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 variables. The course alternates between lectures and computer exercises and covers a variety of subjects, such as geostatistics, data preparation, machine learning for soil mapping, quantification of uncertainty and soil map validation.

For the computer exercises, the software R for statistical computing will be used. To spend the time during the training as efficiently as possible you should come prepared and should try install the required software and test the installation before coming the training. Some sample code is provided in this document so that you can check if everything has been installed correctly.

Installation instructions

Required software:

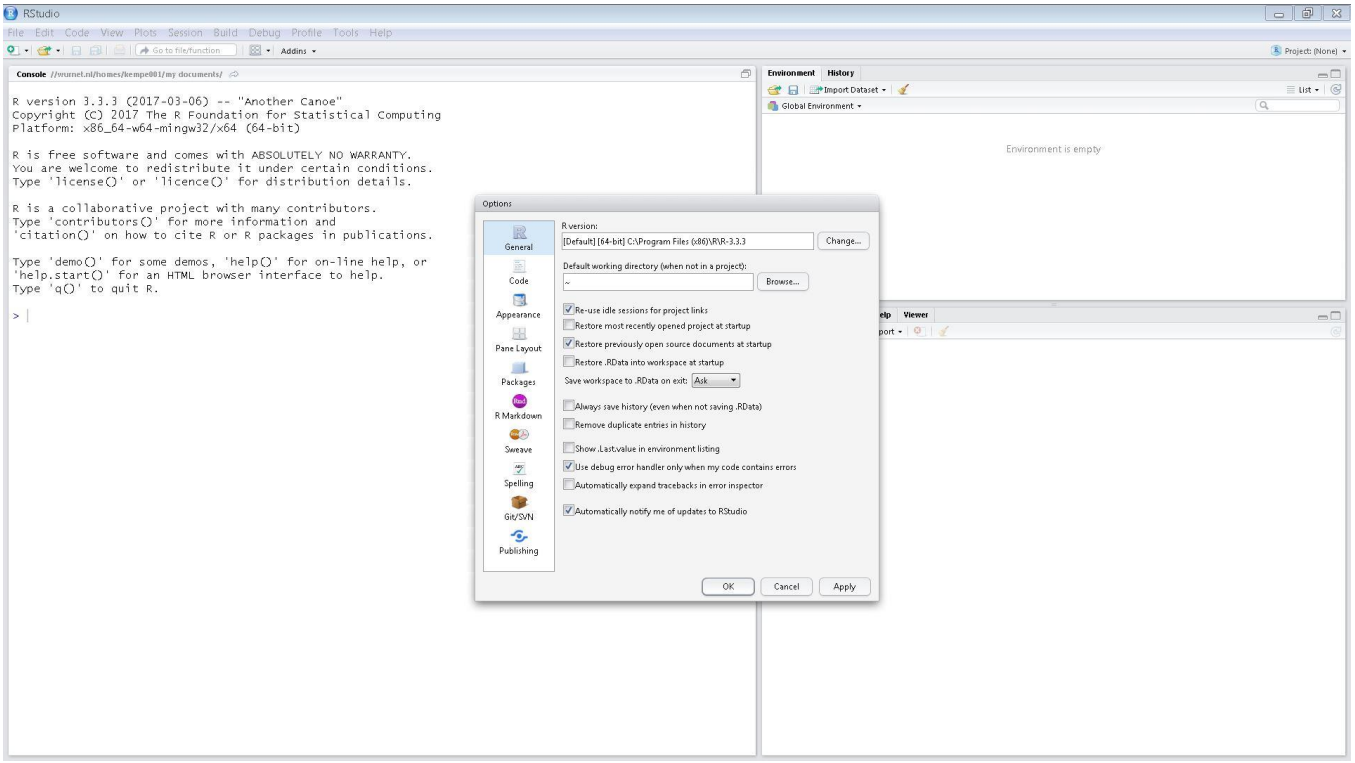
- **R** [http://r-project.org], click 'download R' and select a CRAN (Comprehensive R Archive Network) mirror, e.g. the ones located in Germany at the University of Munster, select your operating system, select 'base', install version 3.4.x.
- **RStudio** [https://www.rstudio.com/products/rstudio/download/], install RStudio desktop version 1.x;

Recommended (free) software, but not mandatory:

- **Google Earth** [http://www.google.com/earth/download/ge].
- **GDAL v. 2.x** [Windows-32bit / Windows-64bit]. GDAL can be used within R for processing of GIS layers.
- **SAGA GIS [**https://sourceforge.net/projects/saga-gis/files/]
- **QGIS v. 3.x** [http://www.qgis.org/en/site/forusers/download].

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

RStudio is in principle 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 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**: sp, rgdal, gstat, rgeos, GSIF, plotKML, caret, plyr, raster, randomForest, ggplot2, e1071, leaflet, htmlwidgets, MASS, dplyr (see also: how to install R package).

There are several ways how to install a package. First you have to open RStudio.

To install these R packages, you can **copy the following lines to the RStudio console** and then **press 'enter'** (copy the three lines in one go).

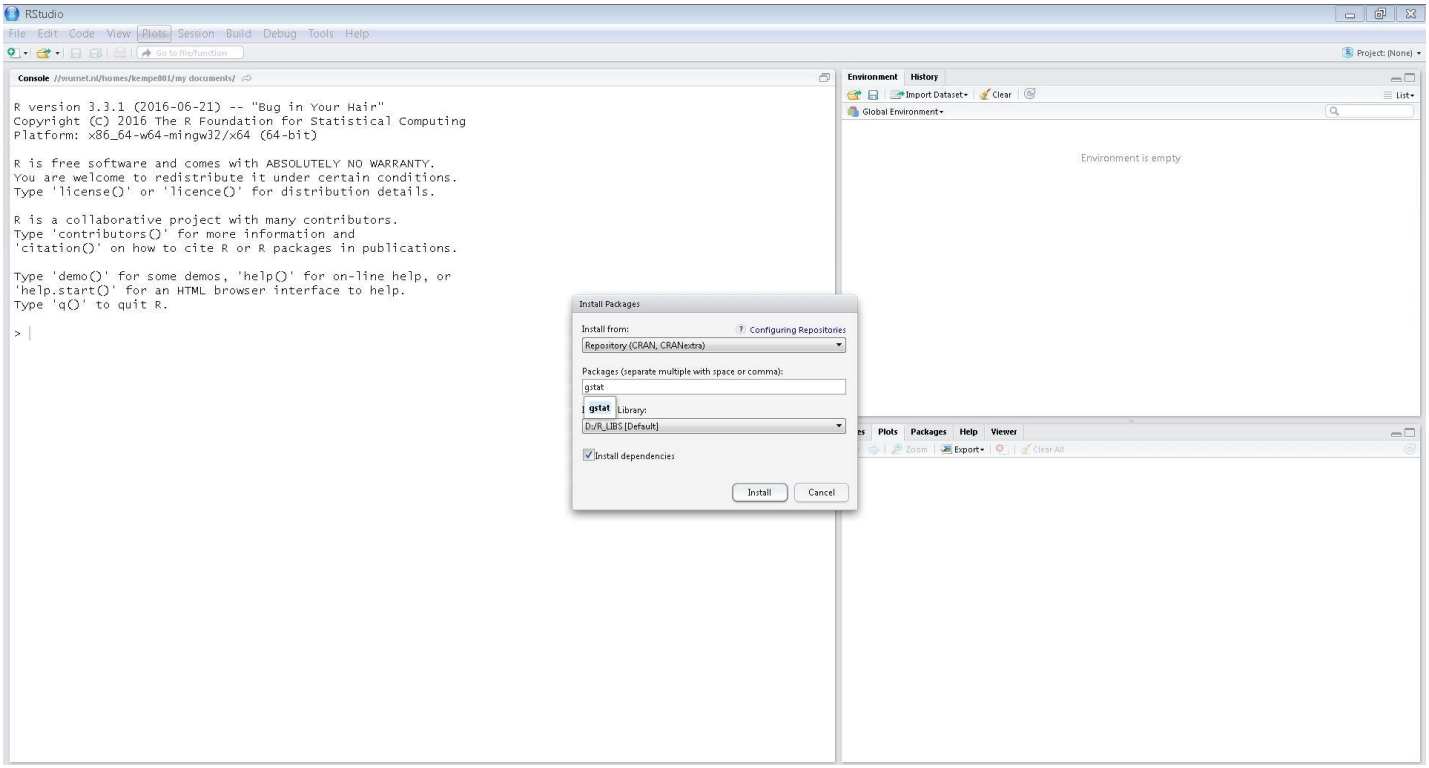
```
list.of.packages <- c("sp", "rgdal", "gstat", "rgeos", "GSIF", "plotKML", "caret", "plyr"
, "raster", "randomForest", "ggplot2", "e1071", "leaflet", "htmlwidgets", "MASS ", "dplyr
")

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.



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.

Now try to **test to check if everything is properly installed** and working. Note the **code** used for testing can be downloaded from **here**. After downloading **open the script** in RStudio. The code should run without any errors.

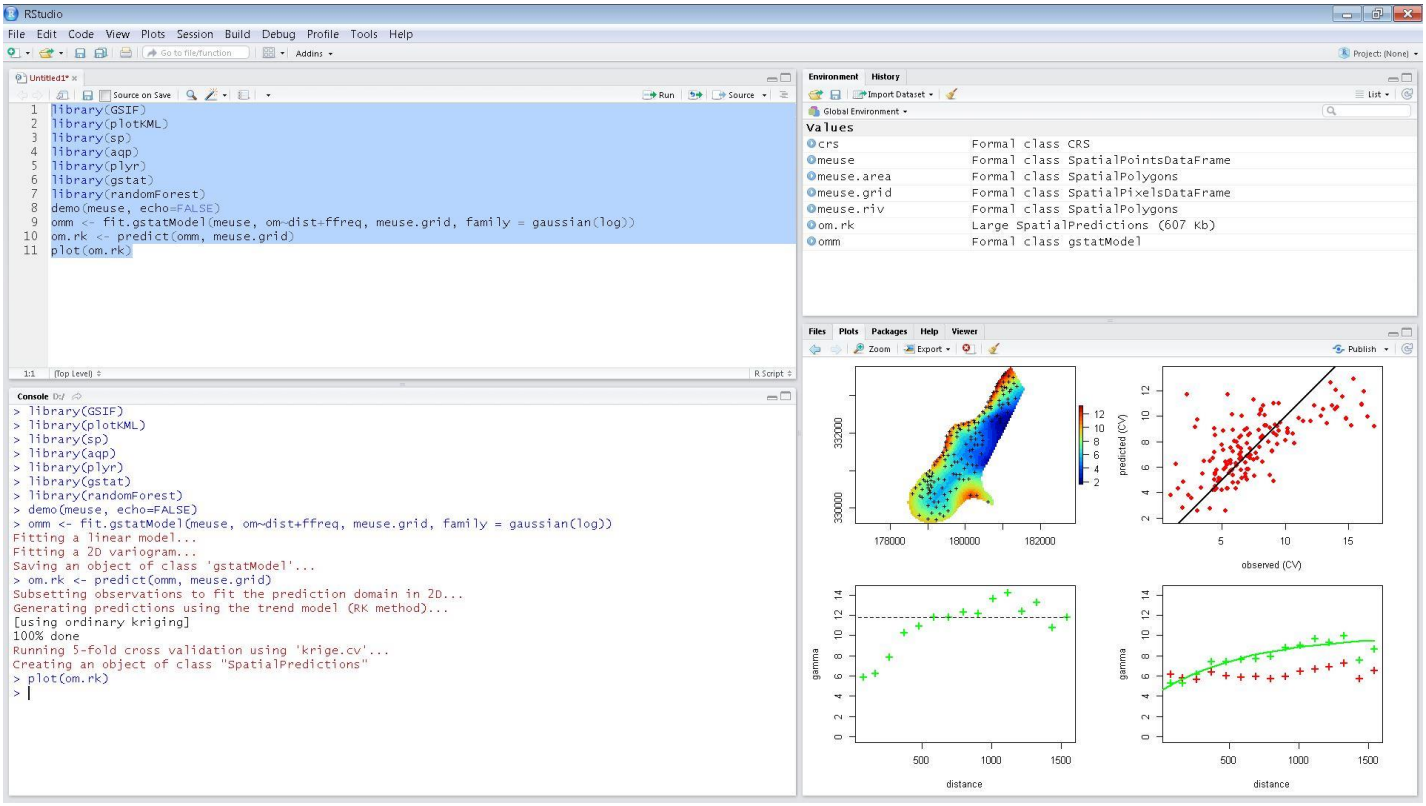
Testing: R

To test if the installed packages work properly create soil maps by running the following lines of code (just run, ignore possible warnings) from the downloaded script. You can run by selecting the code with the mouse (the code will highlight; make sure you select the entire code otherwise you will 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').

```
library(GSIF)
library(plotKML)
library(sp)
```

```
library(plyr)
library(gstat)
library(randomForest)
demo(meuse, echo=FALSE)
omm <- fit.gstatModel(meuse, om~dist+ffreq, meuse.grid, family = gaussian(log))
om.rk <- predict(omm, meuse.grid)
plot(om.rk)
```

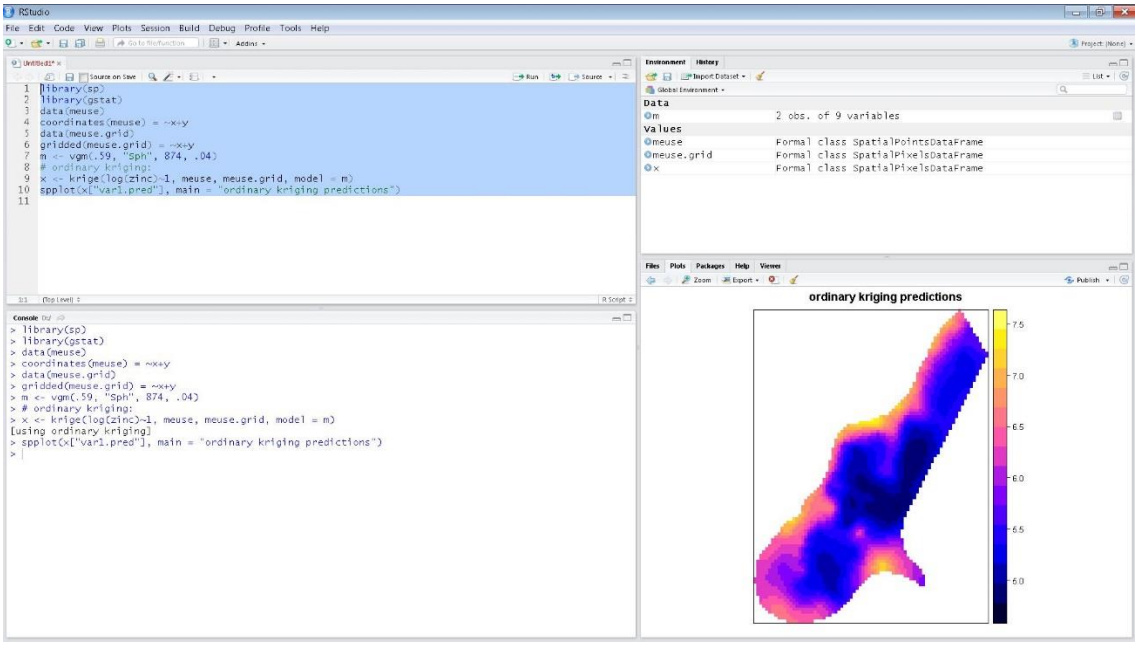
This should give the following output:



Also try running:

```
library(sp)
library(gstat)
data(meuse)
coordinates(meuse) = ~x+y
data(meuse.grid)
gridded(meuse.grid) = ~x+y
m <- vgm(.59, "Sph", 874, .04)
# ordinary kriging:
x <- krige(log(zinc)~1, meuse, meuse.grid, model = m)
spplot(x["var1.pred"], main = "ordinary kriging predictions")
```

This should give the following output:

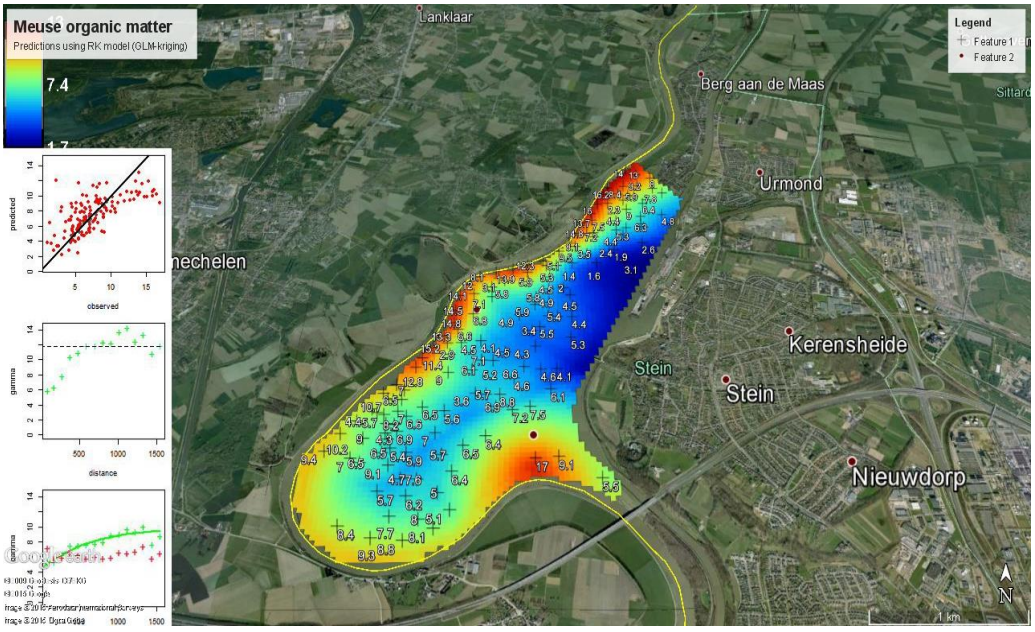


Testing: R – Google Earth (not mandatory)

Plotting the object `om.rk` using `plotKML` package (requires the first example to have run succesfully):

```
plotKML(om.rk)
```

should return the following if you have installed Google Earth installed on your machine (colour legend might differ):



Testing: R - SAGA GIS (not mandatory)

SAGA GIS is an extensive GIS geo-processor software with over 600 functions. SAGA GIS cannot be installed from RStudio (it is not a package for R). Instead, you need to install SAGA GIS using the installation instructions from the software homepage. After you have installed SAGA GIS, you can send processes from R to SAGA GIS by using the `saga_cmd` command line interface:

```
# define path to SAGA command line
saga_cmd = "C:/Progra~1/SAGA-GIS/saga_cmd.exe"

# load packages
library(GSIF)
library(rgdal)
library(raster)
library(plotKML)

# load grid (included in the plotKML package)
data("eberg_grid")
gridded(eberg_grid) <- ~x+y
proj4string(eberg_grid) <- CRS("+init=epsg:31467")

# write grid to GeoTiff
writeGDAL(eberg_grid["DEMSRT6"], "DEMSRT6.sdat", "SAGA")

# create a hillshade raster from the DEM
system(paste(saga_cmd, 'ta_lighting 0 -ELEVATION "DEMSRT6.sgrd" -SHADE "hillshade.sgrd" -
EXAGGERATION 2'))
```

This should give the following output:

```
> system(paste(saga_cmd, 'ta_lighting 0 -ELEVATION "DEMSRT6.sgrd" -SHADE "hillshade.sgrd" -EXAGGERATION 2'))
_________________________
   #####   ##   #####    ##
  ###     ###  ##       ###
   ###   # ## ##  #### # ##
    ### ##### ##    # #####
 ##### #   ##  ##### #   ##
_________________________

SAGA Version: 4.0.0 (64 bit)

_________________________
library path: C:\Progra~1\SAGA-GIS\tools\
library name: ta_lighting
library     : Lighting, Visibility
tool        : Analytical Hillshading
author      : O.Conrad, V.Wichmann (c) 2003-2013
processors  : 4 [4]
_________________________

Load grid: DEMSRT6.sgrd...

100%okay


Parameters


Grid system: 100; 100x 100y; 3570050x 5708050y
Elevation: DEMSRT6
Analytical Hillshading: Analytical Hillshading
Shading Method: Standard
Azimuth: 315.000000
Height: 45.000000
Exaggeration: 2.000000


101%Save grid: hillshade.sgrd...

100%okay
>
```

Testing: R – GDAL (not mandatory)

Another very important software for handling spatial data (and especially for exchanging / converting spatial data) is GDAL. GDAL also needs to be installed separately and then can be called from command line (note the example below builds on the R-SAGA example on the previous page, this):

```
# define paths to gdal functions
if(.Platform$OS.type == "windows"){
  gdal.dir <- shortPathName("C:/Program files/GDAL")
  gdal_translate <- paste0(gdal.dir, "/gdal_translate.exe")
  gdalwarp <- paste0(gdal.dir, "/gdalwarp.exe")
} else {
  gdal_translate = "gdal_translate"
  gdalwarp = "gdalwarp"
}

# reproject to WGS84
system(paste(gdalwarp, ' DEMSRT6.sdat DEMSRT6_ll.tif -t_srs \"+proj=longlat
+datum=WGS84\"'))

# plot map
plot(raster("DEMSRT6_ll.tif"))
```

This should give the following output:

Resources

The 'Hands-on Digital Soil Mapping' course is a very intensive one-week training. During this week we will not have time to teach you the R basics. If you are new to R, we strongly recommend to do some 'self-study' before coming to the spring school. 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:

- R online tutorial: http://www.r-tutor.com/
- RStudio tutorial: https://www.rstudio.com/online-learning/
- A Crash Course in R: http://spatial.ly/2013/05/crash/
- R introductions on YouTube, e.g.: https://www.youtube.com/watch?v=qEJHYIa-EhI or https://www.youtube.com/watch?v=7cGwYMhPDUY
- R reference card: https://cran.r-project.org/doc/contrib/Baggott-refcard-v2.pdf
- R general search: http://rseek.org/

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

- Beaudette: "R: advanced statistical package"
- Rossiter: "R applications"

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.