UNIVERSIDAD PONTIFICIA
ICAI ICADE
COMILLAS
MADRID

Advanced techniques in R

Contents

1. Preparing data for machine learning.

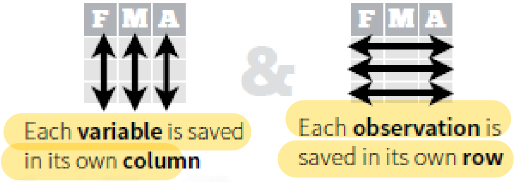
2. Exploratory analysis.

3. ggplot2 graphics library.

Preparation of data for machine learning

Data structure for machine learning

- Machine learning consists in extracting information from observed data. The data should be ordered in a meaningful way to perform analysis.
- We will work with data frames (tables) where:



Each **variable** is saved in its own **column**

Each **observation** is saved in its own **row**

- Useful tools for data wrangling:
  - *aggregate* and *rapply* functions.
  - *tidyr* and *dplyr* libraries.

- Type of variables:
  - Numeric: Continuous or discrete data.
  - Factors: Categorical variables.
  - Char: Variables containing text.

R is a high level language and many functions “interpret” what the user wants.
This is very useful, but can lead to mistakes.

Hints:

- Identify each variable and set the appropriate type.
- Identify NA values that can contaminate the analysis.

Preparation of data for machine learning

Example

Example_DataWrangling.R exemplifies how to:

- Load *Titanic* dataset.
- Identify each type of variable in the dataset. Make the necessary conversions.
- Identify missing values and eliminate those observations from the dataset.
- Create two datasets, one for each passenger sex. Then, join again the two datasets only with Age, Sex and Survived variables.

Exploratory analysis

- Exploratory analysis of data is essential in machine learning.

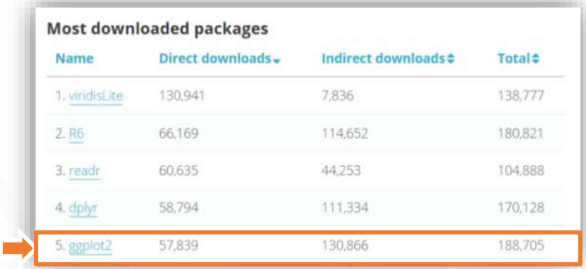
- Objectives:
  - Identify outliers.
  - Relations between input and output variables.

- Some tools available:
  - Summary functions (mean, variance, quantiles…).
  - Plotting
    - Plot(), coplot(), pairs(), …
    - Lattice library.
    - ggplot2 library.

ggplot2 library

Introduction

- Why ggplot2? → Top downloaded R packages



Most downloaded packages

Name	Direct downloads	Indirect downloads	Total
1. viridisLite	130,941	7,836	138,777
2. R6	66,169	114,652	180,821
3. readr	60,635	44,253	104,888
4. dplyr	58,794	111,334	170,128
5. ggplot2	57,839	130,866	188,705

Source: https://www.rdocumentation.org/trends. Visited 29/10/2017

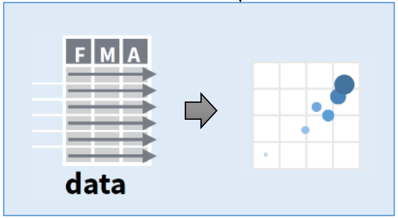
- ggplot2 implements the **grammar of graphics**, a coherent system for describing and building graphs.
- Some references:
  - Documentation: http://ggplot2.tidyverse.org/index.html
  - Introduction: http://r4ds.had.co.nz/data-visualisation.html
  - Book: H. Wickham (2016). *ggplot2: Elegant Graphics for Data Analysis*. 2nd Ed. Springer
  - Cheatsheet: https://github.com/rstudio/cheatsheets/raw/master/data-visualization-2.1.pdf
  - Top 50 visualizations: http://r-statistics.co/Top50-Ggplot2-Visualizations-MasterList-R-Code.html

ggplot2 library

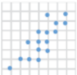
Basics

- **ggplot2** is based on the idea that you can build every graph from the same components: a **data set**, a **coordinate system**, and **geoms** (layers of graphics).
- The visual properties of the geoms are called **aesthetics.**
- Command example:

```
ggplot(data) + geom_point( aes(x = F, y = A, color = F, size = A))
```



- Description of syntax:
  - `ggplot(data)` → Begin a ggplot graphics using data as the dataset.
  - `geom_point()` → geom function to add a points to the graph.
    This function admits several aesthetic properties:



e + geom_point(), x, y, alpha, color, fill, shape, size, stroke

  - `aes()` → Function to specify aesthetic properties of the points depending on the values of variables:
    - `x = F` → The x coordinates of the points are given by variable F of data.
    - `y = A` → The y coordinates of the points are given by variable A of data.
    - `color = F` → The color of the points are given by the values of variable F of data.
    - `size = A` → The size of the points are given by the values of variable A of data.

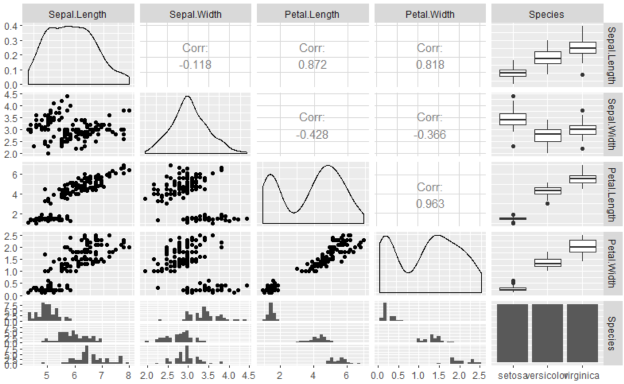
- Open Example_ggplot.R for more details.

ggplot2 library

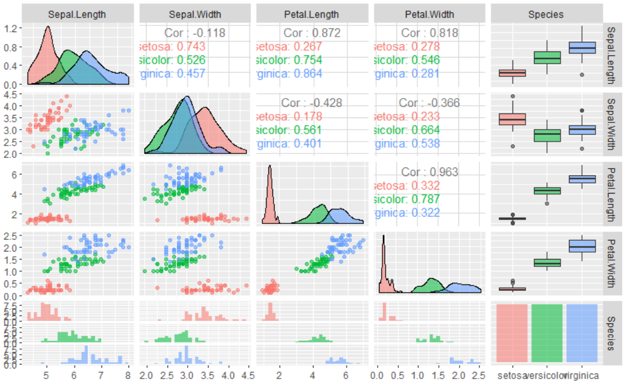
ggpairs

- `ggpairs()` function provides an extension of `pairs()` following ggplot philosophy.
- It is located in GGally package.
- Example with iris dataset:
  - Compute ggpairs plot for *iris* dataset.
  - Colour each plot in function of Species variable.

```
ggpairs(iris)
```



```
ggpairs(iris,
   aes(color = Species, alpha = 0.4))
```



Bibliography

- H. Wickham (2016). *ggplot2: Elegant Graphics for Data Analysis*. 2nd Ed. Springer.
- W. Chang (2012). *R Graphics Cookbook*. O'Reilly.
- D. Teutonico (2015). *ggplot2 Essentials*. Packt Publishing.
- L. Wilkinson (2003). *The Grammar of Graphics* (Statistics and Computing). 2nd Ed. Springer.

Alberto Aguilera 23, E-28015 Madrid - Tel: +34 91 542 2800 - http://www.iit.comillas.edu