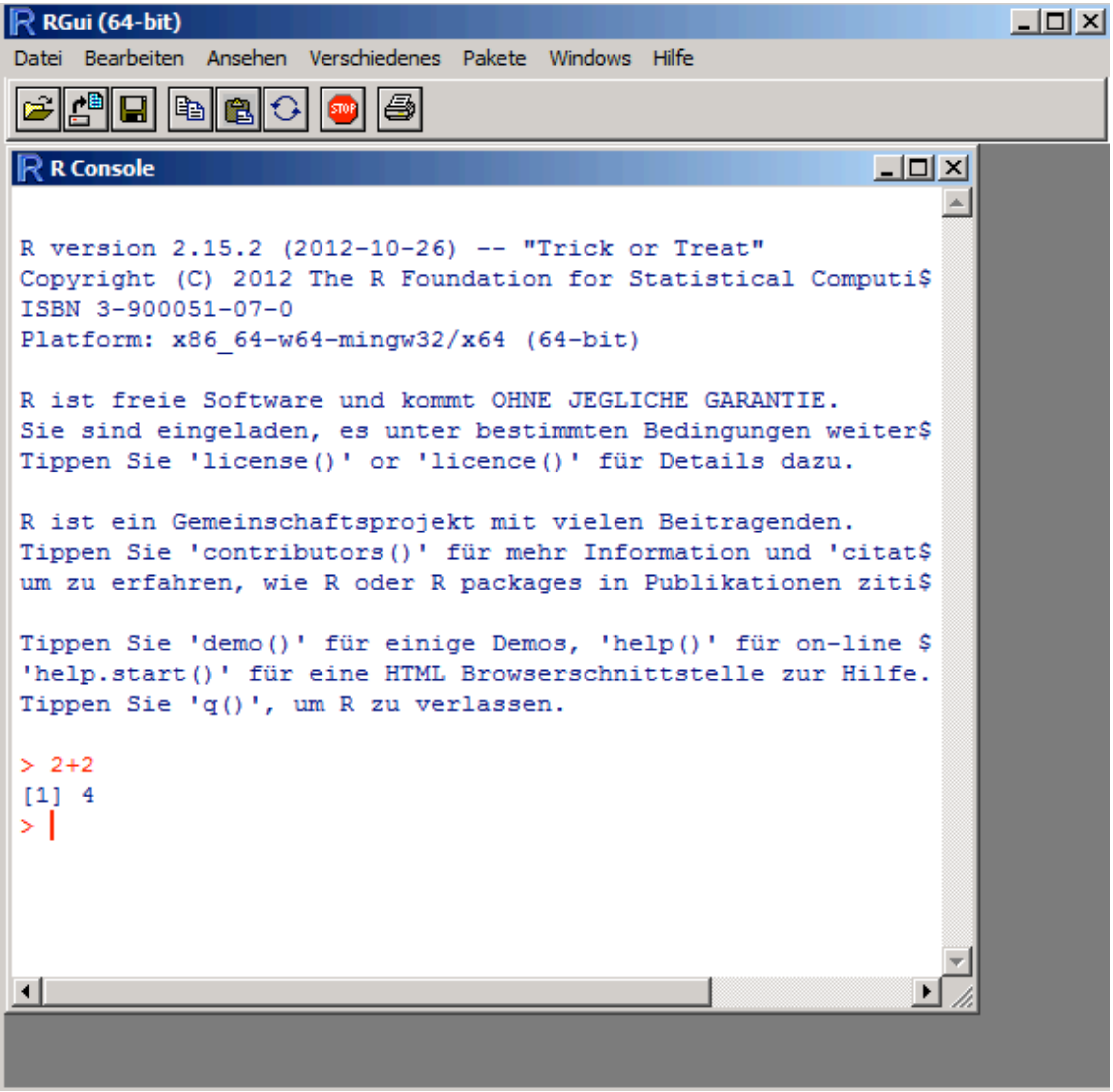
Important: if you had problems to follow the practical

- This series of video lectures on youtube basically goes through the same steps http://www.youtube.com/playlist?list=UUAeWj0GhZ94wuvOIYu1XVrg
- This website, which I also recommended in the stats lecture notes, has a list of written tutorials http://biometry.github.io/APES/R/R10-gettingStarted.html

Practical Graphics in R (ca. 1h)

We will work in RStudio, but as a demonstration, let's have a look at the RGui first. Start up the program R from your programs in Windows, look at the window that popping up, and try writing 2+2 in the main window



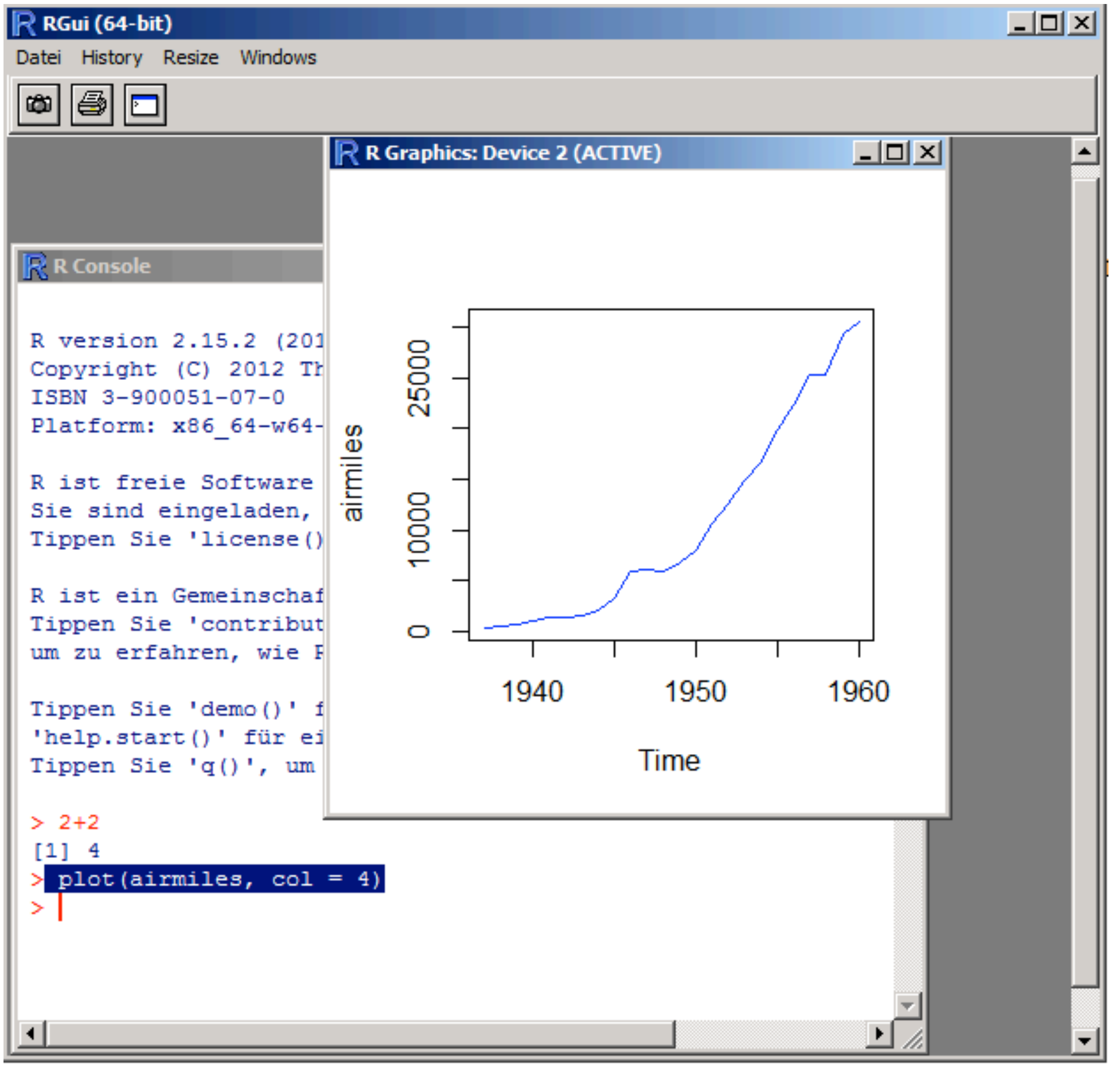
```
R version 2.15.2 (2012-10-26) -- "Trick or Treat"
Copyright (C) 2012 The R Foundation for Statistical Computi$
ISBN 3-900051-07-0
Platform: x86_64-w64-mingw32/x64 (64-bit)

R ist freie Software und kommt OHNE JEGLICHE GARANTIE.
Sie sind eingeladen, es unter bestimmten Bedingungen weiter$
Tippen Sie 'license()' or 'licence()' für Details dazu.

R ist ein Gemeinschaftsprojekt mit vielen Beitragenden.
Tippen Sie 'contributors()' für mehr Information und 'citat$
um zu erfahren, wie R oder R packages in Publikationen ziti$

Tippen Sie 'demo()' für einige Demos, 'help()' für on-line $
'help.start()' für eine HTML Browserschnittstelle zur Hilfe.
Tippen Sie 'q()', um R zu verlassen.

> 2+2
[1] 4
> 
```

This main window, called the R Console, is how you interact with the core R program that does all the communcation. Let’s write something else

R has a few standard datasets that are already loaded to try out stuff. Before we go to RStudio, let’s try out plotting one of those. Type

```
plot(airmiles, col = 4)
```

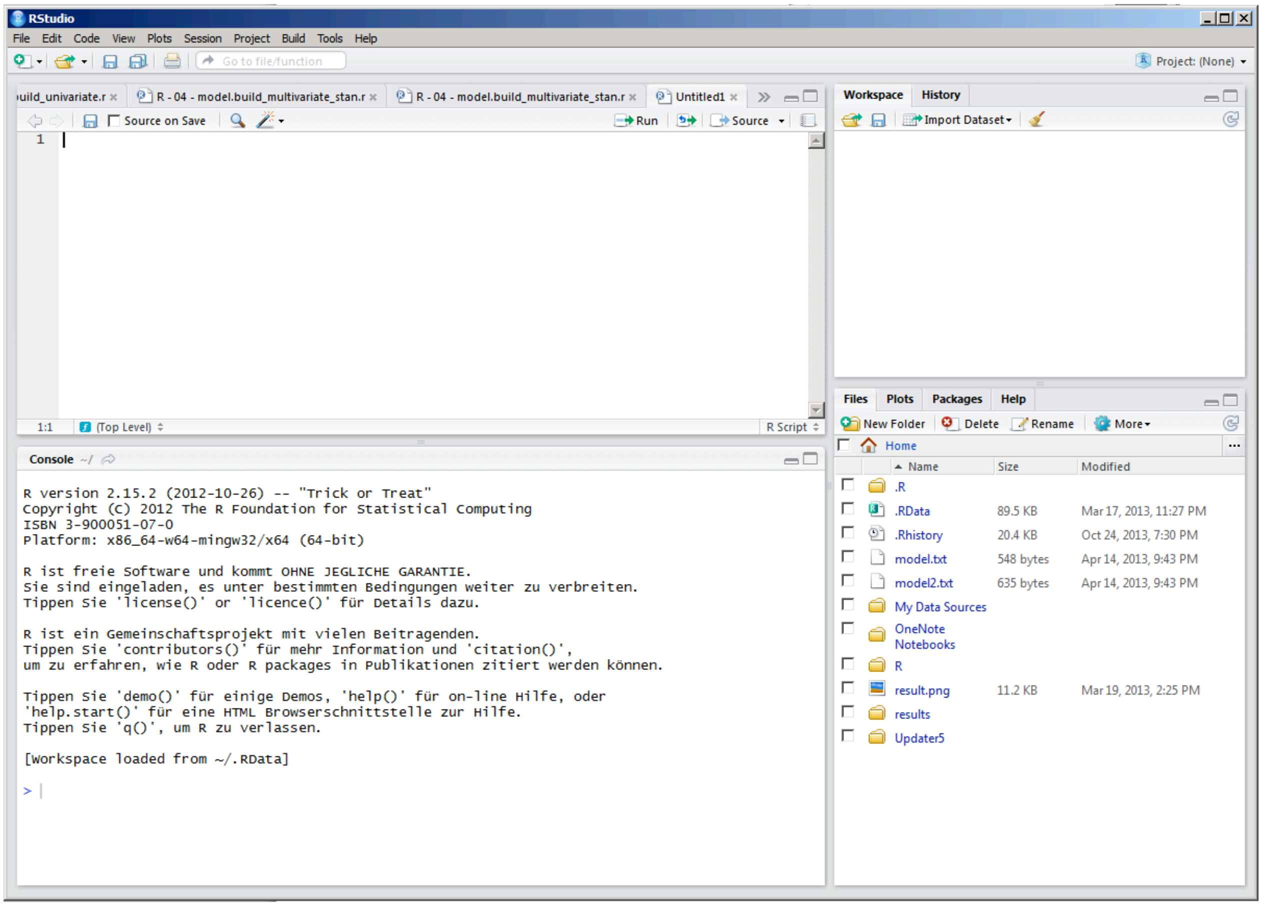
The result should look like that



```
R version 2.15.2 (201
Copyright (C) 2012 Th
ISBN 3-900051-07-0
Platform: x86_64-w64-

R ist freie Software
Sie sind eingeladen,
Tippen Sie 'license()

R ist ein Gemeinschaf
Tippen Sie 'contribut
um zu erfahren, wie F

Tippen Sie 'demo()' f
'help.start()' für ei
Tippen Sie 'q()', um

> 2+2
[1] 4
> plot(airmiles, col = 4)
>
```

So, as you see, this apparently plots airmiles against time. We’ll discuss why and how this works, but let’s first move to RStudio, an alternative program to interact with R.

RStudio basically offers the same functions as the RGui, but a few things can be done easier or are handled more comfortable for you. This is how it looks like



```
R version 2.15.2 (2012-10-26) -- "Trick or Treat"
Copyright (C) 2012 The R Foundation for Statistical Computing
ISBN 3-900051-07-0
Platform: x86_64-w64-mingw32/x64 (64-bit)

R ist freie Software und kommt OHNE JEGLICHE GARANTIE.
Sie sind eingeladen, es unter bestimmten Bedingungen weiter zu verbreiten.
Tippen Sie 'license()' or 'licence()' für Details dazu.

R ist ein Gemeinschaftsprojekt mit vielen Beitragenden.
Tippen Sie 'contributors()' für mehr Information und 'citation()',
um zu erfahren, wie R oder R packages in Publikationen zitiert werden können.

Tippen Sie 'demo()' für einige Demos, 'help()' für on-line Hilfe, oder
'help.start()' für eine HTML Browserschnittstelle zur Hilfe.
Tippen Sie 'q()', um R zu verlassen.

[Workspace loaded from ~/.RData]

> 
```

Name	Size	Modified
.R		
.RData	89.5 KB	Mar 17, 2013, 11:27 PM
.Rhistory	20.4 KB	Oct 24, 2013, 7:30 PM
model.txt	548 bytes	Apr 14, 2013, 9:43 PM
model2.txt	635 bytes	Apr 14, 2013, 9:43 PM
My Data Sources		
OneNote Notebooks		
R		
result.png	11.2 KB	Mar 19, 2013, 2:25 PM
results		
Updater5		

The console is in the bottom left … try to put in the same commands as we did before, i.e. 2+2, and plotting the graph

On the top of the console, r script files that are open are displayed … you can work of course in the console, typing everything in directly, but typically you work in the script and then send the stuff you were writing to the console. Typical script

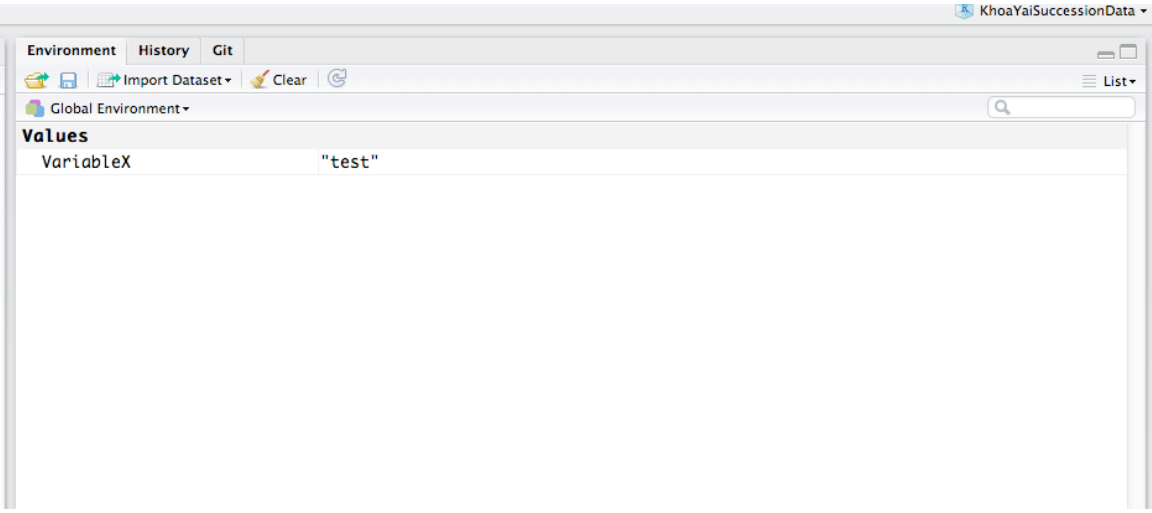
```
# the hash means this is treated as a comment
# this file is written by FH, 25.10.13

rm(list=ls(all=TRUE))  # this command means all variables in the memory are erase

# load some data

# do some plots
```

Variables

Who has ever worked with a programming language? In a programming language, data is stored in variables / objects

```
VariableX = “test”
```

Now the content I have specified (i.e. the word “test”) is stored in the computer. Note that this is shown in RStudio in the right hand corner, under Environment



```
Values
  VariableX                    "test"
```

I can access the variable by typing its name in the console, and it will return the value that is stored.

```
> VariableX
[1] "test"
```

Data types and structures

A variable can store different things: a number, a word, or a whole dataset. If it is a single value, we call it in R an atomic type. Important atomic types are:

boolean, numeric, factor, character

If we have a collection of several atomic types, we speak of a data structure or an object (there is a difference but it doesn’t matter here). Important examples of this are

vector (a row of the same atomic types, e.g. 1,2,3,4,5), and data.frame (a list of vectors, this is the standard format for data)

A full list of data types is here http://www.statmethods.net/input/datatypes.html

If you want to see which type or structure a variable has:

str() gives you the structure of a variable / object
summary() tries to give you a summary, depending on the type you have
plot() tries to make a plot, depending on the type you have

Try this out with the object airquality.

The data frame

Most common structure in r, basically columns are stored as a list of vectors, so each column can be a different data type

You can select columns in a number of ways

By name: airquality$Ozone
By index: airquality[,1]

Slicing: try out airquality[,1:2], airquality[4:6,1], airquality[c(1,2,3,4,7,8),1] … what do you get?

How to load data?

Try to load the airquality.txt file (provided to you) with

- with RStudio
- from the script with the read.table() option.

After loading the data, always check whether the data format is correct (numeric, factors) with the str(YOURDATA) command.

After loading the data, check whether the types are as you want them (sometimes numeric is read in as factor, or vice versa). If a column would have the wrong type, we have to change this, e.g. as.factor(x) or as.numeric(x)

See http://www.statmethods.net/input/importingdata.html for more options, e.g. excel import.

Note: this data file is already included in R, so if you don't manage to load it, you can continue anyway.

Plotting

OK, now you know how to select data from your data

Lets do some standard plots

plot(airquality) automatically chooses a particular plot, namely pairs(airquality)

Lets plot only two variables against each other

plot(airquality$Ozone, airquality$Temp) … or should it rather be the other way around?
plot(airquality$Temp , airquality$Ozone)

```
hist(airquality$Ozone)
```

Getting help and modifying functions

Functions in R have typically a lot of options … you get them by looking at the help of a function, either on it and F1, or ?functionname, try ?hist

Default values can be overwritten

```
hist(airquality$Ozone, breaks = 30, col = "darkred")
```

Colors at http://research.stowers-institute.org/efg/R/Color/Chart/

Can also modify

```
plot(airquality$Ozone, airquality$Temp, pch = 5)
```

```
plot(airquality$Ozone, airquality$Temp, pch = 3, cex = airquality$Wind / 10, col =
"darkred")
```

Some more plotting functions

See also http://www.statmethods.net/graphs/index.html

```
# Simple Bar Plot
counts <- table(mtcars$gear)
barplot(counts, main="Car Distribution",
   xlab="Number of Gears")
```

```
# Grouped Bar Plot
counts <- table(mtcars$vs, mtcars$gear)
barplot(counts, main="Car Distribution by Gears and VS",
  xlab="Number of Gears", col=c("darkblue","red"),
  legend = rownames(counts), beside=TRUE)
```

```
# Boxplot of MPG by Car Cylinders
boxplot(mpg~cyl,data=mtcars, main="Car Milage Data",
   xlab="Number of Cylinders", ylab="Miles Per Gallon")

# Notched Boxplot of Tooth Growth Against 2 Crossed Factors
# boxes colored for ease of interpretation
boxplot(len~supp*dose, data=ToothGrowth, notch=TRUE,
  col=(c("gold","darkgreen")),
  main="Tooth Growth", xlab="Suppliment and Dose")
```

Modifying a function afterwards with inkscape

Short demonstration, you can try this later: Save as a pdf, open with inkscape, modify e.g. colors of the figure, but a different background, etc.

Fitting a model

We'll do this in more detail on Wednesday, but just as a preview:

```
fit = lm(airquality$Temp~airquality$Ozone)
abline(fit, col = "blue")

fit2 = lm(airquality$Temp~airquality$Ozone + I(airquality$Ozone^2))
```

check out ?formula to understand how to put in formulas

Websites with examples / tutorials

- **Start with http://www.statmethods.net/index.html**

More examples for graphics here:

- http://rgm3.lab.nig.ac.jp/RGM/R_image_list?page=3073&init=true
- https://www.stat.auckland.ac.nz/~paul/RGraphics/rgraphics.html
- http://rgraphgallery.blogspot.de/search/label/heatmap

- http://biometry.github.io/APES/R/R10-gettingStarted.html (list with R tutorials)

and google is your friend!