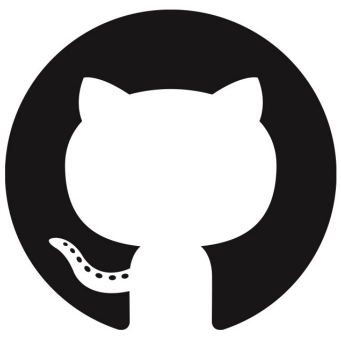
Open online self-study modules to learn R

Claudia Brauer, Paul Torfs, Remko Uijlenhoet

Hydrology and Quantitative Water Management Group,
Wageningen University, The Netherlands



Instructor's problem

"I want to use R in my course,

but I don't want to sacrifice precious practical time

to teach students basic programming skills."

Our famous R intro

A (very) short introduction to R

Paul Torfs & Claudia Brauer
Hydrology and Quantitative Water Management Group,
Wageningen University, The Netherlands

31 January 2018

1 Introduction

R is a powerful language and environment for statistical computing and graphics. It is a public domain (a so called "GNU") project which is similar to the commercial S language and environment which was developed at Bell Laboratories (formerly AT&T, now Lucent Technologies) by John Chambers and colleagues. R can be considered as a different implementation of S, and is widely used as an educational language and research tool.

The main advantages of R are the fact that R is freeware and that there is a lot of help available on-

`www.r-project.org`

and do the following (assuming you work on a windows computer):

1. click `download CRAN` in the left bar
2. choose a download site
3. choose `Windows` as target operation system
4. click `base`
5. choose `Download R 3.4.3 for Windows`[3] and choose default answers for all questions

It is also possible to run R and RStudio from a USB

The basics in only 10 pages

www.github.com/ClaudiaBrauer/A-very-short-introduction-to-R

8 self-study modules

1. A (very) short introduction to R
2. R Programming MOOC first part
3. R Programming MOOC second part
4. Basic plotting
5. Pretty plotting
6. Reading data files
7. Matrix operations
8. Spatial data

1–3 hours each (depending on background)

Self-study modules downloadable from GitHub

GitHub, Inc. [US] | https://github.com/ClaudiaBrauer/A-very-short-introduction-to-R

A (very) short introduction to R

Here you'll find three documents that my colleague Paul Torfs and I wrote about learning R:

1. A (very) short introduction to R

The base document, with 10 pages of background and exercises and 2 pages listing useful functions (to use as a reference). Working through this document takes 1 to 2 hours (depending on your background). An older version of this document can also be downloaded from the R website (as contributed document): https://cran.r-project.org/doc/contrib/Torfs+Brauer-Short-R-Intro.pdf

2. Doing "A (very) short introduction to R" in the interactive swirl environment

Instead of reading the pdf and doing the ToDo exercises, you can also go through the text and exercises in an interactive environment called swirl (developed by swirlstats.com). This short manual gets you started with the (very) short introduction to R. It also points you to some nice follow-up classes created by others.

3. Writing your own R scripts

After learning the basics, you have to gain experience in building R scripts. In this document you learn to set up a script step by step. The examples are from hydrology, but the exercises are useful for everyone.

4. Portable versions of R and RStudio

In case you want to take R everywhere you go (you may want to install the programs on a USB stick in case of administrator rights issues).

Module 1: the basics

A (very) short introduction to R

Paul Torfs & Claudia Brauer
Hydrology and Quantitative Water Management Group,
Wageningen University, The Netherlands

31 January 2018

1 Introduction

R is a powerful language and environment for statistical computing and graphics. It is a public domain (a so called "GNU") project which is similar to the commercial S language and environment which was developed at Bell Laboratories (formerly AT&T, now Lucent Technologies) by John Chambers and colleagues. R can be considered as a different implementation of S, and is widely used as an educational language and research tool.

The main advantages of R are the fact that R is freeware and that there is a lot of help available on-

`www.r-project.org`

and do the following (assuming you work on a windows computer):

1. click `download CRAN` in the left bar
2. choose a download site
3. choose `Windows` as target operation system
4. click `base`
5. choose `Download R 3.4.3 for Windows`[3] and choose default answers for all questions

It is also possible to run R and RStudio from a USB

Designed for people who never progammed before

Module 1 – swirl version

Doing

A (very) short introduction to R

in the interactive swirl environment

2.3 Starting swirl

On the swirl website `www.swirlstats.com`, on the tab "Learn", you'll find the steps you'll have to take to get swirl. You have already installed R and RStudio, so you can skip the first steps and:

1. Open RStudio.
2. Install the `swirl` package by typing `install.packages("swirl")`
3. Load the package by typing `library(swirl)`
4. Start swirl by typing `swirl()`

2.4 Installing the course

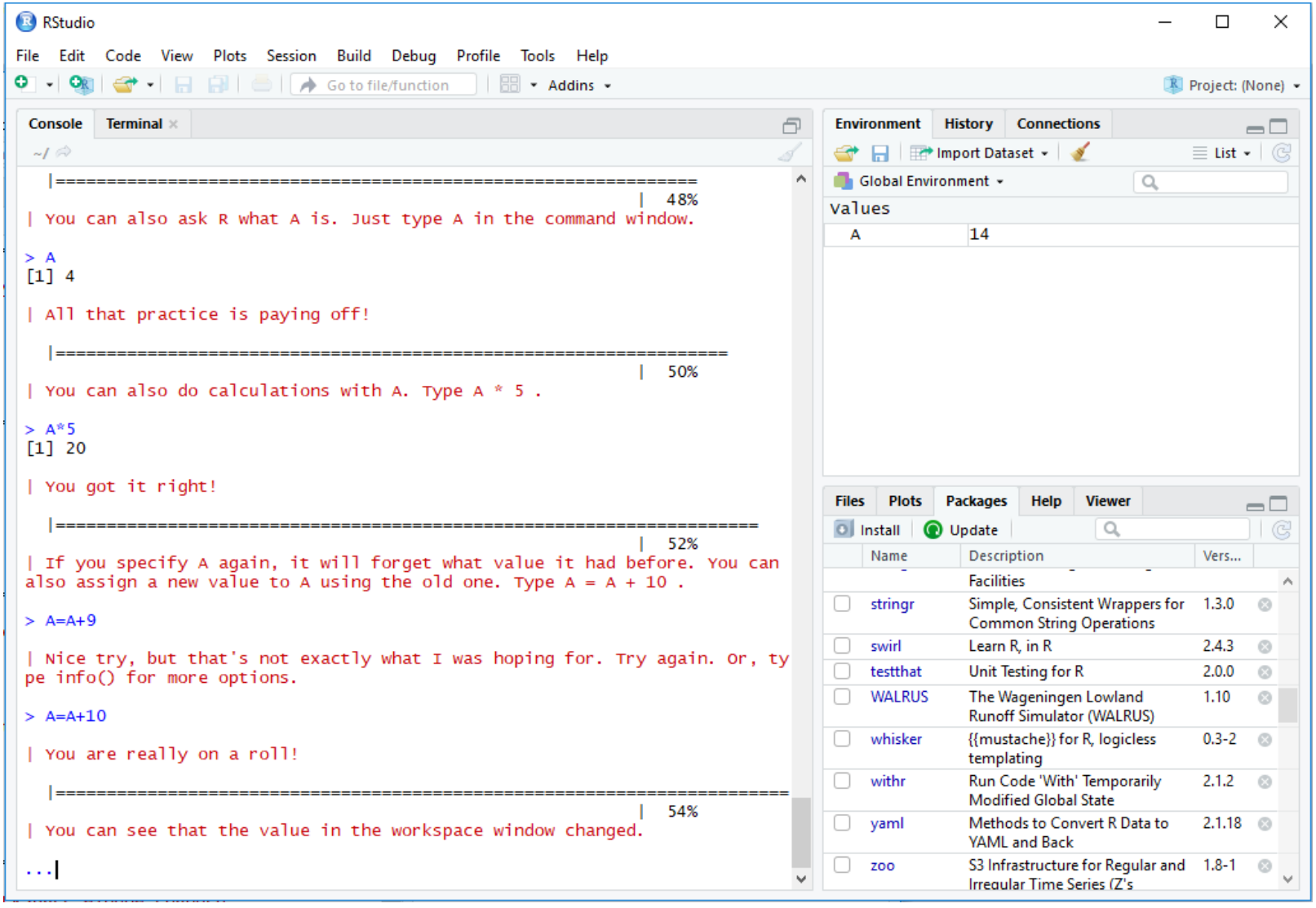
After entering `swirl()`, the program will ask you (interactively) which course you want. If you want a new course, you have to install it first from internet . Type:

```
install_course("A-(very)-short-
introduction-to-R")
```

"A (very) short introduction" is divided into 3 swirl modules, corresponding to Sections 2–3, 5–7 and 8–11 in the pdf version

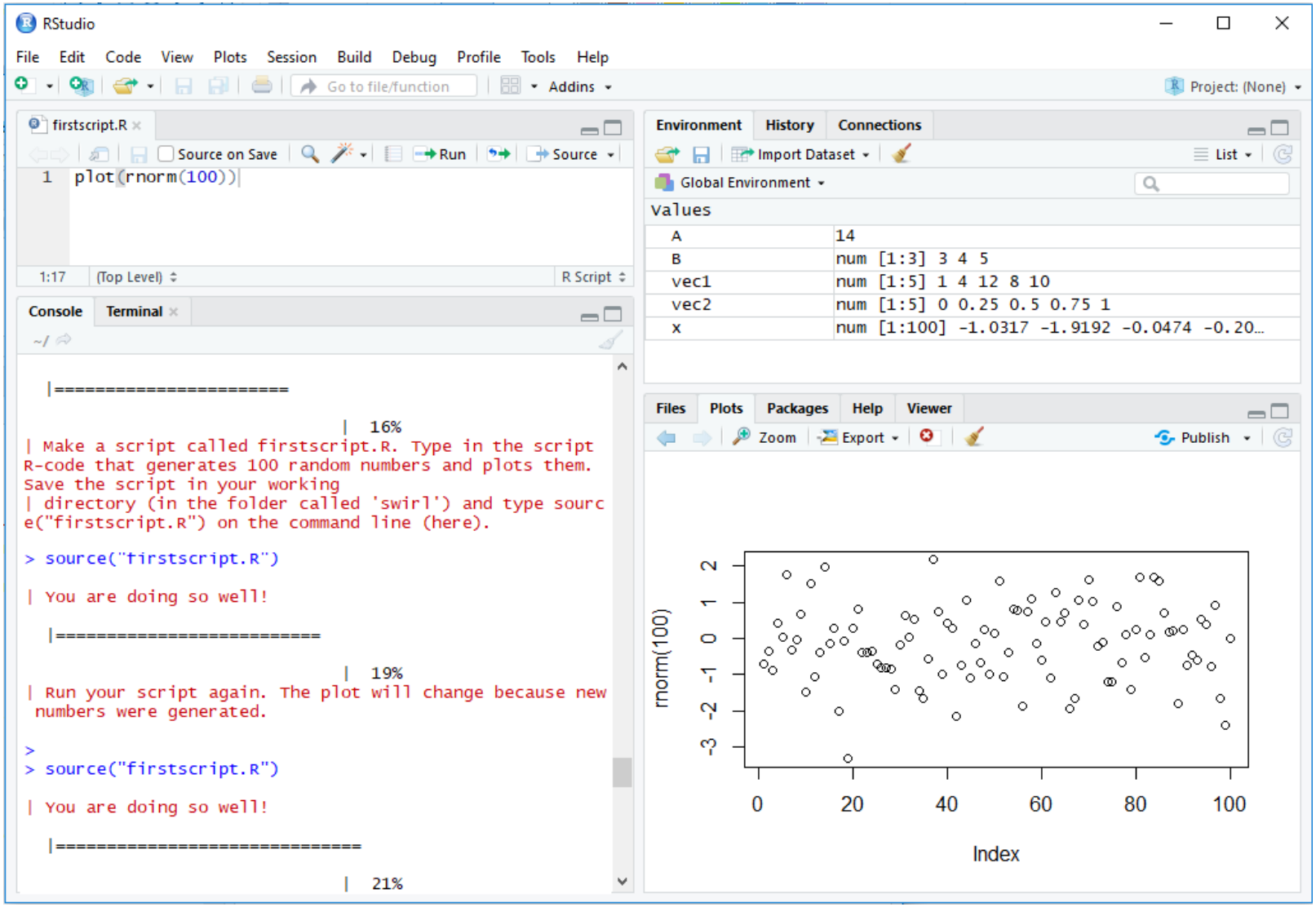
Interactive format: "Learn R, in R", `www.swirlstats.com`

Swirl examples



Red = instructions, blue = response, black = output.

Swirl examples



Also script building assignments.

Modules 2+3: from R programming MOOC

Swirl exercises made by Johns Hopkins University
for the Coursera MOOC "R programming".

`www.swirlstats.com`

Module 2

1. Basic Building Blocks
2. Workspace and Files
3. Sequences of Numbers
4. Vectors
5. Missing Values
6. Subsetting Vectors
7. Matrices and Data Frames
8. Logic

Module 3

9. Functions
10. lapply and sapply
11. vapply and tapply
12. Looking at Data
13. Simulation
14. Dates and Times
15. Base Graphics

Modules 4–8: writing scripts

Writing your own R scripts

A follow-up to "A (very) short introduction to R"

Introduction

With the exercises in this document you learn how to set up an R script from scratch. The exercises explain step by step how to build your script. The scripts in these exercises always follow the same structure with the following Sections (see Fig. 1):

Initializing Clear memory, set working directory and load packages.

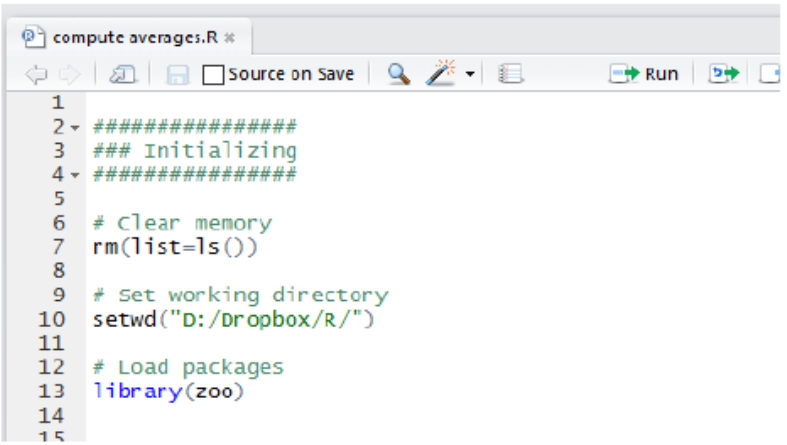
Data Read data from file and data preprocessing (such as extracting and renaming one column of the total data set).

Processing Do computations, run a model, etc.

Output Make a dataframe and/or figure and save these to file.

(different languages). You can also Google your problem (type "R" + keyword / problem) and get solutions (functions, packages, code).

- Save the scripts you made as inspiration for your future (thesis) work.



Examples from hydrology

Example of script-writing module

Basic plotting

Compare rainfall measurements

The aim of this exercise is to reproduce Figure 2. You will build it in steps. Don't forget to run (part of) the code every time you add something, to check if everything is still error-free.

1. Download the file `P_gauge_radar.dat` and save it on your computer. Make sure that the folders for scripts, input (data files) and output (figures and data files) are logically structured.
2. Start with an empty script. Make the main headers `Initializing`, `Data`, `Processing` and `Output`). Don't forget the `#`-sign to indicate that they are comments and not commands (see Sec. 5 in "A (very) short introduction to R").
3. Under `Initializing`, add commands to clear R's memory and set the working directory (see Fig. 1).
4. Add code to read the data file (Section `Data`).

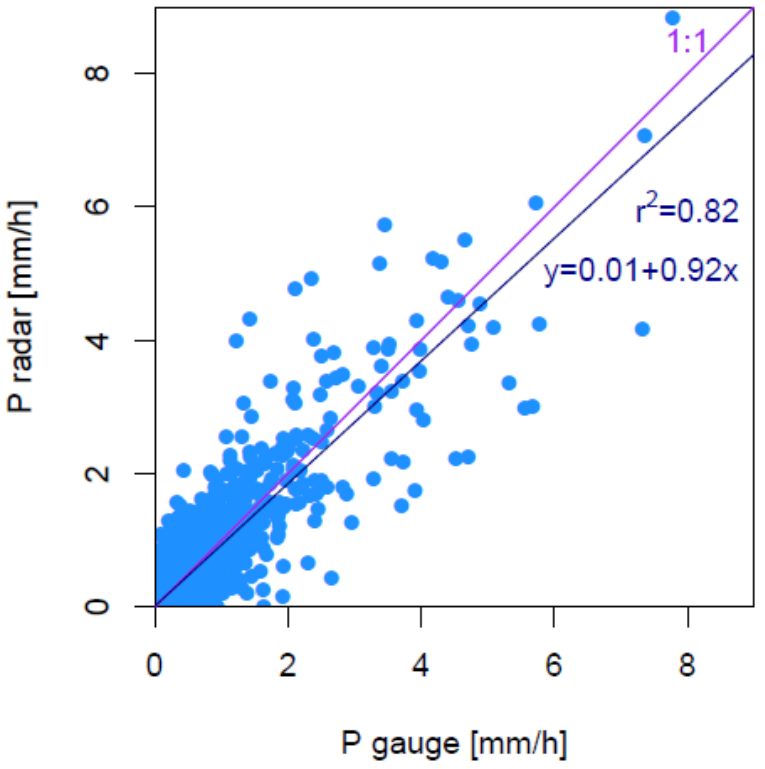


Figure 2: Comparing rainfall measured with a radar to rainfall measured with a rain gauge. Figure source: C.C. Brauer, A. Overeem, H. Leijnse, R. Uijlenhoet (2016): The effect of differences between rainfall measurement techniques on groundwater and discharge simulations in a lowland catchment, Hydrol. Process., 30, 3885–3900

Difficulty increases from module 4 to 8.

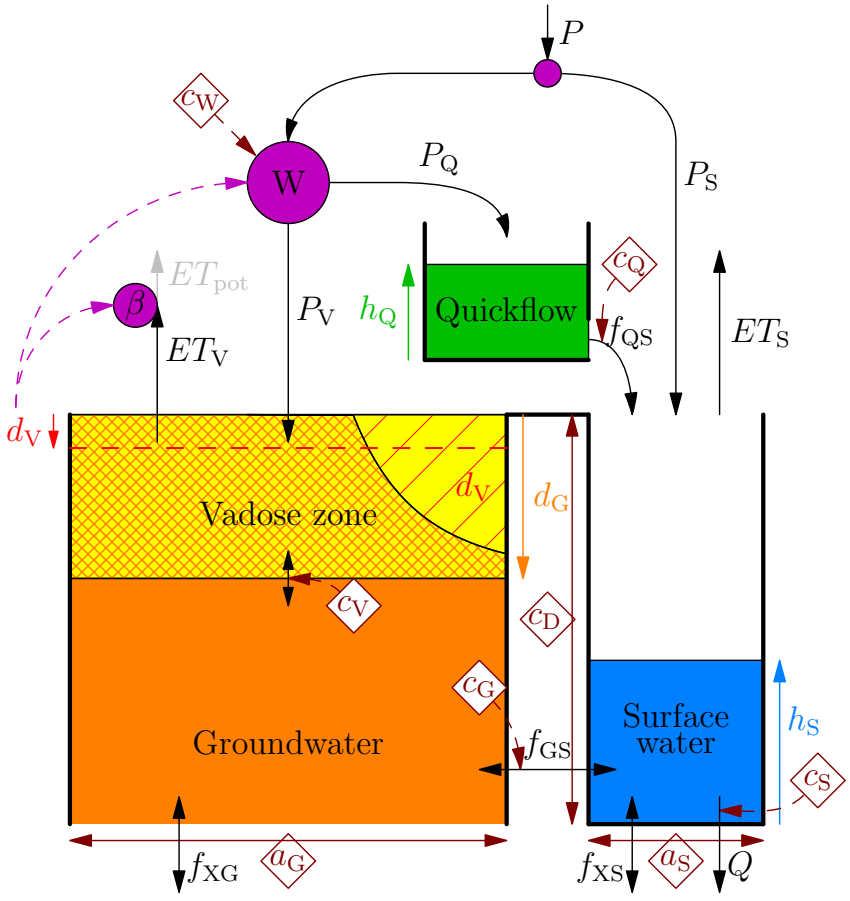
Extra: tutorial for rainfall-runoff model

WALRUS:

- Wageningen Lowland Runoff Simulator
- Simple, parametric model for catchments with shallow groundwater

Tutorial:

- Learn to understand and work with the R package
- For students, water managers, consultants and researchers
- R-package, user manual, publications and more on `www.github.com/ClaudiaBrauer/WALRUS`



Contents WALRUS tutorial

- Run WALRUS
  - The user interface (Shiny)
  - The regular R environment (RStudio)
    - The base script
    - Analyse results
    - Change source code
- Case studies
  - Flood - Berkel
  - Water level management - Polder Cabauw
  - Inlet - Bakelse Aa
  - Upward seepage - Polder Oude Riet
  - Snow - Reusel
  - Stage-discharge relation - Hupsel Brook catchment
  - Other catchments
- Calibration
  - Manual
  - Automatic
  - Random

WALRUS summary

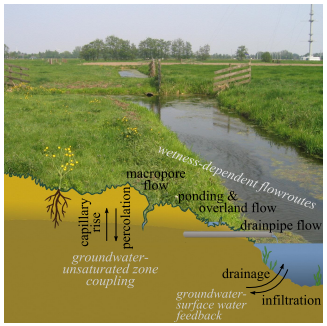
WALRUS

WALRUS (Wageningen Lowland Runoff Simulator) is a rainfall-runoff model developed to fill the gap between complex, spatially distributed models which are often used in lowland catchments and simple, parametric models which have mostly been developed for sloping catchments.

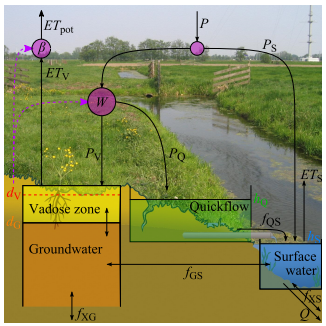
Key features

WALRUS explicitly accounts for:

- groundwater-unsaturated zone coupling
- wetness-dependent flow routes
- groundwater-surface water feedbacks
- seepage and surface water supply

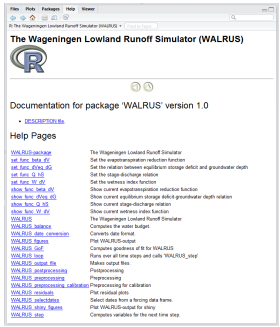


Model structure

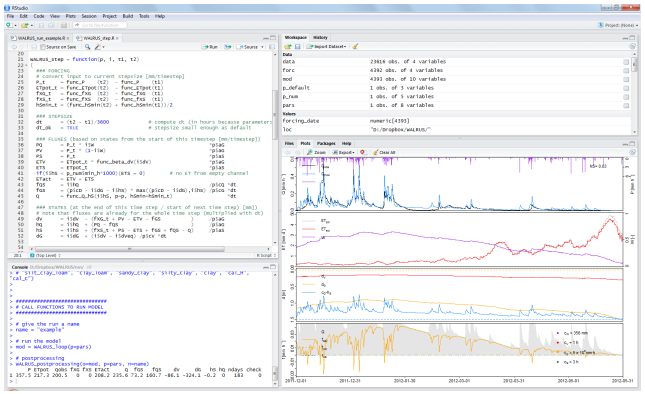


The simple model structure and the clear (qualitative) relation between model states and measurable variables makes WALRUS suitable for teaching. You can use WALRUS to explain hydrological processes or use it as an example of a conceptual rainfall-runoff model to explain .

Using WALRUS



WALRUS is programmed in R (open source and freeware programming environment) and wrapped in a package. We also wrote a (very) short introduction to R, to learn the basics of R in 1 hour: **github.com/ClaudiaBrauer/A-very-short-introduction-to-R**



With a tutorial, user manual and examples, students and water managers can start modelling quickly.

User days for water managers and consultants

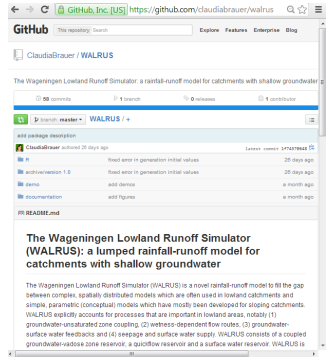
In February 2015 and March 2016 we organised WALRUS user days where 30–40 Dutch water managers, consultants and researchers learned to work with WALRUS.

Download

WALRUS is freeware and open source and can be downloaded from **github.com/ClaudiaBrauer/WALRUS**.

Downloads include:

- R-package
- User manual
- Tutorial
- Publications
- Old versions
- User contributions
- ...



Summary

Advantages

- For freely draining catchments and polders with controlled water levels
- Fast
- Few parameters
- Clear (qualitative) relation between model states and measurable variables
- Standard options for initial conditions
- Freeware and open source

Applications

- Operational forecasting
- Real-time control
- Input for hydraulic model
- Risk analyses
- Scenario analyses
- Design of infrastructure
- Gap-filling
- ...

Outlook

We are using WALRUS to generate examples for simple interactive tools to explain hydrological principles effectively. We are also working on an interactive tool with which students and water managers can see the effect of (change in) catchment characteristics on hydrological processes.

Summary

We developed:

- 8 modules of 1-3 hours:
  - 1 A (very) short introduction to R
  - 2–3 Swirl exercises from R programming MOOC (Johns Hopkins U.)
  - 4–8 Writing scripts from scratch
- Tutorial for rainfall-runoff model WALRUS
- Everything freely available from `www.github.com/ClaudiaBrauer`