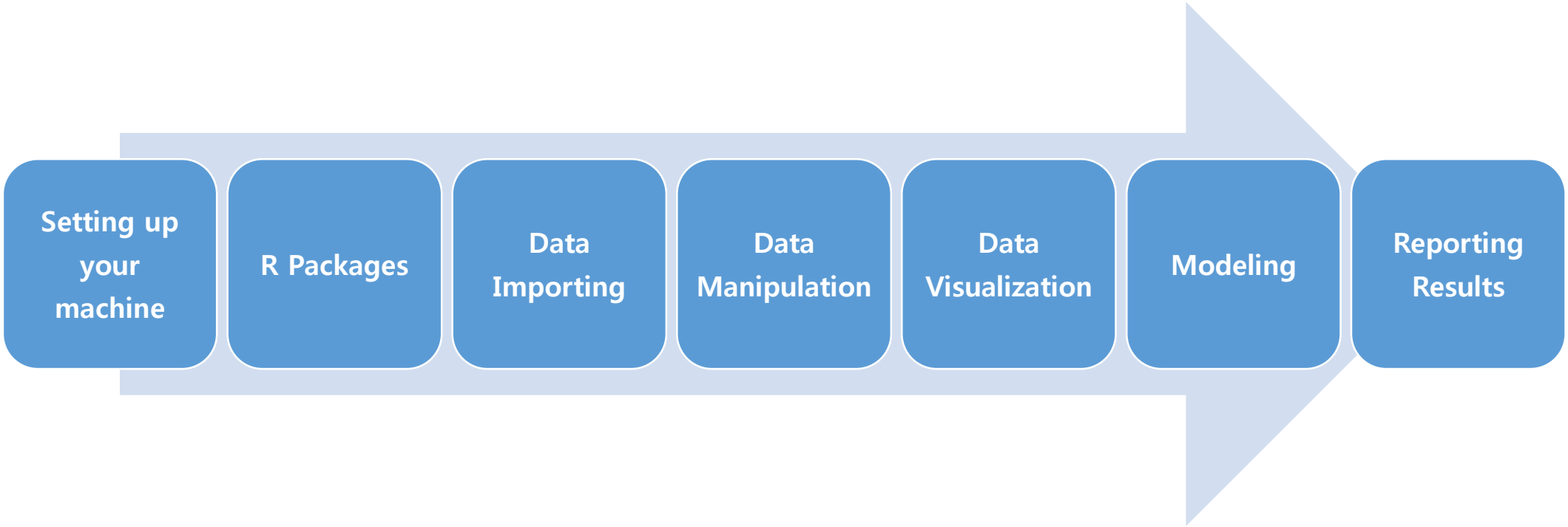
Introduction to R

Introduction to R

- 통계 계산과 그래픽을 위한 프로그래밍 언어이자 소프트웨어 환경
- R은 다양한 통계 기법과 수치 해석 기법을 지원함 (linear and nonlinear modelling, classical statistical tests, time-series analysis, classification, clustering, …)
- R은 사용자가 제작한 패키지를 추가하여 기능을 확장할 수 있음. 핵심적인 패키지는 R과 함께 설치되며, **CRAN**(the **C**omprehensive **R** **A**rchive **N**etwork)을 통해 2017년 현재 10,000개 이상의 패키지를 내려 받을 수 있음
- R의 또다른 강점은 그래픽 기능으로 수학 기호를 포함할 수 있는 출판물 수준의 그래프를 제공
- R은 윈도, 맥 OS 및 리눅스를 포함한 UNIX 플랫폼에서 이용 가능

Learning Path



1. Setting up: R & RStudio

RStudio

An IDE that was built just for R

- Syntax highlighting, code completion, and smart indentation
- Execute R code directly from the source editor
- Quickly jump to function definitions

Bring your workflow together

- Integrated R help and documentation
- Easily manage multiple working directories using projects
- Workspace browser and data viewer

Powerful authoring & Debugging

- Interactive debugger to diagnose and fix errors quickly
- Extensive package development tools
- Authoring with Sweave and R Markdown

2. R Packages

Bayesian	**Bayesian Inference**	Genetics	Statistical Genetics	Pharmacokinetics	Analysis of Pharmacokinetic Data
ChemPhys	Chemometrics and Computational Physics	**Graphics**	**Graphic Displays & Dynamic Graphics & Graphic Devices & Visualization**	Phylogenetics	Phylogenetics, Especially Comparative Methods
ClinicalTrials	Clinical Trial Design, Monitoring, and Analysis	**HighPerformanceComputing**	**High-Performance and Parallel Computing with R**	Psychometrics	Psychometric Models and Methods
Cluster	**Cluster Analysis & Finite Mixture Models**	**MachineLearning**	**Machine Learning & Statistical Learning**	ReproducibleResearch	Reproducible Research
DifferentialEquations	**Differential Equations**	MedicalImaging	Medical Image Analysis	Robust	Robust Statistical Methods
Distributions	**Probability Distributions**	MetaAnalysis	Meta-Analysis	SocialSciences	Statistics for the Social Sciences
Econometrics	**Econometrics**	**Multivariate**	**Multivariate Statistics**	Spatial	Analysis of Spatial Data
Environmetrics	Analysis of Ecological and Environmental Data	NaturalLanguageProcessing	Natural Language Processing	SpatioTemporal	Handling and Analyzing Spatio-Temporal Data
ExperimentalDesign	**Design of Experiments (DoE) & Analysis of Experimental Data**	**NumericalMathematics**	**Numerical Mathematics**	Survival	Survival Analysis
ExtremeValue	**Extreme Value Analysis**	OfficialStatistics	Official Statistics & Survey Methodology	**TimeSeries**	**Time Series Analysis**
Finance	**Empirical Finance**	**Optimization**	**Optimization and Mathematical Programming**	WebTechnologies	Web Technologies and Services
				gR	gRaphical Models in R

3. Data Importing

4. Data Manipulation

dplyr

tidyr

Data Wrangling with dplyr and tidyr Cheat Sheet

RStudio

Syntax - Helpful conventions for wrangling

dplyr::**tbl_df(iris)**
Converts data to tbl class. tbl's are easier to examine than data frames. R displays only the data that fits onscreen:

```
Source: local data frame [150 x 5]

   Sepal.Length Sepal.Width Petal.Length
1           5.1         3.5          1.4
2           4.9         3.0          1.4
3           4.7         3.2          1.3
4           4.6         3.1          1.5
5           5.0         3.6          1.4
..          ...         ...          ...
Variables not shown: Petal.Width (dbl),
  Species (fctr)
```

dplyr::**glimpse(iris)**
Information dense summary of tbl data.

utils::**View(iris)**
View data set in spreadsheet-like display (note capital V).

dplyr::**%>%**
Passes object on left hand side as first argument (or . argument) of function on righthand side.

`x %>% f(y)` *is the same as* `f(x, y)`
`y %>% f(x, ., z)` *is the same as* `f(x, y, z )`

"Piping" with %>% makes code more readable, e.g.

```
iris %>%
  group_by(Species) %>%
  summarise(avg = mean(Sepal.Width)) %>%
  arrange(avg)
```

Tidy Data - A foundation for wrangling in R

In a tidy data set:

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

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

Tidy data complements R's **vectorized operations**. R will automatically preserve observations as you manipulate variables. No other format works as intuitively with R.

M * A

Reshaping Data - Change the layout of a data set

tidyr::**gather(cases, "year", "n", 2:4)**
Gather columns into rows.

tidyr::**spread(pollution, size, amount)**
Spread rows into columns.

dplyr::**data_frame(a = 1:3, b = 4:6)**
Combine vectors into data frame (optimized).

dplyr::**arrange(mtcars, mpg)**
Order rows by values of a column (low to high).

dplyr::**arrange(mtcars, desc(mpg))**
Order rows by values of a column (high to low).

dplyr::**rename(tb, y = year)**
Rename the columns of a data frame.

tidyr::**separate(storms, date, c("y", "m", "d"))**
Separate one column into several.

tidyr::**unite(data, col, ..., sep)**
Unite several columns into one.

Subset Observations (Rows)

dplyr::**filter(iris, Sepal.Length > 7)**
Extract rows that meet logical criteria.

dplyr::**distinct(iris)**
Remove duplicate rows.

dplyr::**sample_frac(iris, 0.5, replace = TRUE)**
Randomly select fraction of rows.

dplyr::**sample_n(iris, 10, replace = TRUE)**
Randomly select n rows.

dplyr::**slice(iris, 10:15)**
Select rows by position.

dplyr::**top_n(storms, 2, date)**
Select and order top n entries (by group if grouped data).

Logic in R - ?Comparison, ?base::Logic

<	Less than	!=	Not equal to
>	Greater than	%in%	Group membership
==	Equal to	is.na	Is NA
<=	Less than or equal to	!is.na	Is not NA
>=	Greater than or equal to	&, |, !, xor, any, all	Boolean operators

Subset Variables (Columns)

dplyr::**select(iris, Sepal.Width, Petal.Length, Species)**
Select columns by name or helper function.

Helper functions for select - ?select

select(iris, contains("."))
Select columns whose name contains a character string.

select(iris, ends_with("Length"))
Select columns whose name ends with a character string.

select(iris, everything())
Select every column.

select(iris, matches(".t."))
Select columns whose name matches a regular expression.

select(iris, num_range("x", 1:5))
Select columns named x1, x2, x3, x4, x5.

select(iris, one_of(c("Species", "Genus")))
Select columns whose names are in a group of names.

select(iris, starts_with("Sepal"))
Select columns whose name starts with a character string.

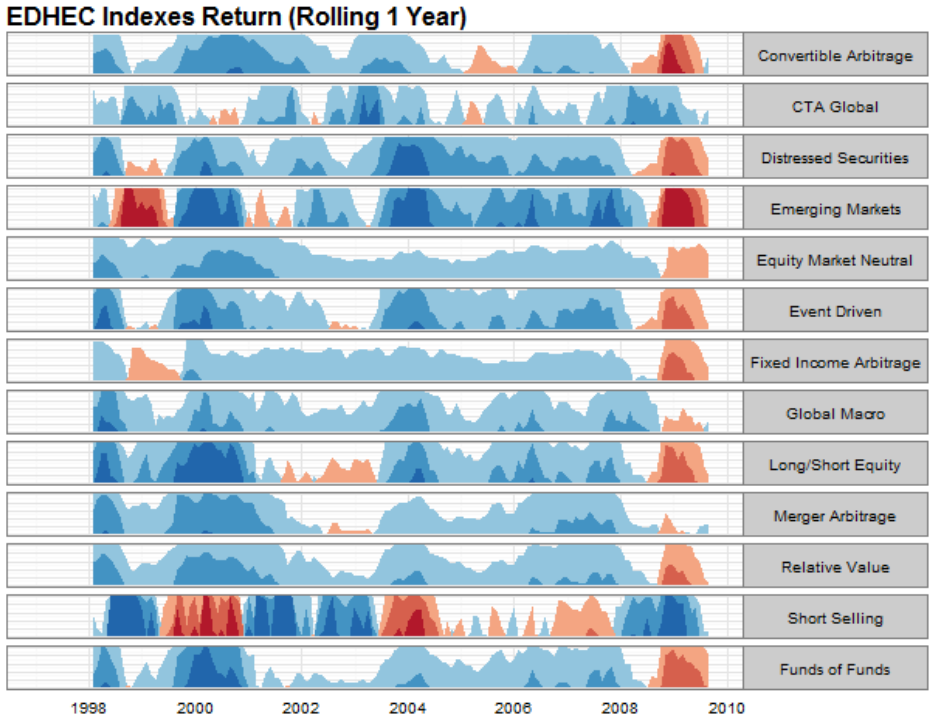
select(iris, Sepal.Length:Petal.Width)
Select all columns between Sepal.Length and Petal.Width (inclusive).

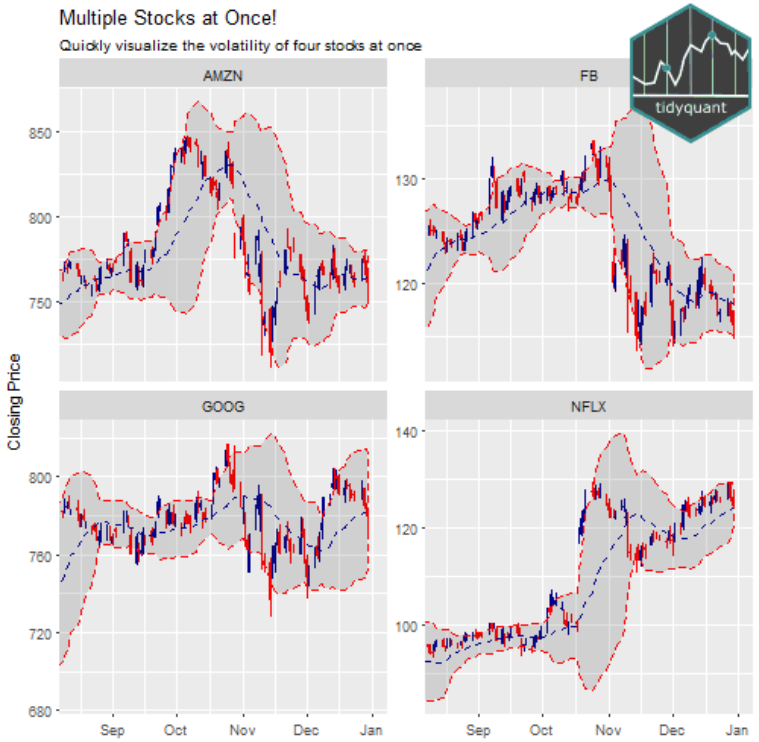
select(iris, -Species)
Select all columns except Species.

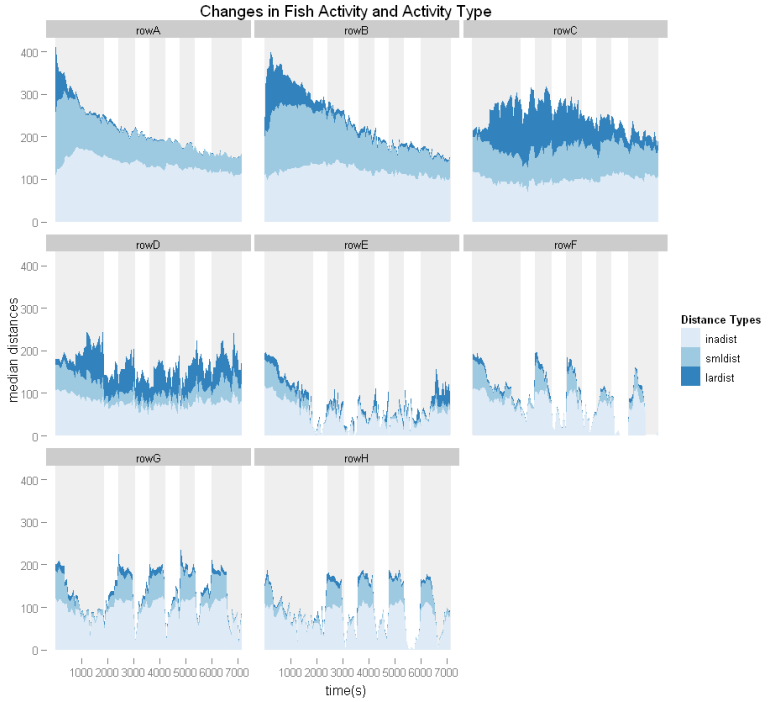
devtools::install_github("rstudio/EDAWR") for data sets

Learn more with browseVignettes(package = c("dplyr", "tidyr")) • dplyr 0.4.0 • tidyr 0.2.0 • Updated: 1/15

5. Data Visualization







7. Reporting Results

Get inspired
(gallery)

Get started
(tutorial)

Go deeper
(articles)

Getting help and learning more