

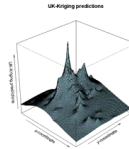
SimSurvey - an R-based E-learning tool for geo-statistical analyses

Mario Gellrich¹, Rudolf Gubler², Andreas Papritz¹, Andreas Schönborn³,
Rainer Schulin¹

¹Institute of Terrestrial Ecosystems, Swiss Federal Institute of Technology ETH Zürich (<http://www.ethz.ch>)

²Terraplan Gubler (gubler@terraplan.ch)

³Armadillo Media GmbH (<http://www.armadillo-media.ch>)



1. Introduction
2. Aim of the project
3. SimSurvey
 - 3.1 Structure
 - 3.2 Functionality
4. The graphical user interface (GUI)
 - 4.1 How does the GUI work?
 - 4.2 What's special?
5. State of project/outlook

1. Introduction

SimSurvey

2. Aims Of The Project

SimSurvey

Problems:

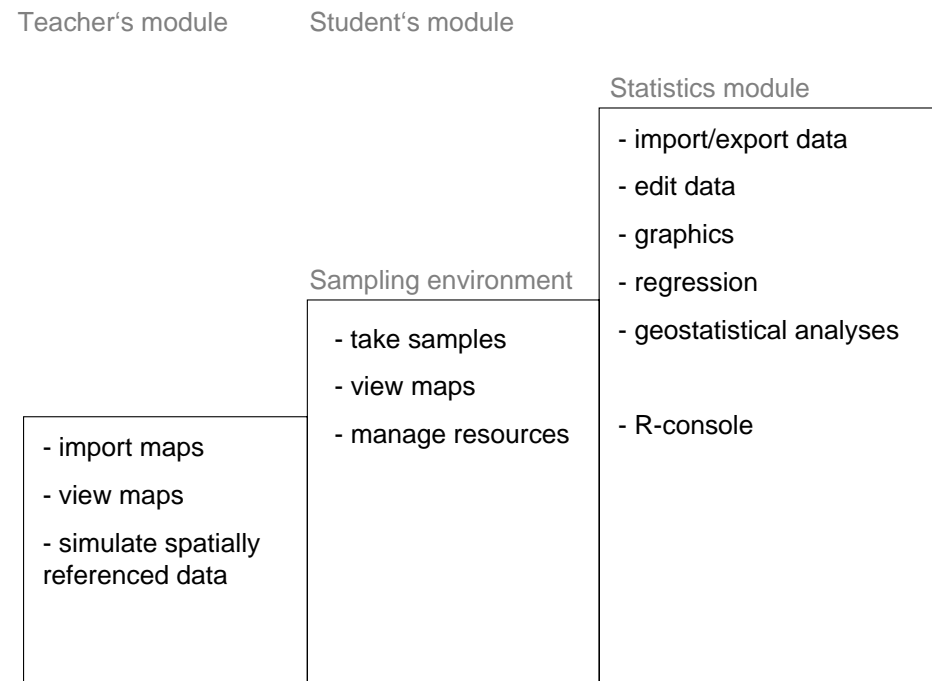
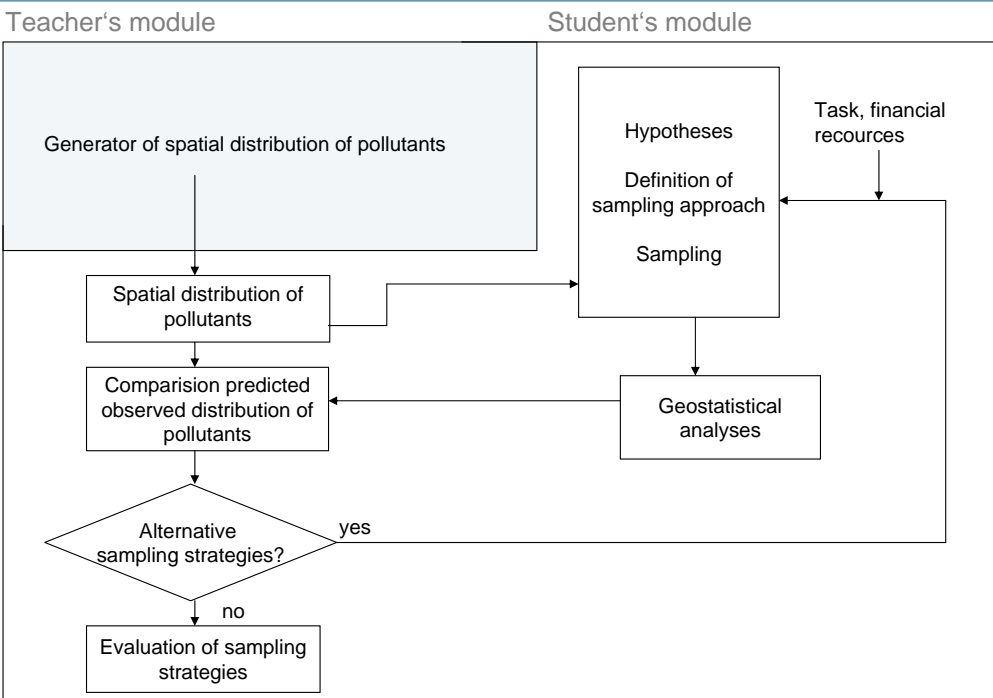
1. Geostatistics is part of the curriculum of environmental scientists, but topic is quite difficult for students
2. Students often have little (or no) R-programming skills
3. Installation and updating of R on multiple computers is often time-consuming

The aims of the project are to:

- a) make the learning of geostatistics easier
- b) complement lessons for students



Development of a web-based E-learning tool



SimSurvey is installed on a Linux-Server and runs in a browser

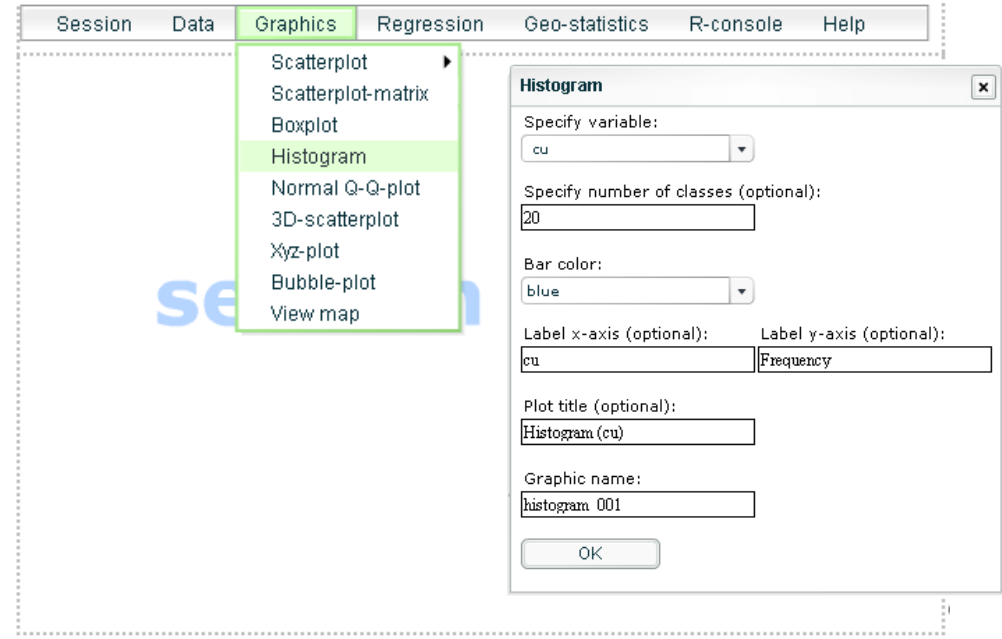
The following software is required:

- Apache (web server)
- Macromedia Flash Player 8 (visualisation)
- PHP (interface between Flash, R and MySQL)
- R (statistics, text-output, graphics-output)
- MySQL (database to block 'dangerous' commands in the R-console)

The graphical user interface (GUI)



Screenshot menu

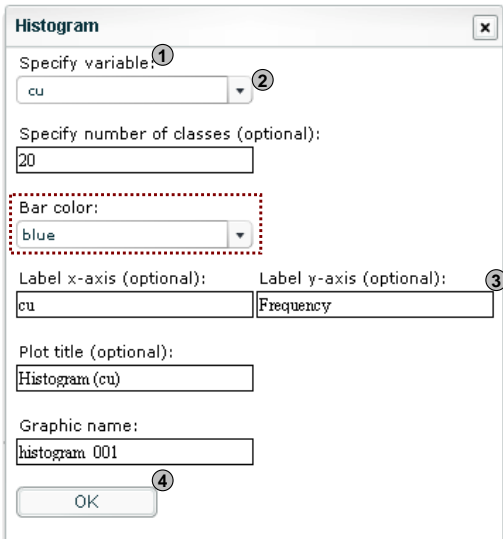


Graphics-menu & histogram dialogbox

4.1 How does the GUI work?

4.1 How does the GUI work?

1. Flash uses an XML-file to create a dialogbox



Flash-items:

- ① label
- ② dropdown box
- ③ textbox
- ④ button

2. The value of the Flash-item **hisDdmCol** is "blue"



XML-code

```
<outputField column="left" text="Bar color:"
name="hisTxt003"/>
  <dropDown column="left" name="hisDdmCol">
    <dropDownItem text="gray" value="gray"/>
    <dropDownItem text="green" value="green"/>
    <dropDownItem text="yellow" value="yellow"/>
    <dropDownItem text="blue" value="blue"/>
    <dropDownItem text="red" value="red"/>
    <dropDownItem text="wheat" value="wheat"/>
  </dropDown>
```

Each Flash-item has a name and value

Macromedia Flash passes the value of **hisDdmCol** to PHP.

3. PHP takes the value of hisDdmCol and assigns it to a PHP-variable \$eigenschaften2

```
if ($_REQUEST["hisDdmV01"]) //graphics, histogram
{
    $ausdruck=$_REQUEST["hisTinGm"]; //title
    if (!$ausdruck)
    {
        $ausdruck="tmp";
    }
    $variable1=$_REQUEST["hisDdmV01"]; //variable
    $eigenschaften1=$_REQUEST["hisTinCla"]; //number of classes
    $eigenschaften2=$_REQUEST["hisDdmCol"]; //barcolor
    $eigenschaften3=$_REQUEST["hisTinXax"]; //label x
    $eigenschaften4=$_REQUEST["hisTinYax"]; //label y
    $eigenschaften5=$_REQUEST["hisTinTit"]; //plot titel
    $arbeitsverzeichnis=$projektppfad;
    $maske="$pfad/include/graphics/histogram.R";
    if (!file_exists("$projektppfad/$ausdruck/"))
    {
        mkdir("$projektppfad/$ausdruck/", 0777);
        chmod ("$projektppfad/$ausdruck/", 0777);
    }
    $grafikverzeichnis=$ausdruck;
}
```

PHP-code

\$eigenschaften2=\$_REQUEST["hisDdmCol"]; //barcolor ←

4. The PHP-variable "***eigenschaften2***" is part of a 'dynamic' R-script

```
setwd("***Arbeitsverzeichnis**/")
load(".RData")
library(grDevices)

user.workspace <- "***Arbeitsverzeichnis***"

##### dynamic part 1 ##### start
graphic.name <- "***Ausdruck***"
##### dynamic part 1 ##### end

path.to.new.workspace <- paste(c(user.workspace, "/"), graphic.name, collapse="")
setwd(path.to.new.workspace)
load(paste(c(user.workspace, "/", ".RData"), collapse=""))

##### dynamic part 2 ##### start

sel.var <- "***variable1***"
number.classes <- "**eigenschaften1**"
bar.color <- "***eigenschaften2***"
label.xaxis <- "**eigenschaften3**"
label.yaxis <- "**eigenschaften4**"
plot.title <- "**eigenschaften5**"

##### dynamic part 2 ##### end

if(number.classes == "") {
    hist(dat[[sel.var]], nclass=nrow(dat[[sel.var]])/10, col=bar.color,
        xlab=label.xaxis, ylab=label.yaxis, main= plot.title, cex.main=1)
}

if(number.classes != ""){
    hist(dat[[sel.var]], nclass=number.classes, col=bar.color,
        xlab=label.xaxis, ylab=label.yaxis, main= plot.title, cex.main=1)
}

save(list = ls(all=TRUE), file = ".RData")
setwd(user.workspace)
quit(save = "yes",status=0,runLast = FALSE)
```

R-code

bar.color <- "***eigenschaften2***"

5. PHP translates the 'dynamic' R-script into a standard R-script and runs it.

```
setwd("/home/mariog/public_html/simsurvey/tmp/uedi/dornach_project/dornach")

load(".RData")
library(grDevices)
user.workspace <- "/home/mariog/public_html/simsurvey/tmp/uedi/dornach_project/dornach"

##### dynamic part 1 ##### start
graphic.name <- "histogram_001"
##### dynamic part 1 ##### end

path.to.new.workspace <- paste(c(user.workspace, "/"), graphic.name, collapse="")
setwd(path.to.new.workspace)
load(paste(c(user.workspace, "/", ".RData"), collapse=""))

##### dynamic part 2 ##### start

sel.var <- "var4"
number.classes <- 20
bar.color <- "blue"
label.xaxis <- "cu"
label.yaxis <- "Frequency"
plot.title <- "Histogram (cu)"

##### dynamic part 2 ##### end

if(number.classes == ""){
    hist(dat[[sel.var]], nclass=nrow(dat[[sel.var]])/10, col=bar.color, xlab=label.xaxis, ylab=label.yaxis, main= plot.title, cex.main=1)
}

if(number.classes != ""){
    hist(dat[[sel.var]], nclass=number.classes, col=bar.color, xlab=label.xaxis, ylab=label.yaxis, main= plot.title, cex.main=1)
}

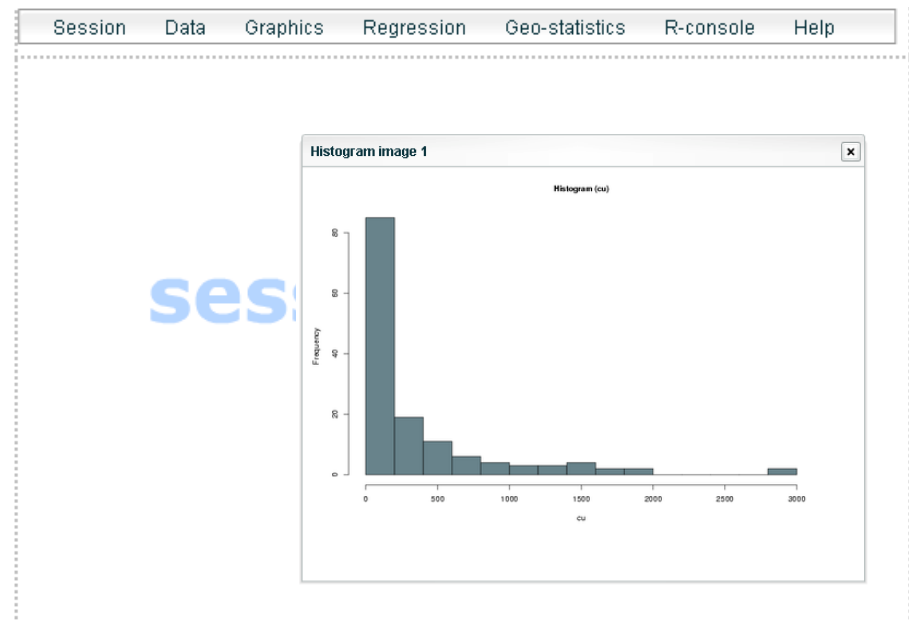
save(list = ls(all=TRUE), file = ".RData")
setwd(user.workspace)
quit(save = "yes",status=0,runLast = FALSE)
```

R-code

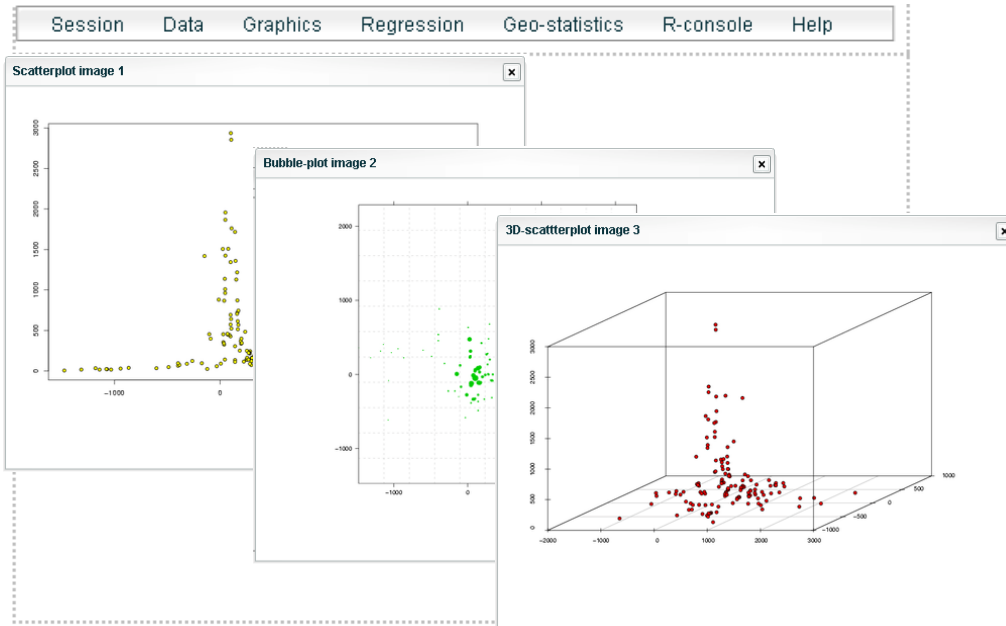
bar.color <- "blue"

col=bar.color

6. Flash takes the graphic produced by R and visualises it in the browser



The R-GUI produces graphic output in separate windows.



Text-output is displayed in a text-output window.

The screenshot shows the R-GUI interface with a menu bar (Session, Data, Graphics, Regression, Geo-statistics, R-console, Help) and two overlapping windows. The 'Linear regression' dialog box is open, showing options for target and independent variables, transformation, and estimation method. The 'Output window' displays the results of an ordinary least square regression.

Linear regression dialog box:

- Specify (one) target variable:
 - x
 - y
 - studie
 - cu
 - zn
 - cd
 - x.siedlung
 - y.siedlung
- Specify (one or more) independent variable(s):
 - x
 - y
 - studie
 - cu
 - zn
 - cd
 - x.siedlung
 - y.siedlung
- Specify transformation for target variable:
 - Log
- Specify estimation method:
 - lm
 - lqs
 - wlm
 - rlm method M
- Specify model name:
 - lm_1

Output window:

Ordinary least square regression (using 'lm'):

Model name: lm_1

Formula: $\log(\text{cu}) \sim x + y + \text{studie} + \text{cd}$

Estimated coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	4.1819	0.1910	21.8927	0.0000
x	-0.0003	0.0001	-2.6595	0.0088
y	0.0000	0.0002	-0.0091	0.9927
Studie.wirz	-0.4640	0.1650	-2.8123	0.0056
cd	0.7125	0.0679	10.4957	0.0000

Degrees of freedom: 136
F-statistic: 49.16
p-value: 1.594e-25

4.2 What's special?

5. State of project/outlook

What is special in R-GUI compared to other R-web interfaces*?

1. R processes run via "socket connections" (faster than "batch mode")
2. Interface between R and Macromedia Flash
3. GUI can easily be extended (modular construction system)
5. Flash-R interface user-friendly (movable dialog-boxes and output-windows in one browser-window)

Done:

- Project/Session management implemented
- R-GUI implemented (graphics, regression, geo-statistics, R-console)

To do:

- Sampling environment/handling of resources
- Tests/bug-fixing

• Outlook:

- bug-fixed alpha-version available in Summer 2007
- distribution of R-GUI/SimSurvey as open source software for teaching

*e.g. R-php (<http://dssm.unipa.it/R-php>) or Rweb (<http://www.math.montana.edu/Rweb>)

Many thanks,
questions/suggestions are
welcome!!!

Mario Gellrich (mario.gellrich@env.ethz.ch)

