Software installation instructions

'Hands-on Digital Soil Mapping' course

The 'Hands-on Digital Soil Mapping' course introduces methods and software for management, analysis and mapping of soil type and soil properties within the R environment for statistical computing. The course alternates between lectures and computer exercises and covers a variety of subjects, such as geostatistics, machine learning for soil mapping, soil functional mapping, proximal soil sensing, quantification of uncertainty, sampling for mapping and statistical validation.

Exercises will be done with the **R** software for statistical computing.

To spend the time during the training as efficiently as possible you should come prepared and should have installed the required software before coming to the training. This document provides software installation instructions for R, RStudio and GDAL (required to work with spatial data in R). Some sample code is provided in this document so that you can check if everything has been installed correctly.

Installation instructions

Required software:

- **R** [https://cran.r-project.org/bin/windows/base/old/3.6.3/], click 'Download R 3.6.3' and follow the installation instructions.

  NOTE 1: the minimum R version required is **3.6.3**. If you are running an older version then update to 3.6.3.

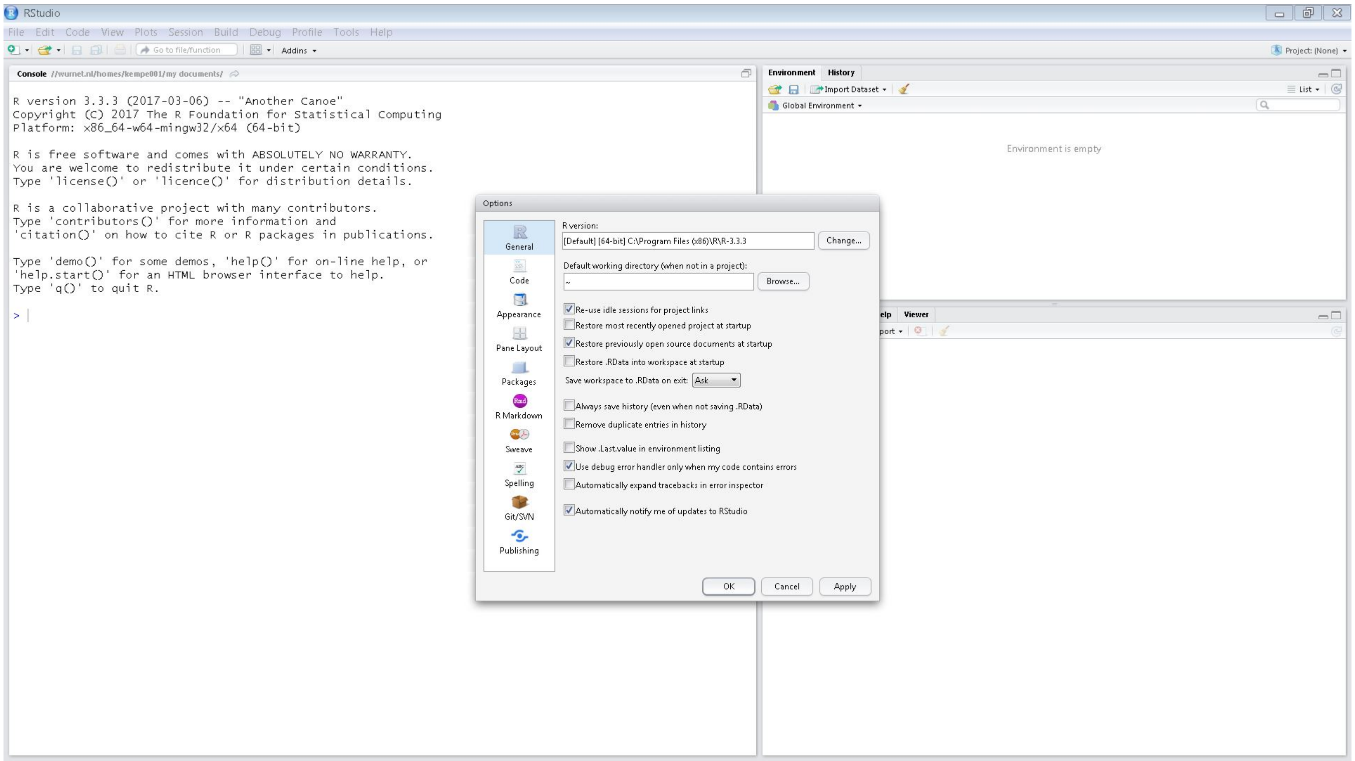
  NOTE 2: the latest R version is 4.1.0. However, we cannot guarantee that all code will run without issues for this R version since the materials were developed with version 3.6.3. To be on the safe side, we recommend installing 3.6.3.

- **RStudio** [https://rstudio.com/products/rstudio/download/#download], install RStudio desktop version 1.3.1093;

- **GDAL**. The most convenient way to install GDAL is via de OSGeo4W installer [https://trac.osgeo.org/osgeo4w/]. Download the 32bit or 64 bit installer, depending on your operating system, and run it. Select *Express Desktop Installer* and then select packages 'QGIS' and 'GDAL'.

Please make sure that you install the software to the typical software installation directory (for Windows this is typically: **C:\Programe Files**) and preferably not in your user profile.

RStudio is the main R scripting environment. Once you start RStudio, it should automatically find and connect to R. If this is not the case you can set the R version by clicking 'Tools' on the menu bar and then selecting 'Global Options'. The path to the R software can be set under 'General' (first box).



```
R version 3.3.3 (2017-03-06) -- "Another Canoe"
Copyright (C) 2017 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> 
```

Options

R version: [Default] [64-bit] C:\Program Files (x86)\R\R-3.3.3

Default working directory (when not in a project): ~

☑ Re-use idle sessions for project links
☐ Restore most recently opened project at startup
☑ Restore previously open source documents at startup
☐ Restore .RData into workspace at startup
Save workspace to .RData on exit: Ask

☐ Always save history (even when not saving .RData)
☐ Remove duplicate entries in history

☐ Show .Last.value in environment listing
☑ Use debug error handler only when my code contains errors
☐ Automatically expand tracebacks in error inspector

☑ Automatically notify me of updates to RStudio

R comes with basic functionality for statistical analyses. For more advanced analyses, including spatial analyses 'packages' need to be installed. For the training the following packages are used and should be installed prior to the training:

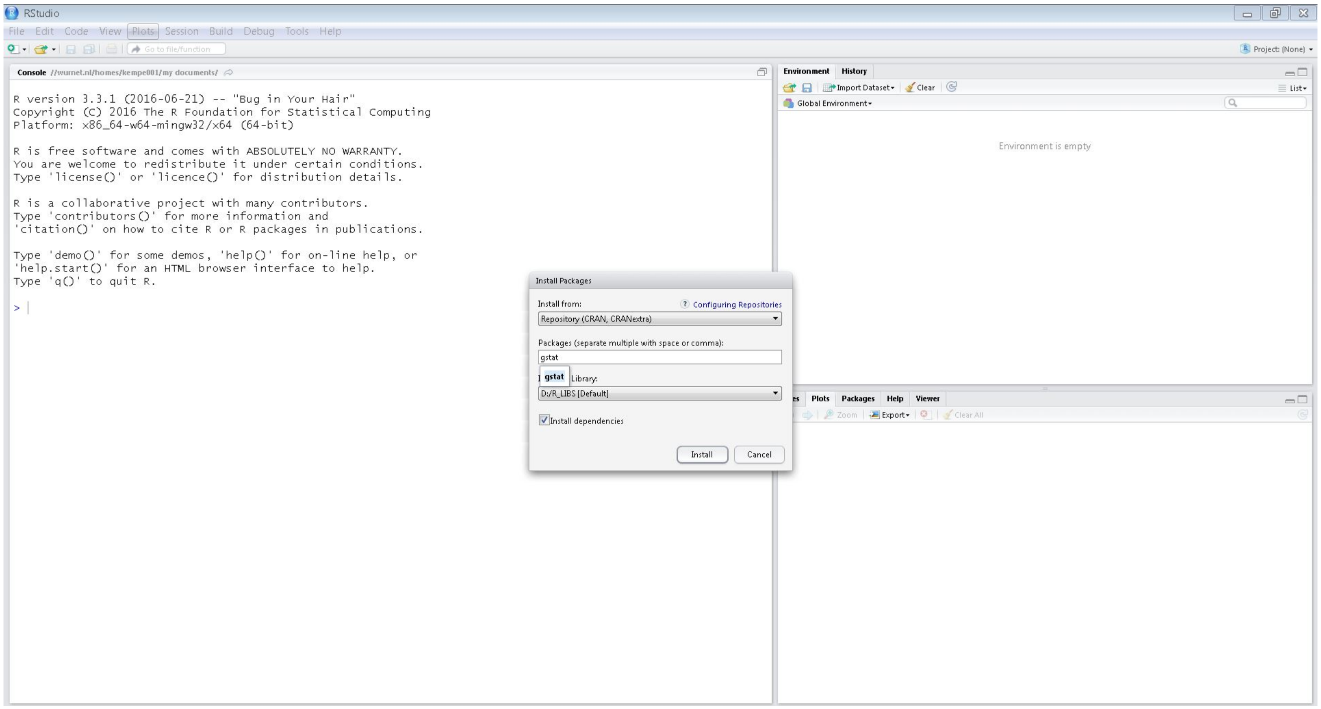
- **R packages**:

    - sf
    - terra
    - raster
    - rgdal
    - ranger
    - caret
    - ggplot2
    - leaflet
    - plotKML
    - htmlwidgets
    - skimr
    - rasterVis
    - plyr
    - sp
    - rgdal
    - maptools
    - gstat
    - rgeos
    - MASS

There are several ways to install packages (see also: how to install R package). Perhapos the easies is to open RStudio and **copy the following lines to the RStudio console** and then **press 'enter'** (copy the three lines in one go).

```
list.of.packages <- c("sf","terra","raster","rgdal","ranger","caret","ggplot2","leaflet",
"plotKML","htmlwidgets","skimr","rasterVis","plyr","sp","rgdal","maptools","gstat","rgeos
","MASS")

new.packages <- list.of.packages[!(list.of.packages %in% installed.packages()[,"Package"]
)]

if(length(new.packages)) {install.packages(new.packages)}
```

Alternatively, you can use the RStudio menu bar. Click ‘Tools’ and then ‘Install Packages’. A window opens with which you can search for package and subsequently install these.



```
R version 3.3.1 (2016-06-21) -- "Bug in Your Hair"
Copyright (C) 2016 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

>
```

Install Packages

Install from: Repository (CRAN, CRANextra)

Configuring Repositories

Packages (separate multiple with space or comma): gstat

Install to Library: D:/R_LIBS [Default]

☑ Install dependencies

When you install a package you will notice that sometimes other packages are automatically installed with it. These packages are ‘dependencies’ that the package you are installing needs to properly function. Make sure the ‘Install dependencies’ box is ticked.

It is advised to **test if everything is properly installed** and working. Note the **code** used for testing is sent with these instructions. After downloading **open the script** in RStudio. The code should run without any errors.

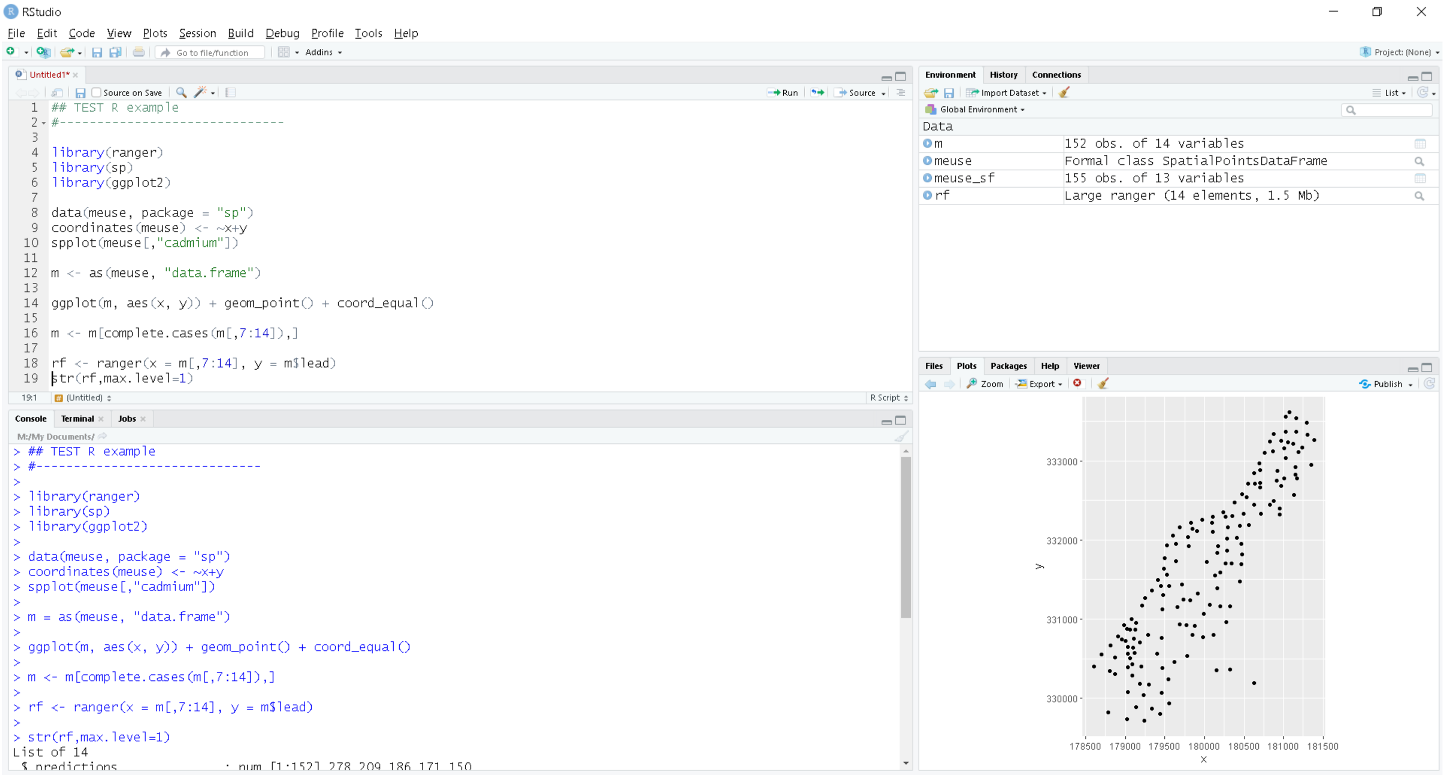
Testing R

To test if the installed packages work properly explore the ‘mtcars’ object by running the following lines of code (just run, ignore possible warnings) from the associated script. You can run by selecting the code with the mouse (the code will highlight; make sure you select the entire code otherwise you get errors) and then pressing the ‘Run’ button in the top-right corner of the console (or press short-cut ‘ctrl-enter’). You can also run line-by-line by placing the cursor on the line you want to run and then clicking the ‘Run’ button (or ‘ctrl-enter’). (Warning messages can be ignored).

Try running:

```
## TEST R example
#------------------------------

library(ranger)
library(sp)
library(ggplot2)

data(meuse, package = "sp")
coordinates(meuse) <- ~x+y
spplot(meuse[,"cadmium"])

m = as(meuse, "data.frame")

ggplot(m, aes(x, y)) + geom_point() + coord_equal()

m <- m[complete.cases(m[,7:14]),]

rf <- ranger(x = m[,7:14], y = m$lead)

str(rf,max.level=1)
```

This should give the following output:



```
## TEST R example
#-----------------------------

library(ranger)
library(sp)
library(ggplot2)

data(meuse, package = "sp")
coordinates(meuse) <- ~x+y
spplot(meuse[,"cadmium"])

m <- as(meuse, "data.frame")

ggplot(m, aes(x, y)) + geom_point() + coord_equal()

m <- m[complete.cases(m[,7:14]),]

rf <- ranger(x = m[,7:14], y = m$lead)
str(rf,max.level=1)
```

Data

m	152 obs. of 14 variables
meuse	Formal class SpatialPointsDataFrame
meuse_sf	155 obs. of 13 variables
rf	Large ranger (14 elements, 1.5 Mb)

```
> ## TEST R example
> #-----------------------------
>
> library(ranger)
> library(sp)
> library(ggplot2)
>
> data(meuse, package = "sp")
> coordinates(meuse) <- ~x+y
> spplot(meuse[,"cadmium"])
>
> m = as(meuse, "data.frame")
>
> ggplot(m, aes(x, y)) + geom_point() + coord_equal()
>
> m <- m[complete.cases(m[,7:14]),]
>
> rf <- ranger(x = m[,7:14], y = m$lead)
>
> str(rf,max.level=1)
List of 14
 $ predictions               : num [1:152] 278 209 186 171 150
```

Resources

If you are new to R, we strongly recommend to do some ‘self-study’ before coming to the training. If you come well prepared you will get most out of the course. There is some excellent introductory material available online. We recommend to study some of these and try to practice by running pieces of code and trying to understand what happens:

- RStudio online learning: https://education.rstudio.com/learn/beginner/

- R introductions on YouTube, e.g.: https://www.youtube.com/watch?v=7cGwYMhPDUY

Other good resources to R and R introductory tutorials are web-pages maintained by Dylan Beaudette of UC Davis and David Rossiter of Cornell University:

- Rossiter: “An introduction to (geo)statistics with R”

This is only a very small selection of what is available on R on the internet. It might be worthwhile to search yourself as well.