MEDAlics Centro di Ricerca per le Relazioni Mediterranee

Università per Stranieri "Dante Alighieri" Reggio di Calabria

Istituto di Ricerca per l'Innovazione e la Tecnologia nel Mediterraneo

Città Metropolitana di Reggio Calabria

Machine Learning

Mariachiara Fortuna | 16 - 17 Marzo 2018

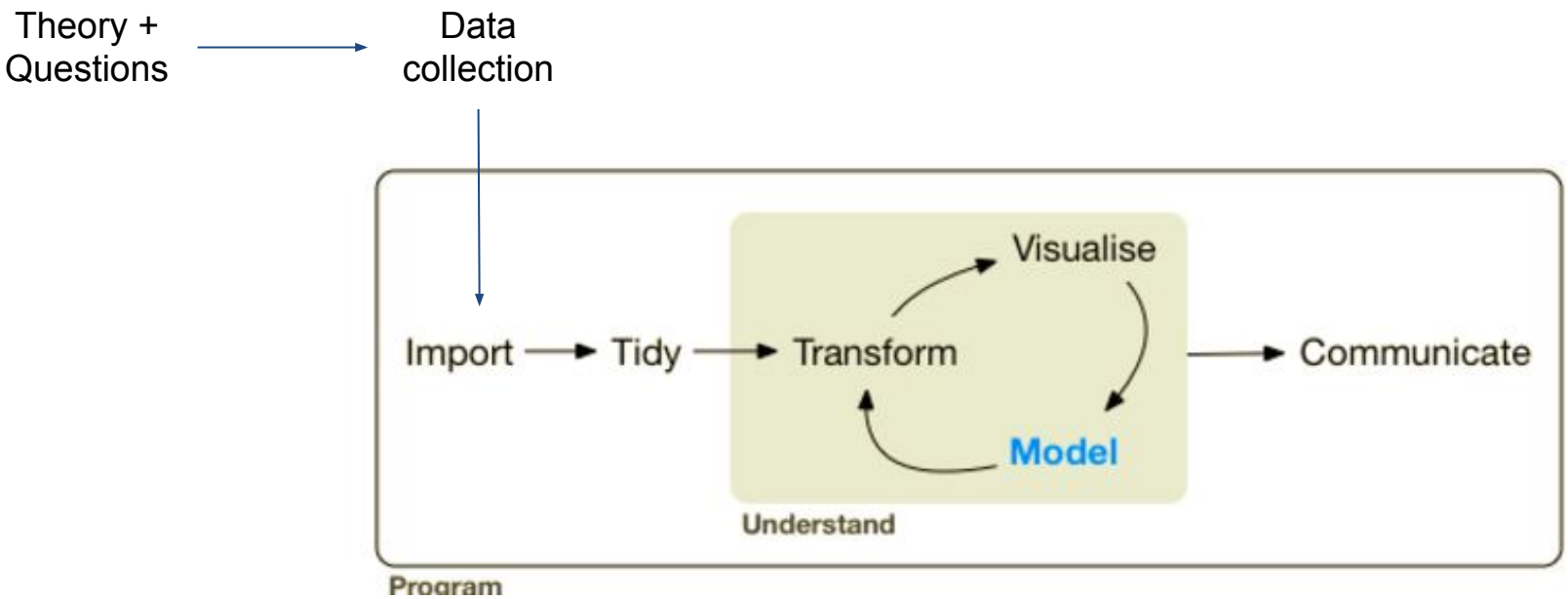
Class Materials

https://github.com/mariachiarafortuna/machineLearningClass

We will work on comics data, using  **!**

Data source: https://github.com/fivethirtyeight/data/tree/master/comic-characters
Inspiration: https://fivethirtyeight.com/features/women-in-comic-books/

The data analysis workflow



https://rviews.rstudio.com/2017/06/08/what-is-the-tidyverse/

"If a predictive signal exist in a set of data, many models will find some degree of that signal regardless of the technique or care placed in developing the model; as the saying goes, "Even a blind squirrel finds a nut"

Nevertheless, irrelevant information can drive down predictive performance of many models. Subject-specific knowledge can help separate potentially meaningful information from irrelevant information, eliminating detrimental noise and strengthening the underlying signal.

To summarize, the foundation of an effective predictive model is laid with **intuition** and **deep knowledge of the problem context**, which are entirely vital for driving decisions about model development.

The process begins with **relevant data**, another key ingredient.

The third ingredient is a **versatile computational toolbox** which includes techniques for data pre-processing and visualization as well as a suite for modelling tools for handling a number of possible scenarios."

"Applied Predictive Models" M. Kuhn, K. Johnson

Our computational toolbox: R

Why 

Free-licensing: everyone can download it for free, install it on an unlimited number of computers and update it whenever he wants.

Open source: everybody can improve the R code and develop new features

Enormous variety of statistical solutions: from machine learning to interactive dashboards, automatic reports, financial models, biomedical, environmental..

State of the art research: the statistical research is more and more developed with R

Easy to integrate: a huge number of integrations with data analysis or data management systems are already available: Oracle, SAS, SPSS, QlikView…

Big-data compliant

Basics

R download:
http://cran.mirror.garr.it/mirrors/CRAN/

RStudio desktop download:
https://www.rstudio.com/products/rstudio/download/

Code

Run current line/selection:
Ctrl + Enter

Help: `?function`

Comments: #

mean {base} R Documentation

Arithmetic Mean

Description

Generic function for the (trimmed) arithmetic mean.

Usage

```
mean(x, ...)

## Default S3 method:
mean(x, trim = 0, na.rm = FALSE, ...)
```

Arguments

x	An R object. Currently there are methods for numeric/logical vectors and date, date-time and time interval objects. Complex vectors are allowed for `trim = 0`, only.
trim	the fraction (0 to 0.5) of observations to be trimmed from each end of x before the mean is computed. Values of trim outside that range are taken as the nearest endpoint.
na.rm	a logical value indicating whether NA values should be stripped before the computation proceeds.
...	further arguments passed to or from other methods.

Example: `?mean`

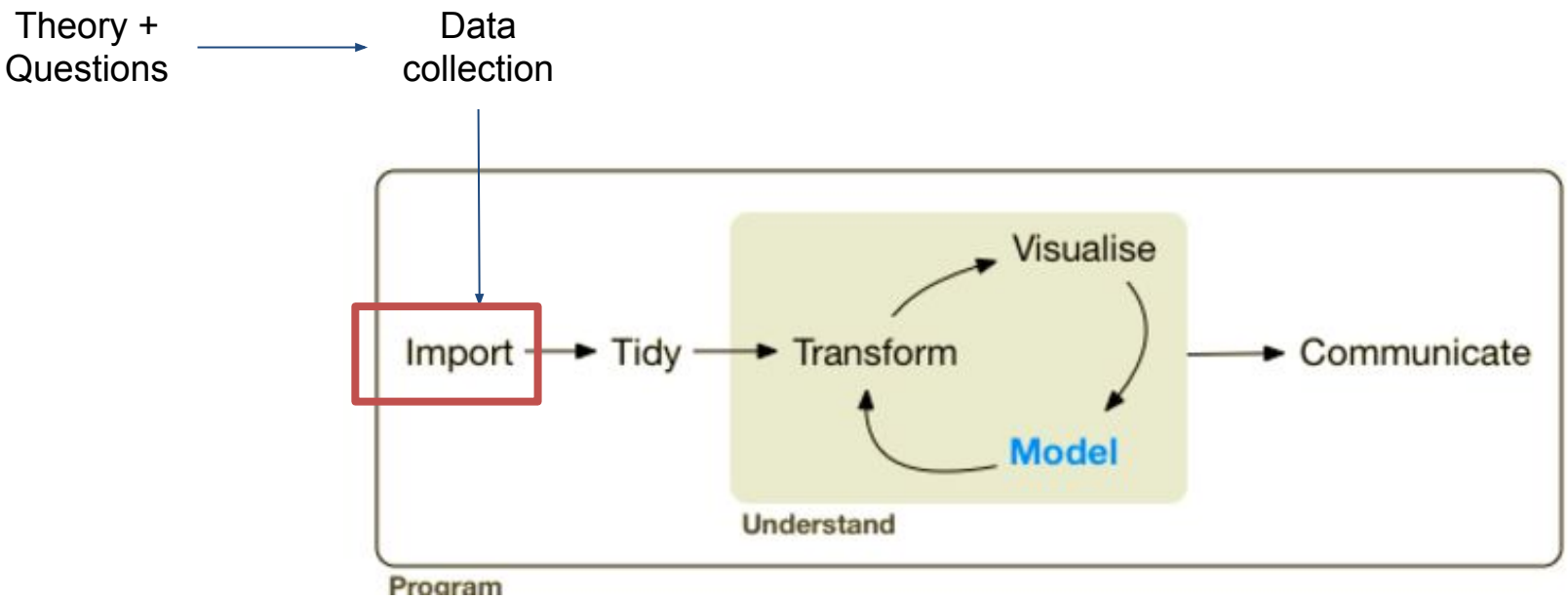
Basics

1. **File > New Project**
   A Project defines a precise “contexts” for your work, with its own working directory, workspace, history, and source documents.

2. **New file:**
   a. **R script**
      Contains only code (plus comments, preceded by #)
   b. **R Markdown / R Notebook:**
      Contains R code + extra text and narration. R Markdown will run the code and append the result to the doc
      - ➔ Write text formatted with markdown
      - ➔ Insert code chunk with Ctrl + Alt + i
      - ➔ Render the document with the Knit button, as an html, pdf or doc document (other options also available)

Exploratory data analysis

The data analysis workflow



https://rviews.rstudio.com/2017/06/08/what-is-the-tidyverse/

Import Data

Read tabular data

```
read.table(file, header = TRUE, sep = ",", dec = ".", ...)
read.csv
read.delim
```

(`header`: column names; `sep`: field separator character; `dec`: decimal sep.)

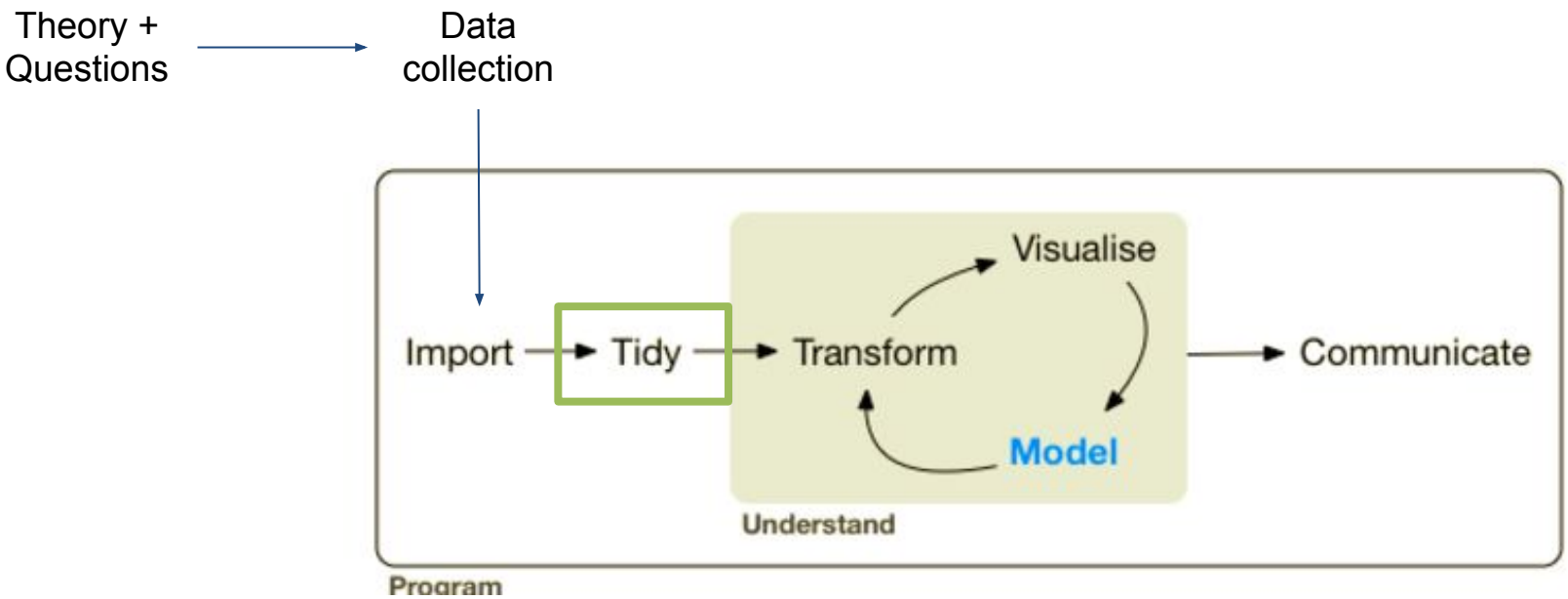
Read Excel data

```
library(readxl)
read_excel("file.xls")
```

Other formats

Via ad hoc libraries, R can query **databases** and **Spark clusters**, as well as importing **data of proprietary format**, as SAS, SPSS, Stata...

The data analysis workflow

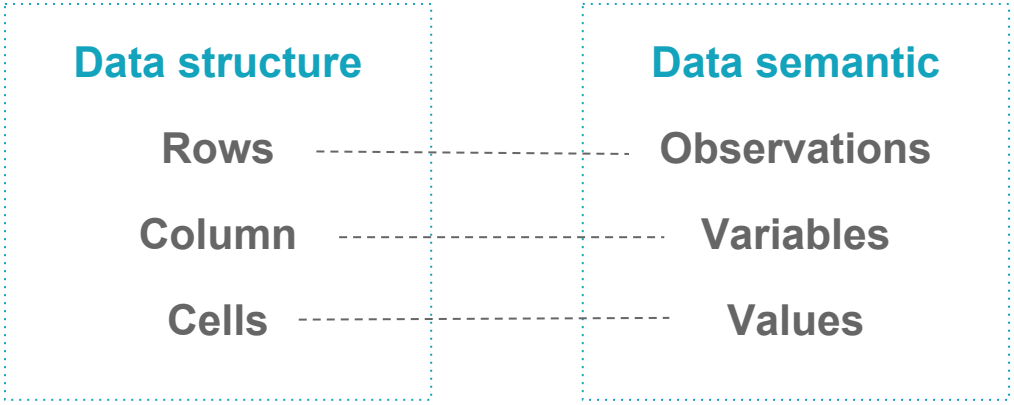


https://rviews.rstudio.com/2017/06/08/what-is-the-tidyverse/

Tidy Data

Tidy datasets provide a standardized way to link the structure of a dataset (its physical layout) with its semantics (its meaning).

https://cran.r-project.org/web/packages/tidyr/vignettes/tidy-data.html



Rules

Each observation forms a row

Each variable forms a column (a variable contains all values that measure the same underlying attribute across units)

Each type of observational unit forms a table

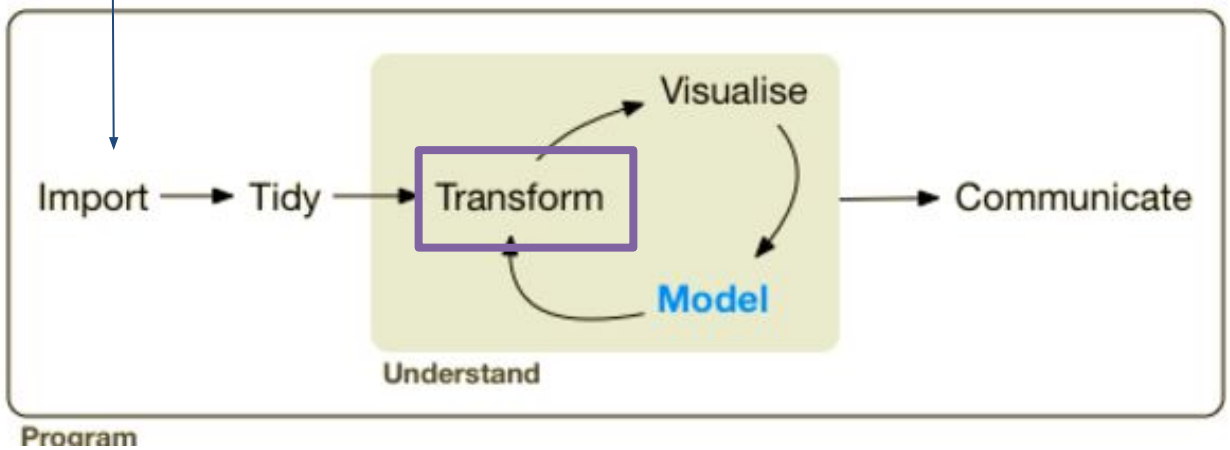
Common Messy Data

- Column headers are values, not variable names.
- Multiple variables are stored in one column.
- Variables are stored in both rows and columns.
- Multiple types of observational units are stored in the same table.
- A single observational unit is stored in multiple tables.

https://cran.r-project.org/web/packages/tidyr/vignettes/tidy-data.html

The data analysis workflow

Theory + Questions → Data collection



https://rviews.rstudio.com/2017/06/08/what-is-the-tidyverse/

dplyr

dplyr is a grammar of data manipulation, providing a consistent set of verbs that help you solve the most common data manipulation challenges:

mutate() adds new variables that are functions of existing variables
select() picks variables based on their names.
filter() picks cases based on their values.
summarise() reduces multiple values down to a single summary.
arrange() changes the ordering of the rows

These all combine naturally with **group_by()** which allows you to perform any operation “by group”.

http://dplyr.tidyverse.org/

dplyr

Select

Picks variables based on their names (or remove them).

```
select(data = marvel, name, EYE, HAIR)
select(data = marvel, -ID)
```

Filter

Picks cases based on their values

```
filter(data = marvel, SEX == "Female Characters")
filter(data = marvel, APPEARANCES < 20)
```

Arrange

Changes the ordering of the rows

```
arrange(data = marvel, desc(YEAR))
```

dplyr

Mutate

Adds new variables that are functions of existing variables

```
mutate(data = marvel, next_year = YEAR + 1)
```

Pipe operator

```
%>%
```

Chain different operations, passing the result of a function as first element of the following function.

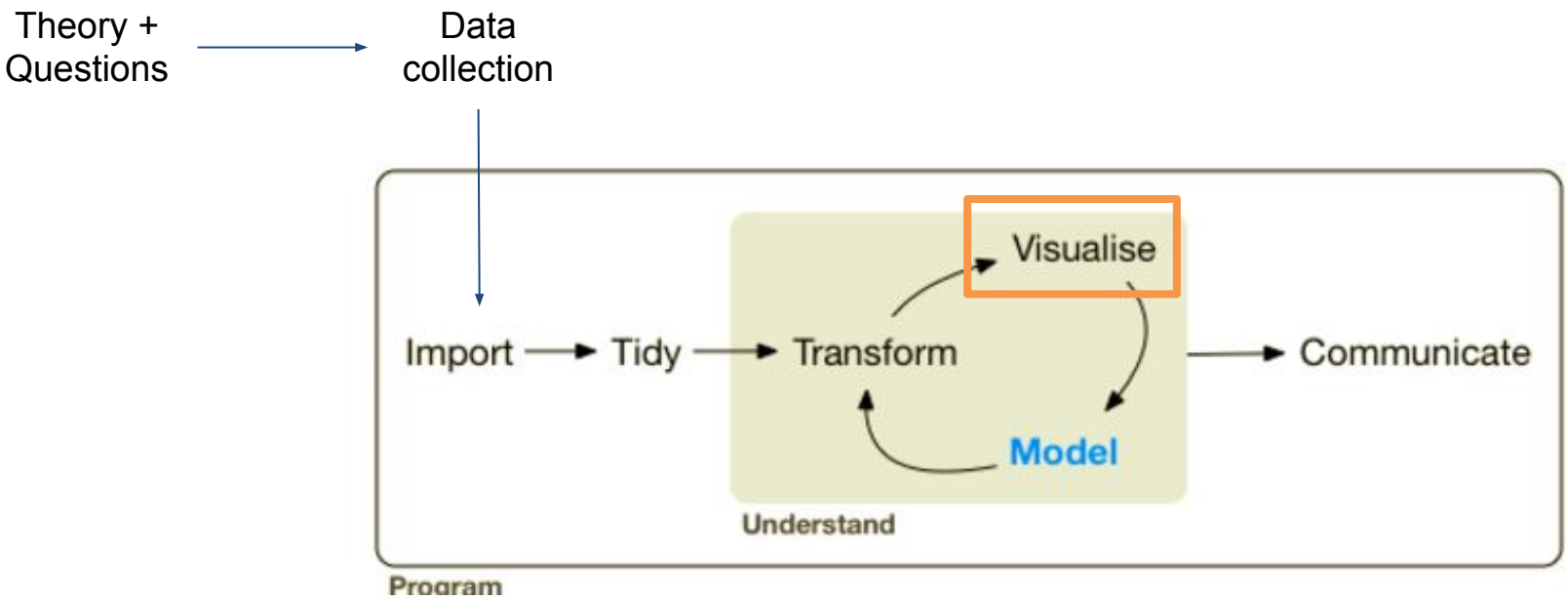
http://adolfoalvarez.cl/plumbers-chains-and-famous-painters-the-history-of-the-pipe-operator-in-r/

Group by & summarise

Reduces multiple values down to a single summary

```
marvel %>%
  group_by(SEX) %>%
  summarise(count = n(),
            avg_app = mean(APPEARANCES))
```

The data analysis workflow



https://rviews.rstudio.com/2017/06/08/what-is-the-tidyverse/

Basics of data visualization

One variable

Categorical
Barplot
Pie chart

Quantitative
Histogram

Two variables

Both Categorical
“Matrix” plot
Tiled barplot

One cat., one quant.
Boxplot
Tiled histogram

Both quantitative
Line plot (time series)
Scatterplot

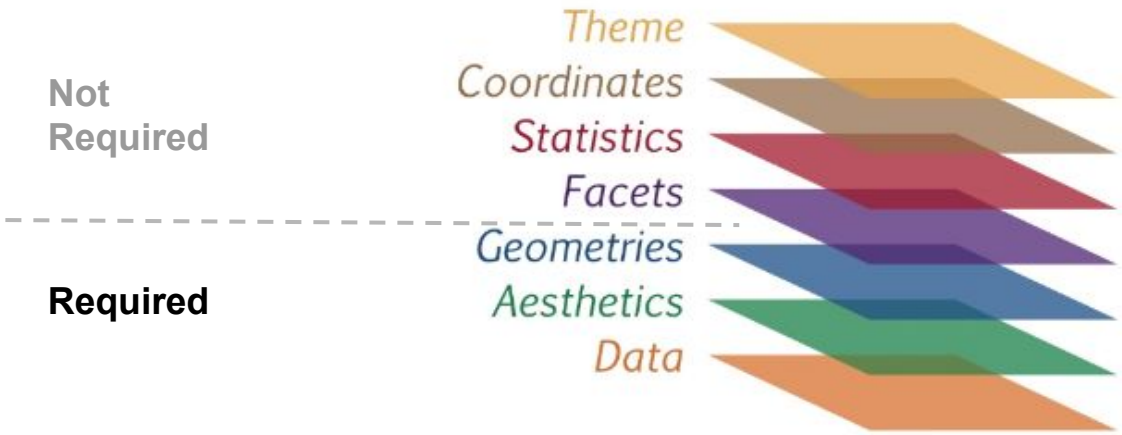
Three variables

Two quant., one cat.
Carpet plot
Grouped scatterplot
Grouped line plot (time series)

Three quantitative
Carpet plot (heatmap)

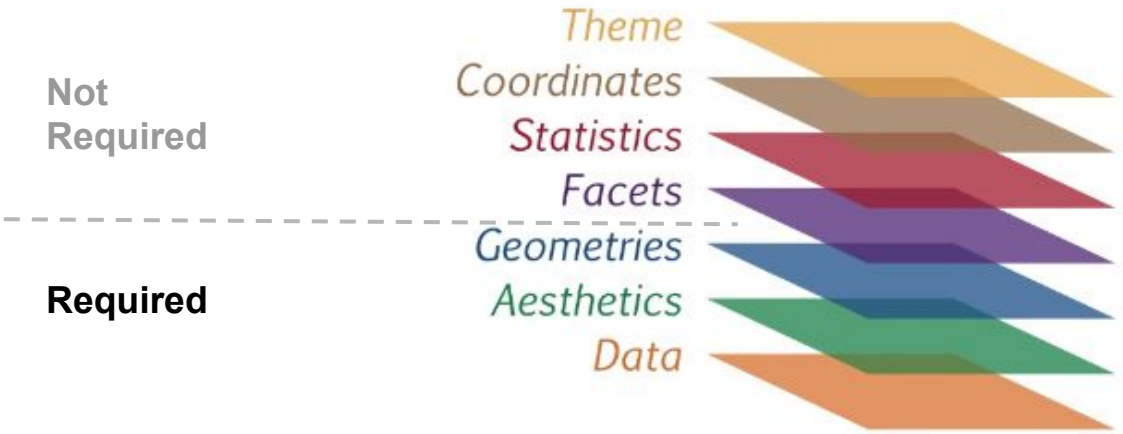
ggplot2

ggplot2 is a plotting system for R based on the grammar of graphics, that makes it easy to produce complex multi-layered graphics



ggplot2

```
ggplot(data = marvel,
    aes(x = active_years, y = appearances)) +
    geom_point() +
    facet_grid( . ~ sex) +
    theme_minimal()
```



ggplot2

Barplot (self count)

```
ggplot(data = data, aes(x = x)) +
   geom_bar()
```

Barplot (given values)

```
ggplot(data = data, aes(x = x)) +
   geom_bar(stat = "identity")
```

Pie chart

```
ggplot(data = data,
   aes(x = factor(1), y = count, fill = factor(z))) +
   geom_bar(width = 1, stat = "identity") +
   coord_polar(theta = "y")
```

ggplot2

Density plot

```
ggplot(data = data, aes(x = x, y = y)) +
   geom_density()
```

Scatterplot

```
ggplot(data = data, aes(x = x, y = y)) +
   geom_point()
```

Line plot

```
ggplot(data = data, aes(x = x, y = y)) +
   geom_line()
```

ggplot2

Add a group variable

```
ggplot(data = data, aes(x = x, y = y, color = a)) +
   geom_point()
```

Divide plot into subplots

```
ggplot(data = data, aes(x = x, y = y)) +
   geom_point() +
   facet_grid( . ~ a)
```

Mariachiara Fortuna

mariachiara.fortuna@quantide.com
mariachiara.fortuna1@gmail.com

www.milanor.net
https://www.facebook.com/MilanoRcommunity/
https://www.meetup.com/it-IT/R-Lab-Milano/

https://github.com/mariachiarafortuna/
@maryclary