SOS2900

Seminar 1: Introduction to R and prediction

Torkild H. Lyngstad

18 januar 2018

What is R?

- R is a statistical programming language
  - Statistical analysis
  - Data visualization
  - Data mining
  - General programming
- It is open-source
  - Free as in “free beer” and “free speech”
  - Large active community around R
  - Many contributed packages

Why do you need to know R?

- R is very powerful, and allows us to manually compute everything we want if we choose to
- For many of the methods we use, R has either the simplest or the only implementation.
- A main tool among Data scientists

What do you need to know about R?

- In this course not much.
  - Tiny bits of programming
  - Cookbook method
  - Strongly advised to learn about R (or Python) for DS

Let's get started

- First use of R and RStudio
  - We will use the environment RStudio for our work in R
- What is the difference?
  - R is the program interpreter. Does the job.
  - RStudio is a "front end" that makes it easier to use R.
- RStudio requires R.

Meet RStudio!

- RStudio has 4 panels:
  - Console: This is the actual R window, you can enter commands here and execute them by pressing enter
  - Source: This is where we can edit scripts. It is where you do most of your work
    - Control-enter sends *selected* code to R (in the console)
    - Control-shift-enter sends *entire* script to R (in the console)
  - Plots/Help: This is where plots and help pages will be shown
  - Workspace: Shows which objects you currently have, working directory contents
  - Anything following a # symbol is treated as a comment!

RStudio

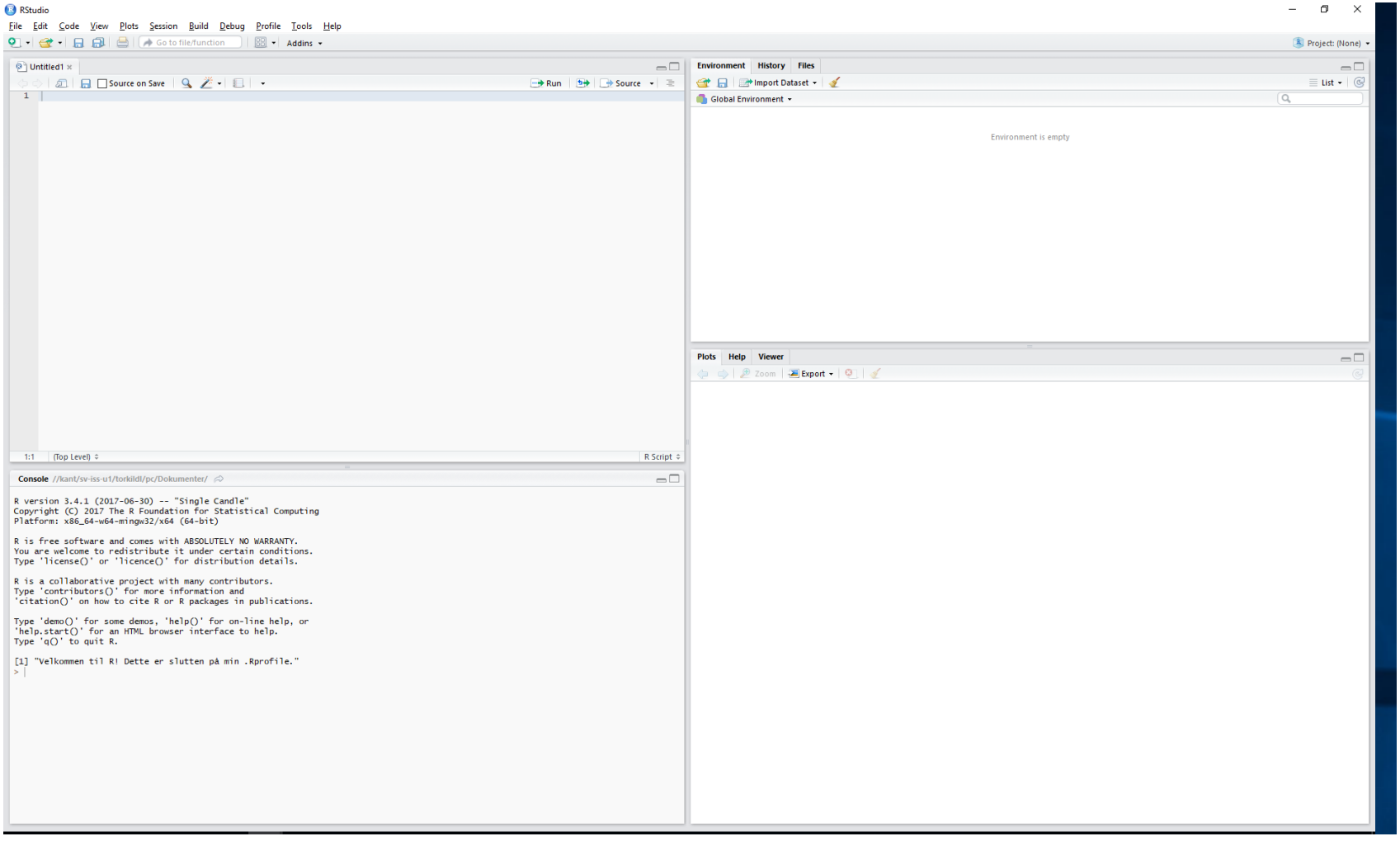

Figure 1:

R as calculator

- You can use R as a calculator

```
1 + 1

## [1] 2

(10 * 20) / 100

## [1] 2

exp(1.5 * (2.1 - 1.8)) / (1 + exp(1.5 * (2.1 - 1.8)))

## [1] 0.6106
```

Figure 2:

In R, everything has a name

- Assign and use objects
  - The <- operator can be used to store values into objects
  - An object can contain anything in R
  - R expressions that are not stored in an object are printed
- Object names are...
  - Unique: If you use the same name, the old will be *overwritten*
  - Sensitive: MyData is *not the same* as mydata

We can use R as a calculator

```
1 + 1

## [1] 2

(10 * 20) / 100

## [1] 2

exp(1.5 * (2.1 - 1.8)) / (1 + exp(1.5 * (2.1 - 1.8)))

## [1] 0.6106
```

Figure 3:

Numeric and character data

- In R, there are several types of data:
  - Numeric: numbers of all kinds
  - Character: one of more text character (which may be a number!)
    - Multiple characters in a row = strings
  - Logical: TRUE or FALSE

Numeric data

```
# Numbers:
1.5

## [1] 1.5

10

## [1] 10
```

Figure 4:

Character data

```
# Strings (within single or double quotes):
'this is a string'

## [1] "this is a string"

"this is also a string"

## [1] "this is also a string"

"
This is a very long string

with multile lines
"

## [1] "\nThis is a very long string\n\nwith multile lines\n"
```

Figure 5:

Assigning stuff to objects

```
a <- 1
a

## [1] 1

b <- 2
a + b

## [1] 3

a <- a + b
a

## [1] 3

b

## [1] 2
```

Figure 6:

How does data get into R?

- We can type it into ourselves
  - Not going to work with data sets of interesting sizes.
- Use *read.csv* or to read simple text files (or *data.table::fread*)
- Convert from other software (Excel, Stata, etc.) using R's *readxl* or *haven* libraries

Data frames (or tibbles)

- Data sets is represented as *data frames*
- R can deal with many data sets at once
  - Data frames stored under different names
  - Stata/SPSS: One data set at the time
- For our purposes:
  - Summarize one data set using regression (old data, "training data")
  - Predict on *another data set* (new data, "testing data")

Looking at a data frame

```
>
> head(iris)
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1          5.1         3.5          1.4         0.2  setosa
2          4.9         3.0          1.4         0.2  setosa
3          4.7         3.2          1.3         0.2  setosa
4          4.6         3.1          1.5         0.2  setosa
5          5.0         3.6          1.4         0.2  setosa
6          5.4         3.9          1.7         0.4  setosa
> summary(iris)
  Sepal.Length    Sepal.Width     Petal.Length    Petal.Width          Species
 Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100   setosa    :50
 1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300   versicolor:50
 Median :5.800   Median :3.000   Median :4.350   Median :1.300   virginica :50
 Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199
 3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800
 Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500
>
```

Figure 7:

More info about R for Data science

- Enormous amount of material on internet
- Look e.g. at this free book: `http://r4ds.had.co.nz/`
  - A full course in using R for data science
- Many, many relevant books. Examples:
  - Kosuke Imai: **Quantitative data analysis**
  - Kieran Healy: **Data visualization for the social sciences**

Nano data science!

- Let us do something REAL!
  - Load data set, and store it in an object
  - Look at data set
  - Summarize data set
  - Model data using linear regression
  - Predict from model using same data
  - Evaluate prediction
  - Model again, but better