrieved from databases, and which geometrical operations should be defined for them.

The standard is widely implemented in spatial databases (such as PostGIS), commercial GIS (e.g., ESRI ArcGIS) and forms the vector data basis for libraries such as GDAL. A subset of simple features forms the GeoJSON standard.

R has well-supported classes for storing spatial data (sp) and interfacing to the above mentioned environment has so far lacked a complete implementation of simple features, making conversions at times convoluted, inef The package sf tries to fill this gap, and aims at succeeding sp in the long term.

This vignette:

- explains what is meant by features, and by simple features
- shows how they are implemented in R
- provides examples of how you can work with them
- shows how they can be read from and written to external files or resources
- discusses how they can be converted to and from sp objects
- shows how they can be used for meaningful spatial analysis

What is a feature?

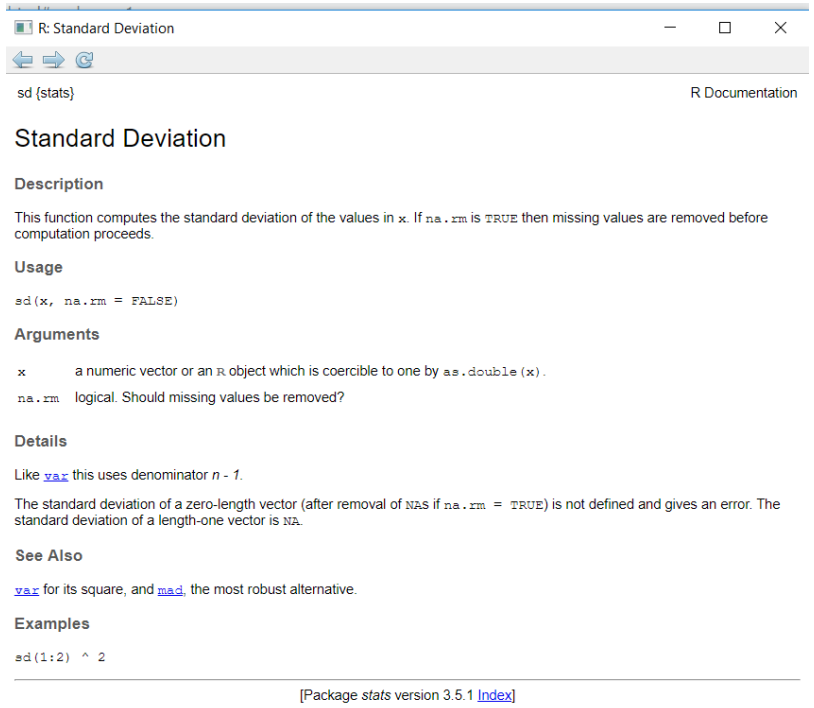
A feature is thought of as a thing, or an object in the real world, such as a building or a tree. As is the case with objects, they often consist of other objects. This is the case with features too: a set of features can form a single feature. A forest stand can be a feature, a forest can be a feature, a city can be a feature. A satellite image pixel can be a feature, a complete image can be a feature too.

Features have a *geometry* describing *where* on Earth the feature is located, and they have attributes, which describe other properties. The geometry of a tree can be the delineation of its crown, of its stem, or the point indicating its centre. Other properties may include its height, color, diameter at breast height at a particular date, and so on.

Getting help in R

ENDANGERED WILDLIFE TRUST

- Help page for each function
- Divided into sections in a standardised format
- Access help: `?sd` or `F1`



R: Standard Deviation

sd {stats} R Documentation

Standard Deviation

Description

This function computes the standard deviation of the values in `x`. If `na.rm` is `TRUE` then missing values are removed before computation proceeds.

Usage

```
sd(x, na.rm = FALSE)
```

Arguments

`x`	a numeric vector or an `R` object which is coercible to one by `as.double(x)`.
`na.rm`	logical. Should missing values be removed?

Details

Like `var` this uses denominator *n - 1*.

The standard deviation of a zero-length vector (after removal of `NA`s if `na.rm = TRUE`) is not defined and gives an error. The standard deviation of a length-one vector is `NA`.

See Also

`var` for its square, and `mad`, the most robust alternative.

Examples

```
sd(1:2) ^ 2
```

[Package *stats* version 3.5.1 Index]

Getting help outside of R

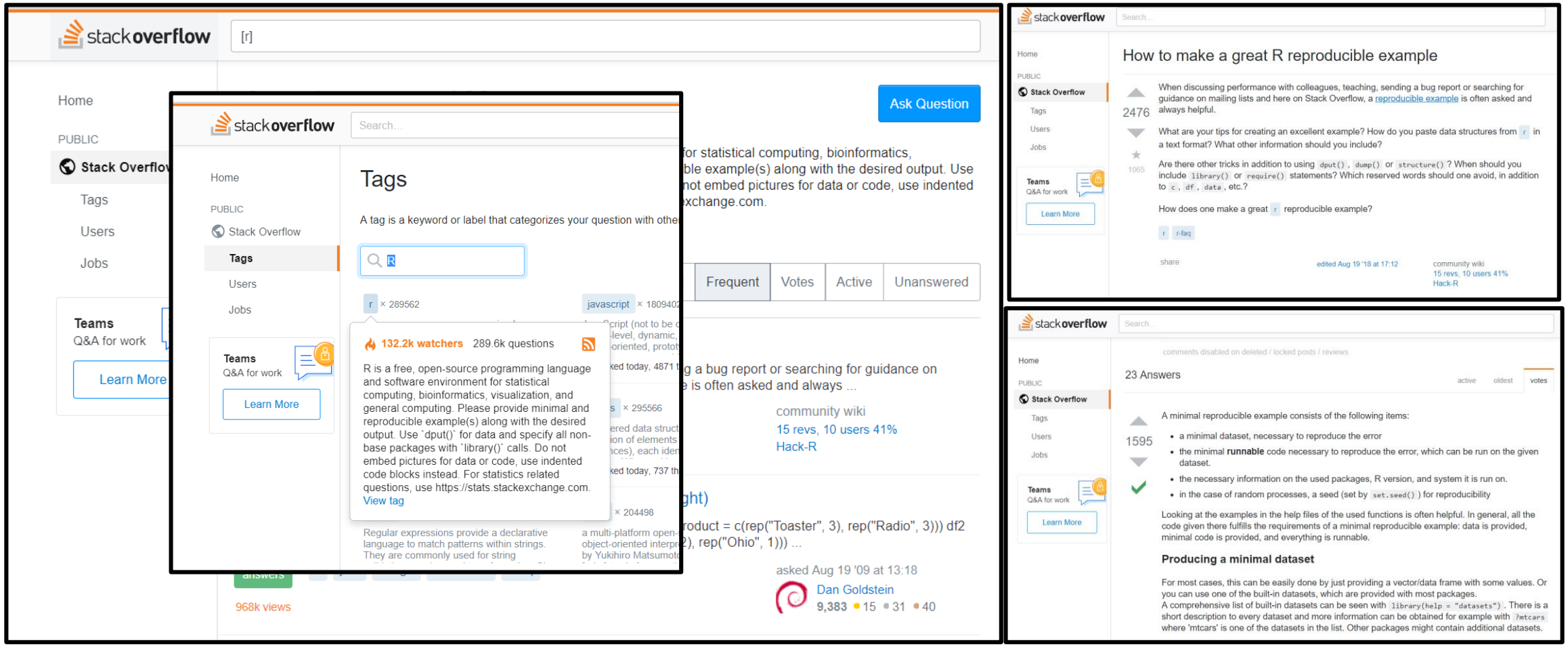
ENDANGERED WILDLIFE TRUST

https://stackoverflow.com/

https://community.rstudio.com/

Getting help outside of R

ENDANGERED WILDLIFE TRUST



R projects

ENDANGERED WILDLIFE TRUST

- Basis for all workflows and analyses
- Best practice – relative directories and file paths
- Makes code and data transferable
- Create skeleton folder structures

Two specific slides generated much discussion and consternation in #rstats Twitter:

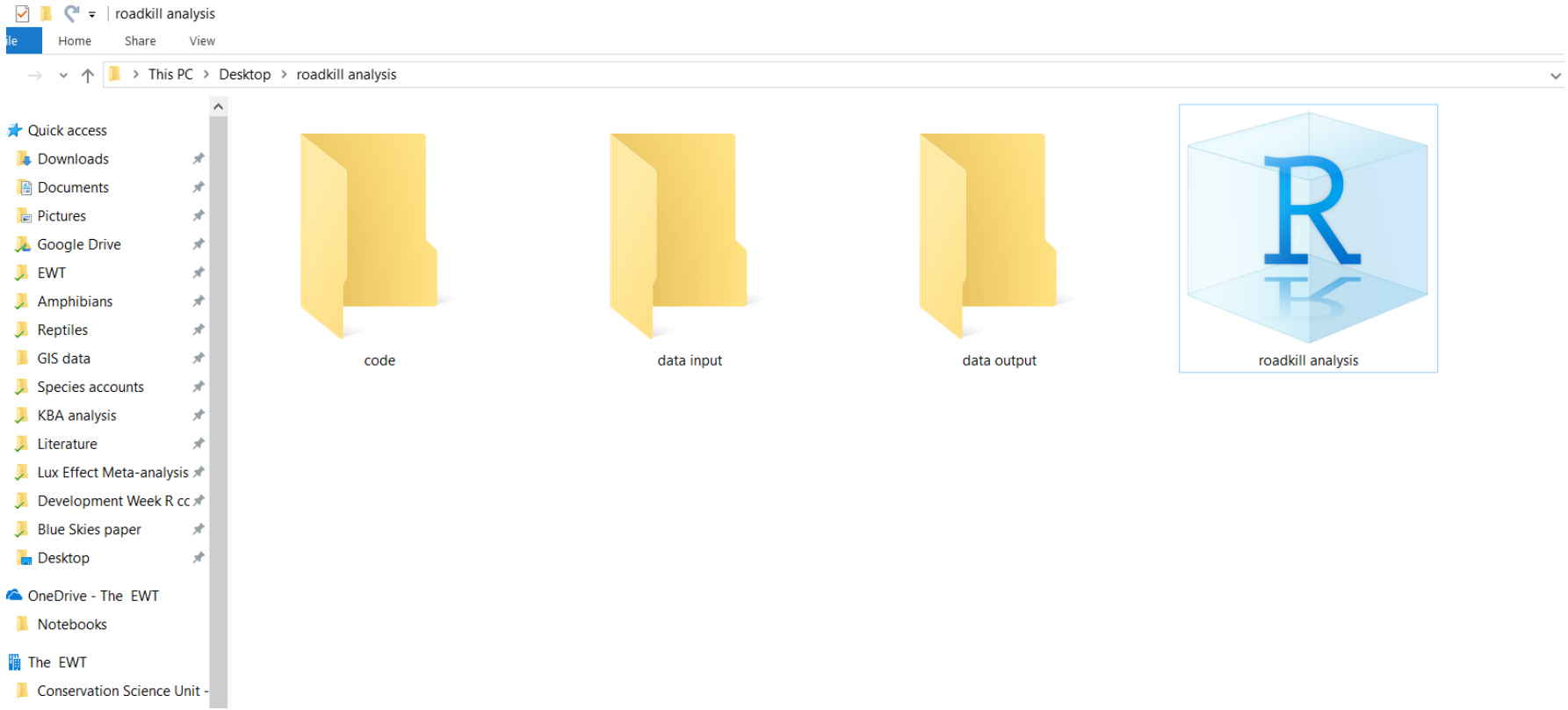
> *If the first line of your R script is*
>
> `setwd("C:\Users\jenny\path\that\only\I\have")`
>
> *I will come into your office and SET YOUR COMPUTER ON FIRE 🔥.*

> *If the first line of your R script is*
>
> `rm(list = ls())`
>
> *I will come into your office and SET YOUR COMPUTER ON FIRE 🔥.*

R projects

ENDANGERED WILDLIFE TRUST

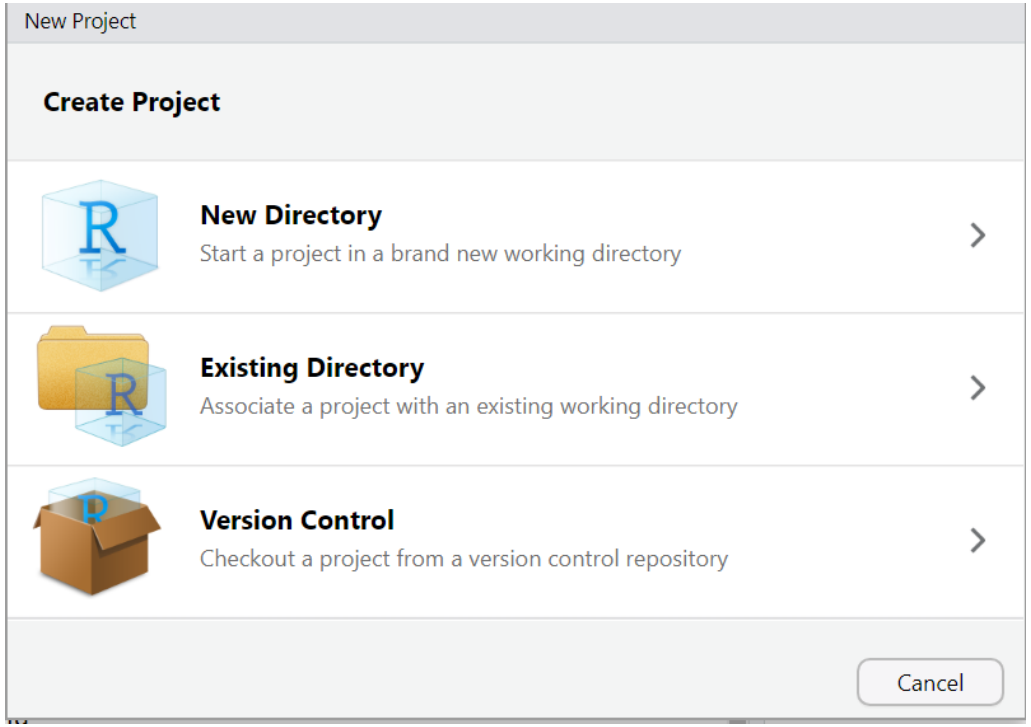
Step 1: Create folder and subfolders



R projects

ENDANGERED WILDLIFE TRUST

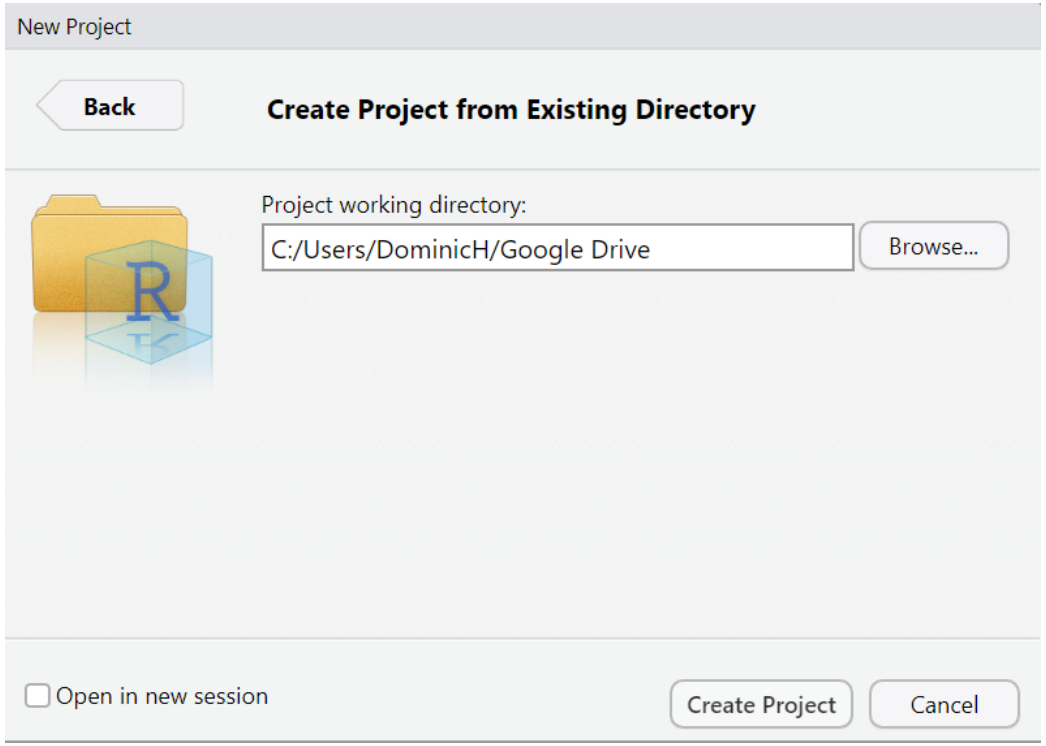
Step 2: Open RStudio, Select *File* -> *New Project*

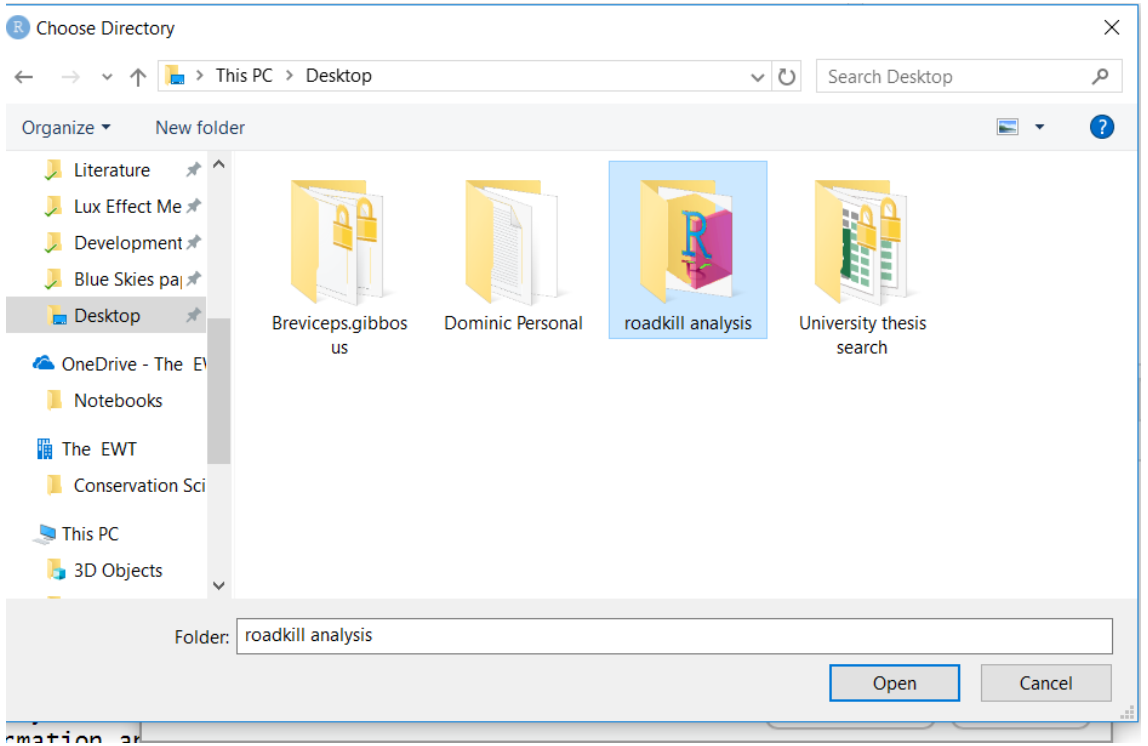


R projects

ENDANGERED WILDLIFE TRUST

Step 3: Select *Existing directory*





RStudio Cloud

ENDANGERED WILDLIFE TRUST

- Demo
- Explanation of assignments
- Interactive introductory tutorials (spare time)

R basics

ENDANGERED WILDLIFE TRUST

- Rmd file link