Multivariate Analysis II

Alboukadel Kassambara

Practical Guide To Principal Component Methods in R

PCA, (M)CA, FAMD, MFA, HCPC, factoextra

sthda.com

Edition 1

Practical Guide to Principal Component Methods in R

Alboukadel KASSAMBARA

Published by STHDA (`http://www.sthda.com`), Alboukadel Kassambara

Contact: Alboukadel Kassambara <alboukadel.kassambara@gmail.com>

For general information contact Alboukadel Kassambara <alboukadel.kassambara@gmail.com>.

Contents

Preface

0.1 What you will learn

Large data sets containing multiple samples and variables are collected everyday by researchers in various fields, such as in Bio-medical, marketing, and geo-spatial fields.

Discovering knowledge from these data requires specific techniques for analyzing data sets containing multiple variables. **Multivariate analysis** (MVA) refers to a set of techniques used for analyzing a data set containing more than one variable.

Among these techniques, there are:

- Cluster analysis for identifying groups of observations with similar profile according to a specific criteria.
- Principal component methods, which consist of summarizing and visualizing the most important information contained in a multivariate data set.

Previously, we published a book entitled "Practical Guide To Cluster Analysis in R" (`https://goo.gl/DmJ5y5`). The aim of the current book is to provide a solid practical guidance to principal component methods in R. Additionally, we developed an R package named factoextra to create, easily, a ggplot2-based elegant plots of the results of principal component method. Factoextra official online documentation: `http://www.sthda.com/english/rpkgs/factoextra`

One of the difficulties inherent in multivariate analysis is the problem of visualizing data that has many variables. In R, there are many functions and packages for displaying a graph of the relationship between two variables (`http://www.sthda.com/english/wiki/data-visualization`). There are also commands for displaying different three-dimensional views. But when there are more than three variables, it is more difficult to visualize their relationships.

Fortunately, in data sets with many variables, some variables are often correlated. This can be explained by the fact that, more than one variable might be measuring the same driving principle governing the behavior of the system. Correlation indicates that there is redundancy in the data. When this happens, you can simplify the problem by replacing a group of correlated variables with a single new variable.

Principal component analysis is a rigorous statistical method used for achieving this simplification. The method creates a new set of variables, called principal components. Each principal component is a linear combination of the original variables. All the principal components are orthogonal to each other, so there is no redundant information.

The type of principal component methods to use depends on variable types contained in the data set. This practical guide will describe the following methods:

1. **Principal Component Analysis (PCA)**, which is one of the most popular multivariate analysis method. The goal of PCA is to summarize the information contained in a continuous (i.e, quantitative) multivariate data by reducing the dimensionality of the data without loosing important information.
2. **Correspondence Analysis (CA)**, which is an extension of the principal component analysis for analyzing a large contingency table formed by two *qualitative variables* (or categorical data).
3. **Multiple Correspondence Analysis (MCA)**, which is an adaptation of CA to a data table containing more than two categorical variables.
4. **Factor Analysis of Mixed Data (FAMD)**, dedicated to analyze a data set containing both quantitative and qualitative variables.
5. **Multiple Factor Analysis (MFA)**, dedicated to analyze data sets, in which variables are organized into groups (qualitative and/or quantitative variables).

Additionally, we'll discuss the **HCPC (Hierarchical Clustering on Principal Component)** method. It applies agglomerative hierarchical clustering on the results of principal component methods (PCA, CA, MCA, FAMD, MFA). It allows us, for example, to perform clustering analysis on any type of data (quantitative, qualitative or mixed data).

Figure 1 illustrates the type of analysis to be performed depending on the type of variables contained in the data set.

0.2 Key features of this book

Although there are several good books on principal component methods and related topics, we felt that many of them are either too theoretical or too advanced.

Our goal was to write a practical guide to multivariate analysis, visualization and interpretation, focusing on principal component methods.

The book presents the basic principles of the different methods and provide many examples in R. This book offers solid guidance in data mining for students and researchers.

Key features

- Covers principal component methods and implementation in R
- Short, self-contained chapters with tested examples that allow for flexibility in designing a course and for easy reference

At the end of each chapter, we present R lab sections in which we systematically work through applications of the various methods discussed in that chapter. Additionally, we provide links to other resources and to our hand-curated list of videos on principal component methods for further learning.

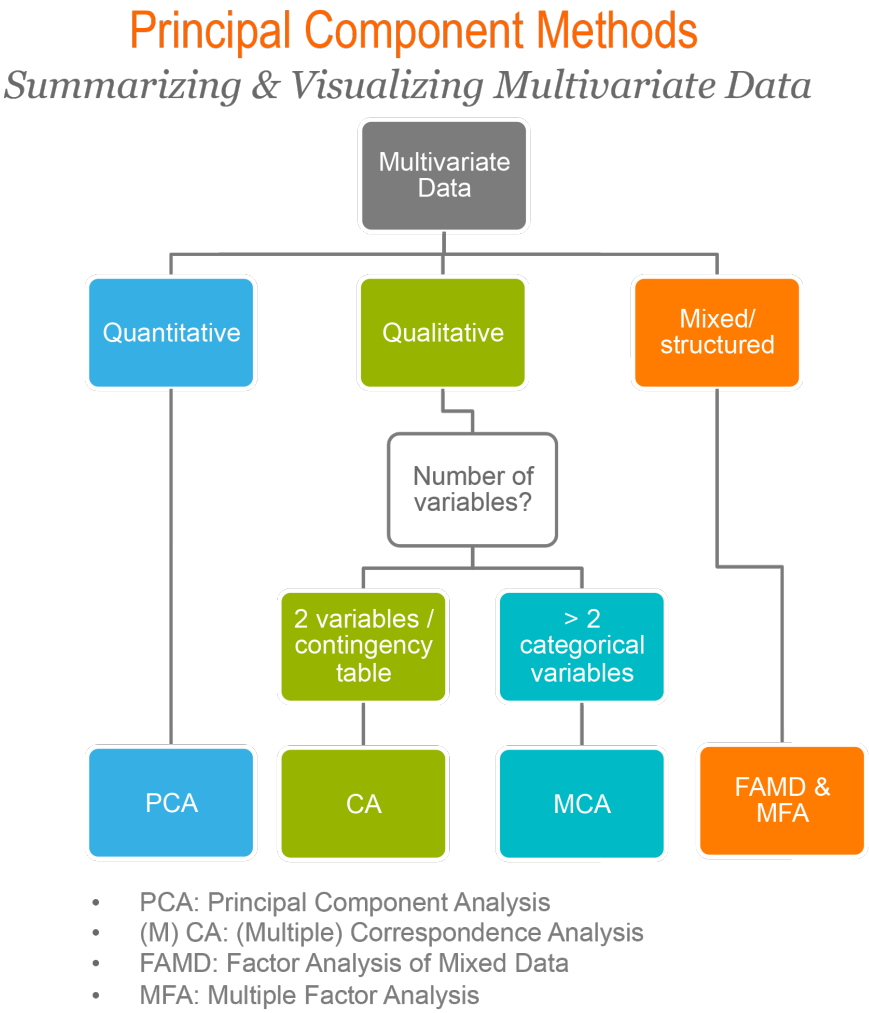


Figure 1: Principal component methods

0.3 How this book is organized

This book is divided into 4 parts and 6 chapters. Part I provides a quick introduction to R (chapter 1) and presents required R packages for the analysis and visualization (chapter 2).

In Part II, we describe classical multivariate analysis methods:

- Principal Component Analysis - PCA (chapter 3)
- Correspondence Analysis - CA (chapter 4)
- Multiple Correspondence Analysis - MCA (chapter 5)

In part III, we continue by discussing advanced methods for analyzing a data set containing a mix of variables (qualitative & quantitative) organized or not into groups:

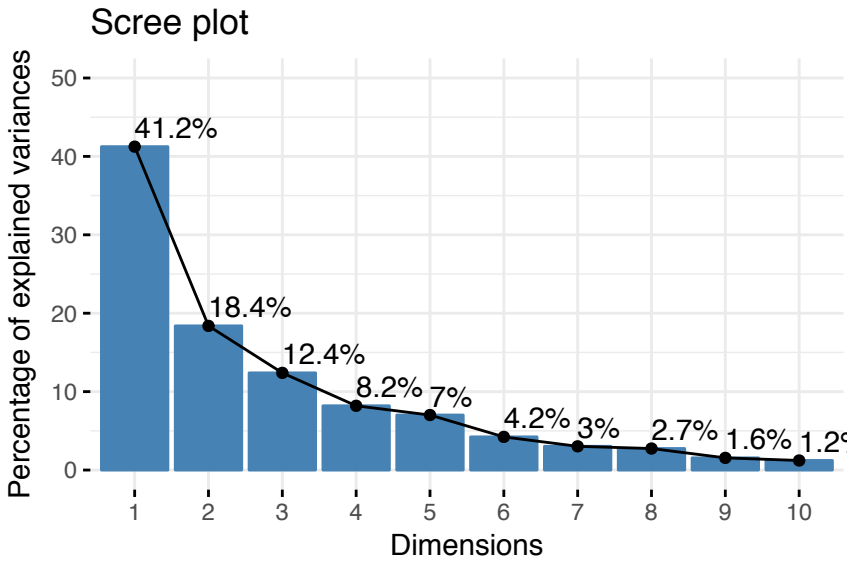
- Factor Analysis of Mixed Data - FAMD (chapter 6) and,
- Multiple Factor Analysis - MFA (chapter 7).

Finally, we show in Part IV, how to perform hierarchical clustering on principal components (HCPC) (chapter 8), which is useful for performing clustering with a data set

containing only qualitative variables or with a mixed data of qualitative and quantitative variables.

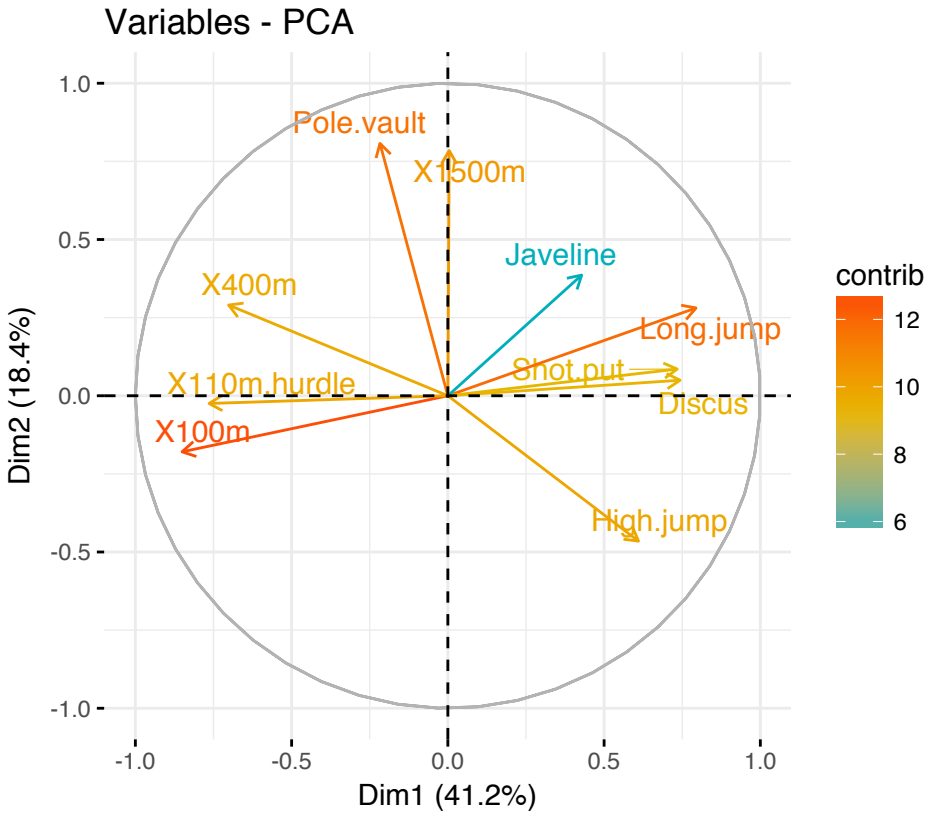
Some examples of plots generated in this book are shown hereafter. You'll learn how to create, customize and interpret these plots.

1) **Eigenvalues/variances of principal components**. Proportion of information retained by each principal component.

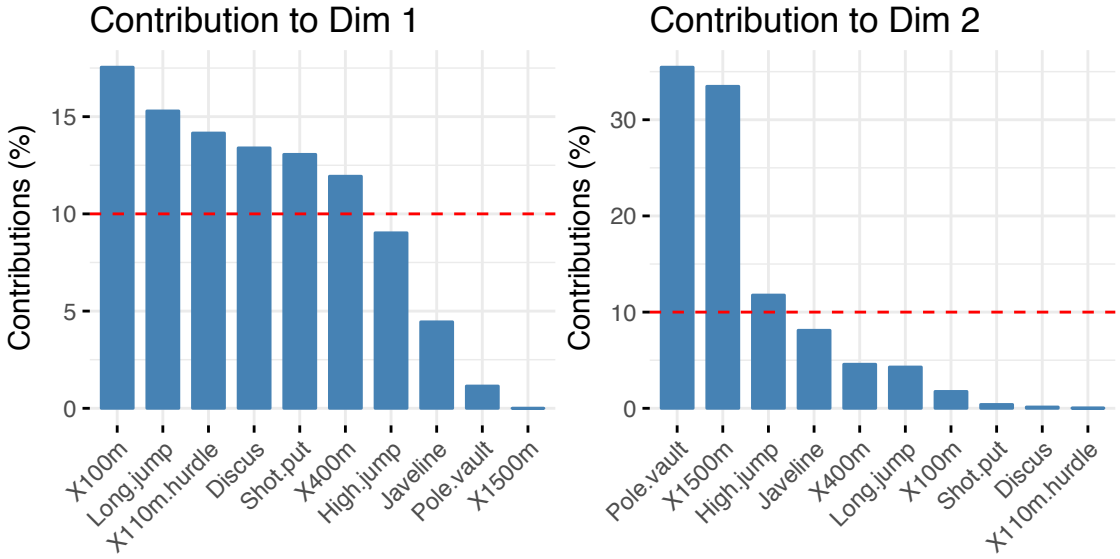


2) **PCA - Graph of variables**:

- Control variable colors using their contributions to the principal components.

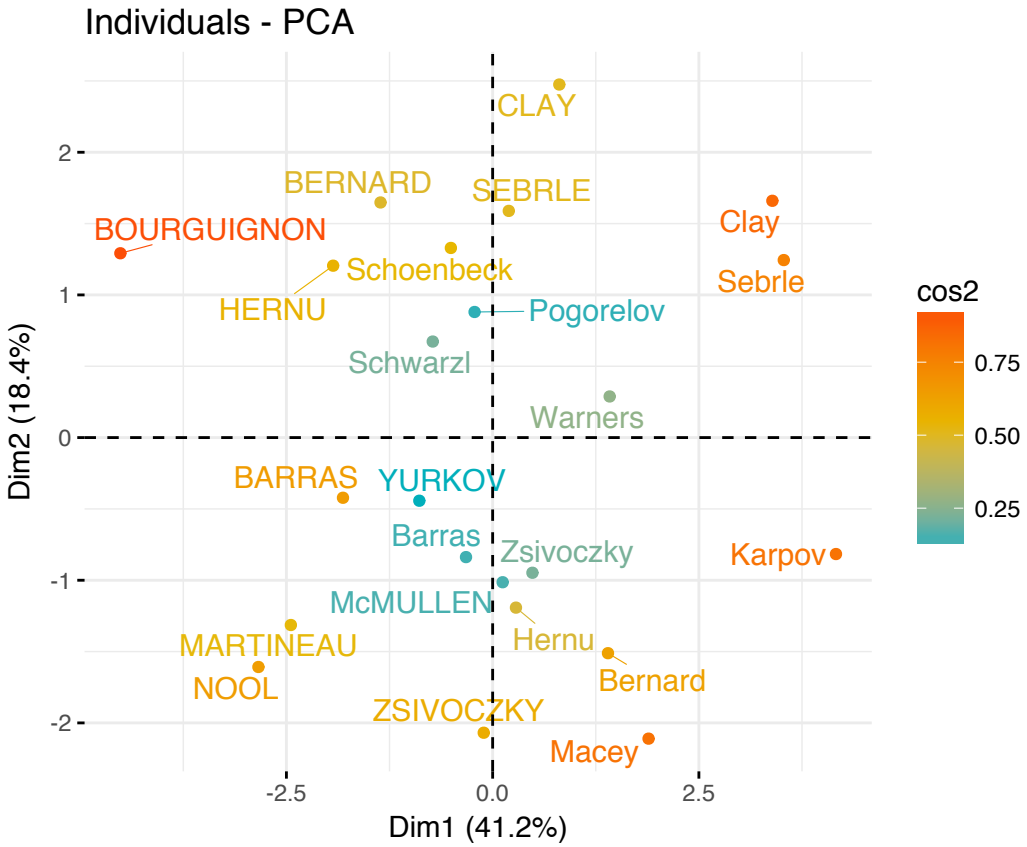


- Highlight the most contributing variables to each principal dimension:

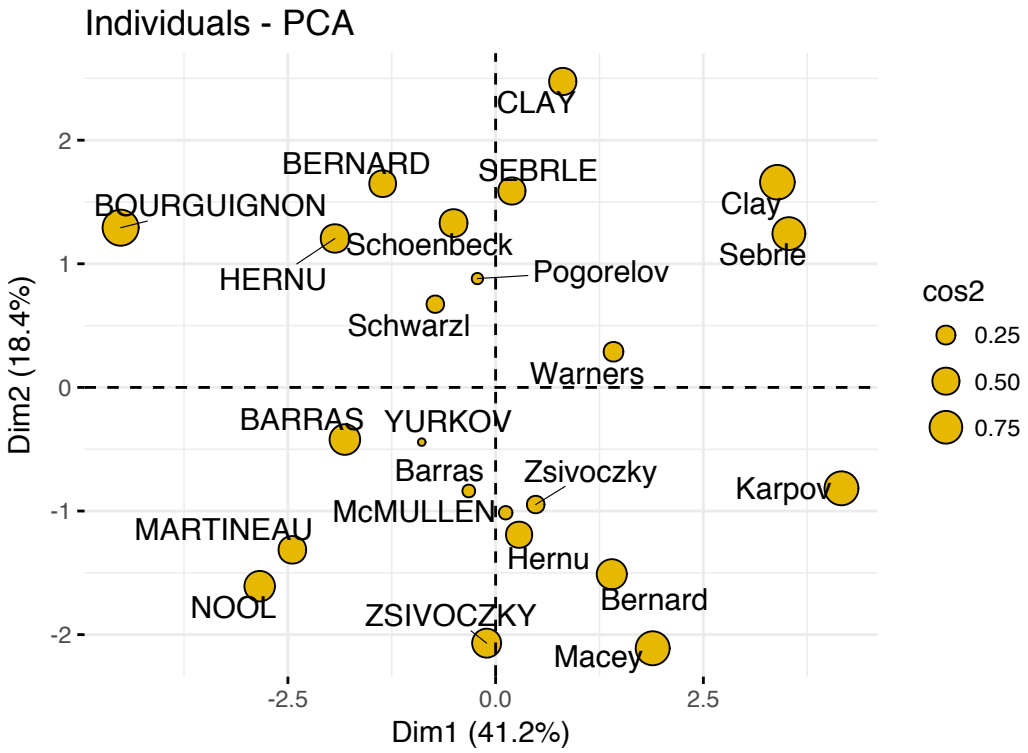


3) **PCA - Graph of individuals**:

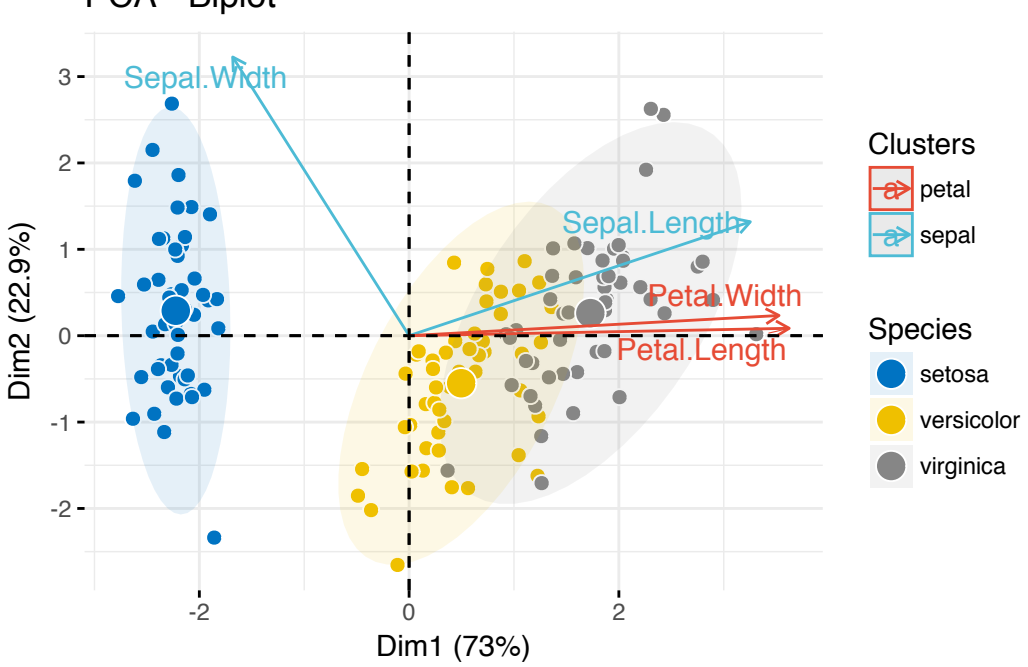
- Control automatically the color of individuals using the cos2 (the quality of the individuals on the factor map)



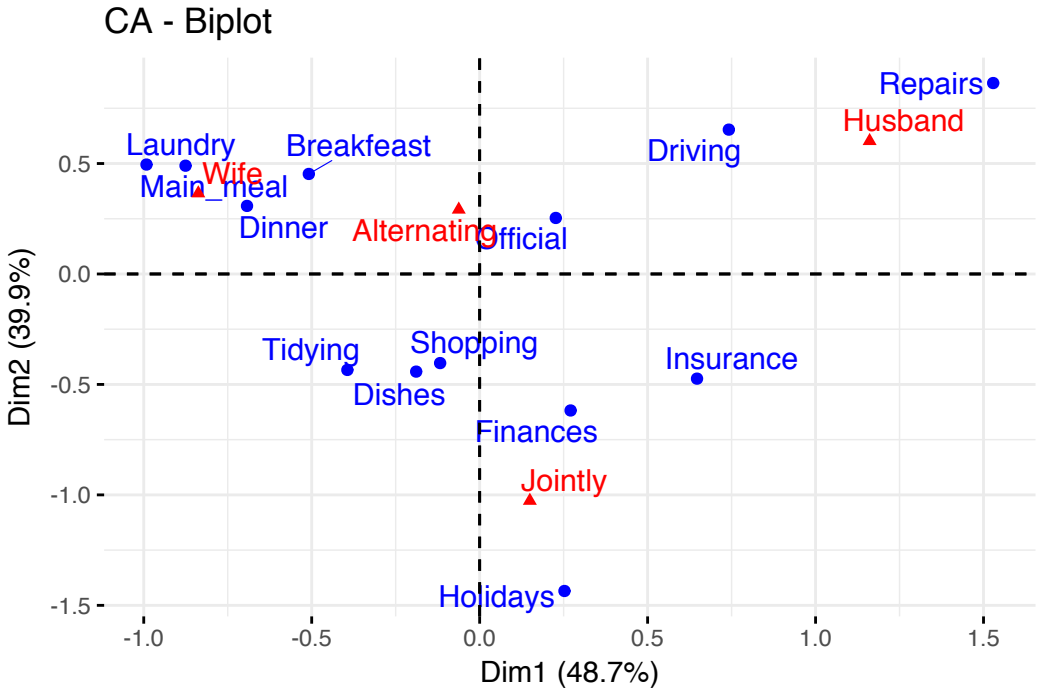
- Change the point size according to the cos2 of the corresponding individuals:



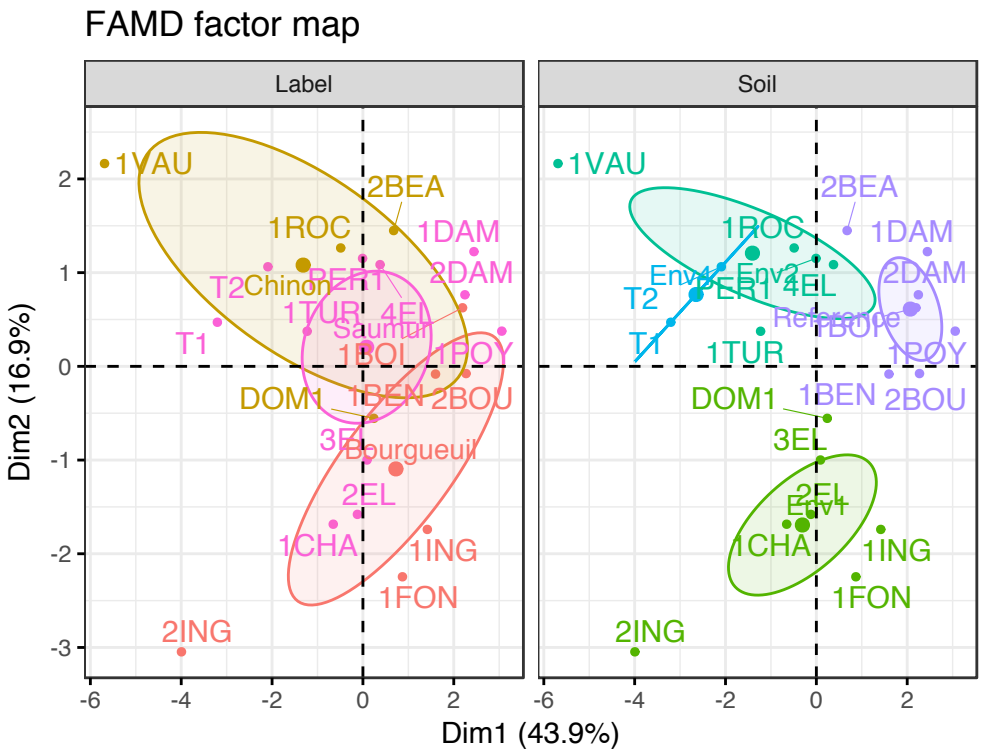
4) **PCA - Biplot of individuals and variables**



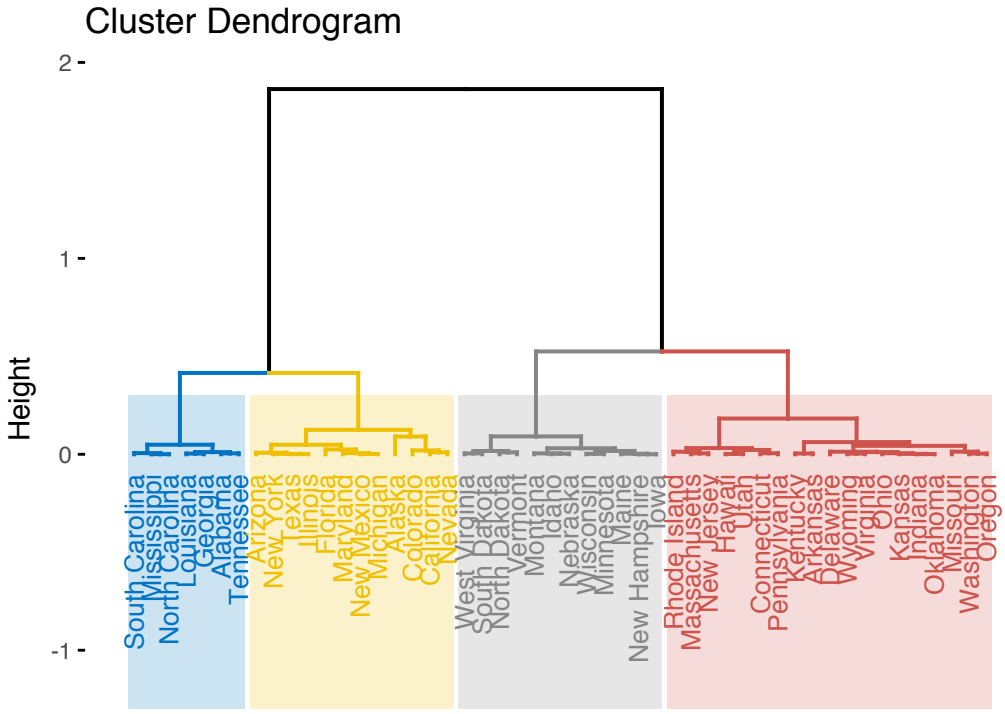
5) **Correspondence analysis**. Association between categorical variables.



6) **FAMD** - Analyzing mixed data



7) **Clustering on principal components**



0.4 Book website

The website for this book is located at : `http://www.sthda.com/english/`. It contains number of resources.

0.5 Executing the R codes from the PDF

For a single line R code, you can just copy the code from the PDF to the R console.

For a multiple-line R codes, an error is generated, sometimes, when you copy and paste directly the R code from the PDF to the R console. If this happens, a solution is to:

- Paste firstly the code in your R code editor or in your text editor
- Copy the code from your text/code editor to the R console

0.6 Acknowledgment

I sincerely thank all developers for their efforts behind the packages that factoextra depends on, namely, ggplot2 (Hadley Wickham, Springer-Verlag New York, 2009), FactoMineR (Sebastien Le et al., Journal of Statistical Software, 2008), dendextend (Tal Galili, Bioinformatics, 2015), cluster (Martin Maechler et al., 2016) and more.

0.7 Colophon

This book was built with:

- R 3.3.2
- factoextra 1.0.5
- FactoMineR 1.36
- ggpubr 0.1.5
- dplyr 0.7.2
- bookdown 0.4.3

About the author

Alboukadel Kassambara is a PhD in Bioinformatics and Cancer Biology. He works since many years on genomic data analysis and visualization (read more: http://www.alboukadel.com/).

He has work experiences in statistical and computational methods to identify prognostic and predictive biomarker signatures through integrative analysis of large-scale genomic and clinical data sets.

He created a bioinformatics web-tool named GenomicScape (www.genomicscape.com) which is an easy-to-use web tool for gene expression data analysis and visualization.

He developed also a training website on data science, named STHDA (Statistical Tools for High-throughput Data Analysis, www.sthda.com/english), which contains many tutorials on data analysis and visualization using R software and packages.

He is the author of many popular R packages for:

- multivariate data analysis (**factoextra**, http://www.sthda.com/english/rpkgs/factoextra),
- survival analysis (**survminer**, http://www.sthda.com/english/rpkgs/survminer/),
- correlation analysis (**ggcorrplot**, http://www.sthda.com/english/wiki/ggcorrplot-visualization-of-a-correlation-matrix-using-ggplot2),
- creating publication ready plots in R (**ggpubr**, http://www.sthda.com/english/rpkgs/ggpubr).

Recently, he published three books on data analysis and visualization:

1. Practical Guide to Cluster Analysis in R (https://goo.gl/DmJ5y5)
2. Guide to Create Beautiful Graphics in R (https://goo.gl/vJ0OYb).
3. Complete Guide to 3D Plots in R (https://goo.gl/v5gwl0).

Part I

Basics

Chapter 1

Introduction to R

R is a free and powerful statistical software for **analyzing** and **visualizing** data. If you want to learn easily the essential of R programming, visit our series of tutorials available on STHDA: `http://www.sthda.com/english/wiki/r-basics-quick-and-easy`.

In this chapter, we provide a very brief introduction to **R**, for installing R/RStudio as well as importing your data into R for computing principal component methods.

1.1 Installing R and RStudio

R and RStudio can be installed on Windows, MAC OSX and Linux platforms. RStudio is an integrated development environment for R that makes using R easier. It includes a console, code editor and tools for plotting.

1. R can be downloaded and installed from the Comprehensive R Archive Network (CRAN) webpage (`http://cran.r-project.org/`)
2. After installing R software, install also the RStudio software available at: `http://www.rstudio.com/products/RStudio/`.
3. Launch RStudio and start use R inside R studio.

1.2 Installing and loading R packages

An **R package** is an extension of R containing data sets and specific R functions to solve specific questions.

For example, in this book, you'll learn how to compute and visualize principal component methods using **FactoMineR** and **factoextra** R packages.

There are thousands other R packages available for download and installation from CRAN[1], Bioconductor[2] (biology related R packages) and GitHub[3] repositories.

[1] `https://cran.r-project.org/`
[2] `https://www.bioconductor.org/`
[3] `https://github.com/`

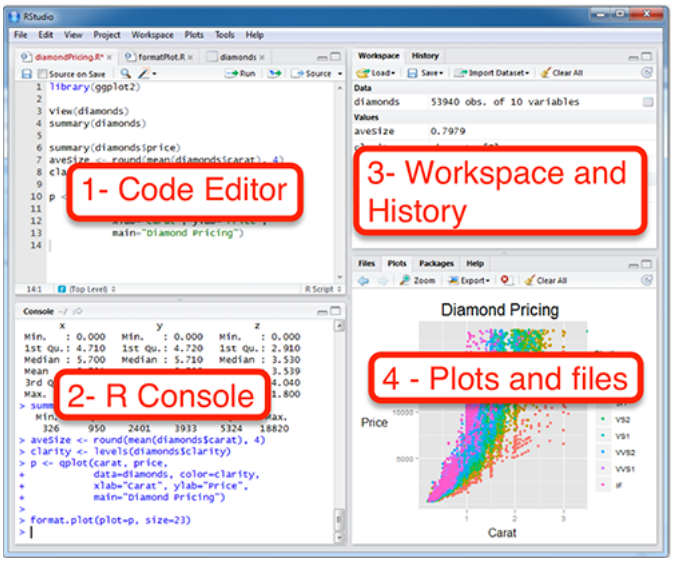


Figure 1.1: Rstudio interface

1. How to install packages from CRAN? Use the function *install.packages*():

```
install.packages("FactoMineR")
install.packages("factoextra")
```

2. How to install packages from GitHub? You should first install *devtools* if you don't have it already installed on your computer:

For example, the following R code installs the latest developmental version of *factoextra* R package developed by A. Kassambara (`https://github.com/kassambara/facoextra`) for multivariate data analysis and elegant visualization.

```
install.packages("devtools")
devtools::install_github("kassambara/factoextra")
```

Note that, GitHub contains the latest developmental version of R packages.

3. After installation, you must first load the package for using the functions in the package. The function *library*() is used for this task.

```
library("FactoMineR")
library("factoextra")
```

Now, we can use R functions, such as *PCA*() [in the FactoMineR package] for performing principal component analysis.

1.3 Getting help with functions in R

If you want to learn more about a given function, say PCA(), type this in R console:

```
?PCA
```

1.4 Importing your data into R

1. **Prepare your file** as follow:

- Use the first row as **column names**. Generally, columns represent **variables**
- Use the first column as **row names**. Generally rows represent **observations** or **individuals**.
- Each row/column name should be unique, so remove duplicated names.
- Avoid names with blank spaces. Good column names: *Long_jump* or *Long.jump*. Bad column name: *Long jump*.
- Avoid names with special symbols: ?, $, *, +, #, (,), -, /, }, {, |, >, < etc. Only underscore can be used.
- Avoid beginning variable names with a number. Use letter instead. Good column names: sport_100m or x100m. Bad column name: 100m
- R is case sensitive. This means that Name is different from Name or NAME.
- Avoid blank rows in your data.
- Delete any comments in your file.
- Replace missing values by **NA** (for not available)
- If you have a column containing date, use the four digit format. Good format: 01/01/2016. Bad format: 01/01/16

2. The **final file** should look like this:

name	x100m	Long.jump	Shot.put	High.jump
SEBRLE	11.04	7.58	14.83	2.07
CLAY	10.76	7.4	14.26	1.86
BERNARD	11.02	7.23	14.25	1.92
YURKOV	11.34	7.09	15.19	2.1
ZSIVOCZKY	11.13	7.3	NA	2.01
McMULLEN	10.83	7.31	13.76	2.13
MARTINEAU	NA	6.81	14.57	1.95
HERNU	NA	7.56	14.41	1.86
BARRAS	11.33	6.97	14.09	1.95
NOOL	11.33	7.27	12.68	1.98
BOURGUIGNON	11.36	6.8	13.46	1.86

Figure 1.2: General data format for importation into R

3. **Save your file**

We recommend to save your file into **.txt** (tab-delimited text file) or **.csv** (comma separated value file) format.

4. **Get your data into R**:

Use the R code below. You will be asked to choose a file:

```
# .txt file: Read tab separated values
my_data <- read.delim(file.choose(), row.names = 1)
```

```
# .csv file: Read comma (",") separated values
my_data <- read.csv(file.choose(), row.names = 1)

# .csv file: Read semicolon (";") separated values
my_data <- read.csv2(file.choose(), row.names = 1)
```

Using these functions, the imported data will be of class **data.frame** (R terminology).

You can read more about how to import data into R at this link: `http://www.sthda.com/english/wiki/importing-data-into-r`

1.5 Demo data sets

R comes with several *built-in data sets*, which are generally used as demo data for playing with R functions. The most used R demo data sets include: **USArrests**, **iris** and **mtcars**. To load a demo data set, use the function **data**() as follow:

```
data("USArrests")   # Loading
head(USArrests, 3)  # Print the first 3 rows
```

```
##          Murder Assault UrbanPop Rape
## Alabama    13.2     236       58 21.2
## Alaska     10.0     263       48 44.5
## Arizona     8.1     294       80 31.0
```

If you want learn more about USArrests data sets, type this:

```
?USArrests
```

To select just certain columns from a data frame, you can either refer to the columns by name or by their location (i.e., column 1, 2, 3, etc.).

```
# Access the data in 'Murder' column
# dollar sign is used
head(USArrests$Murder)
```

```
## [1] 13.2 10.0  8.1  8.8  9.0  7.9
```

```
# Or use this
USArrests[, 'Murder']
# Or use this
USArrests[, 1] # column number 1
```

1.6 Close your R/RStudio session

Each time you close R/RStudio, you will be asked whether you want to save the data from your R session. If you decide to save, the data will be available in future R sessions.

Chapter 2

Required R packages

2.1 FactoMineR & factoextra

There are a number of R packages implementing principal component methods. These packages include: *FactoMineR*, *ade4*, *stats*, *ca*, *MASS* and *ExPosition*.

However, the result is presented differently depending on the used package.

To help in the interpretation and in the visualization of multivariate analysis - such as cluster analysis and principal component methods - we developed an easy-to-use R package named **factoextra** (official online documentation: `http://www.sthda.com/english/rpkgs/factoextra`)(Kassambara and Mundt, 2017).

> No matter which package you decide to use for computing principal component methods, the factoextra R package can help to extract easily, in a human readable data format, the analysis results from the different packages mentioned above. factoextra provides also convenient solutions to create ggplot2-based beautiful graphs.

In this book, we'll use mainly:

- the **FactoMineR** package (Husson et al., 2017a) to compute principal component methods;
- and the **factoextra** package (Kassambara and Mundt, 2017) for extracting, visualizing and interpreting the results.

The other packages - ade4, ExPosition, etc - will be presented briefly.

The Figure 2.1 illustrates the key functionality of FactoMineR and factoextra.

Methods, which outputs can be visualized using the factoextra package are shown on the Figure 2.2:

2.2 Installation

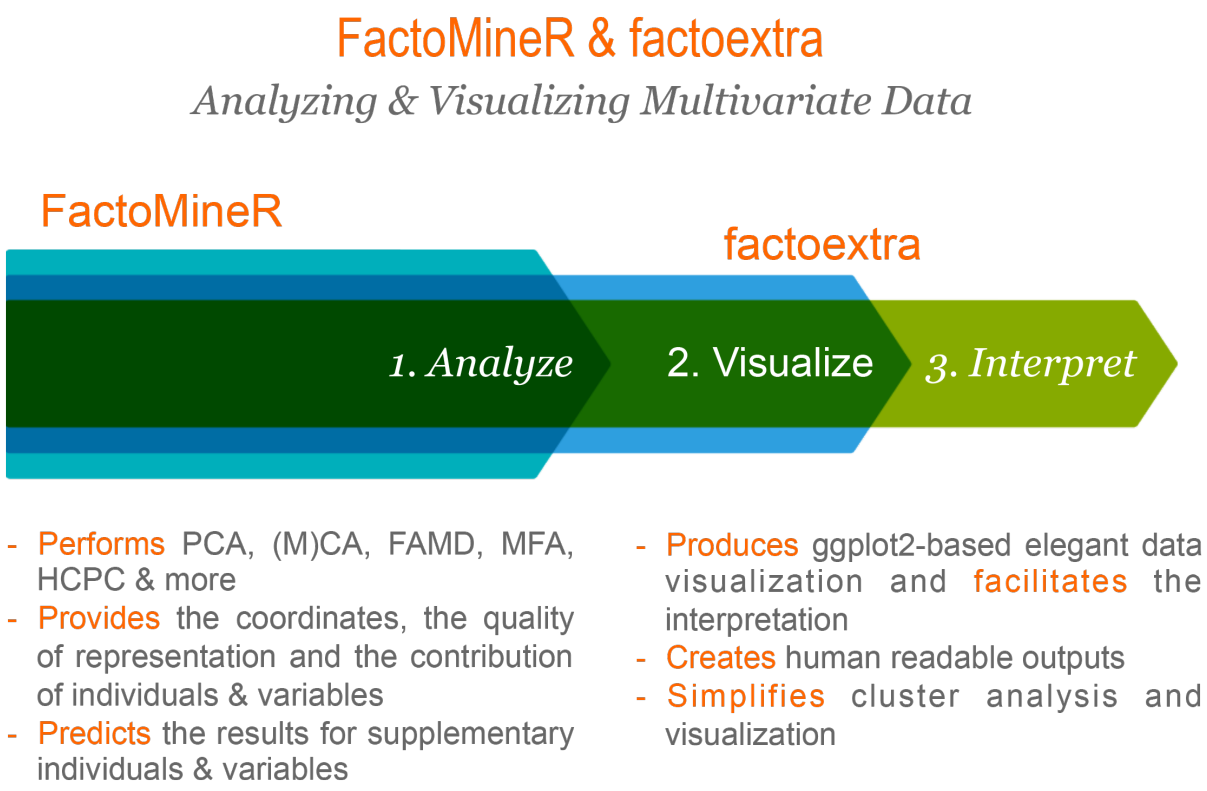


Figure 2.1: Key features of FactoMineR and factoextra for multivariate analysis

2.2.1 Installing FactoMineR

The FactoMineR package can be installed and loaded as follow:

```
# Install
install.packages("FactoMineR")

# Load
library("FactoMineR")
```

2.2.2 Installing factoextra

- factoextra can be installed from CRAN[1] as follow:

```
install.packages("factoextra")
```

- Or, install the latest developmental version from Github[2]

```
if(!require(devtools)) install.packages("devtools")
devtools::install_github("kassambara/factoextra")
```

- Load factoextra as follow :

[1] https://cran.r-project.org/package=factoextra
[2] https://github.com/kassambara/factoextra

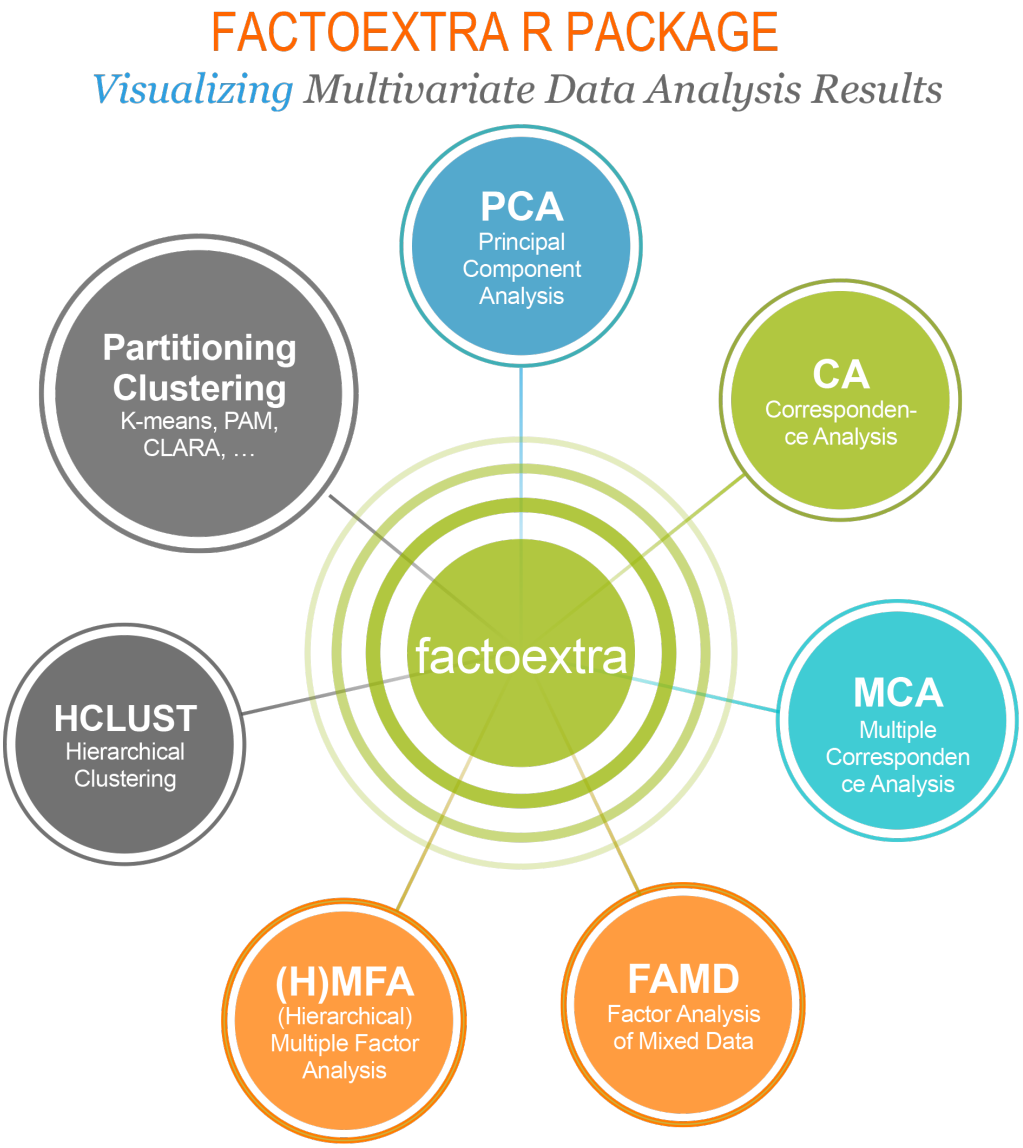


Figure 2.2: Principal component methods and clustering methods supported by the factoextra R package

```
library("factoextra")
```

2.3 Main R functions

2.3.1 Main functions in FactoMineR

Functions for computing principal component methods and clustering:

Functions	Description
PCA	Principal component analysis.
CA	Correspondence analysis.

Functions	Description
MCA	Multiple correspondence analysis.
FAMD	Factor analysis of mixed data.
MFA	Multiple factor analysis.
HCPC	Hierarchical clustering on principal components.
dimdesc	Dimension description.

2.3.2 Main functions in factoextra

factoextra functions covered in this book are listed in the table below. See the online documentation (`http://www.sthda.com/english/rpkgs/factoextra`) for a complete list.

- **Visualizing principal component method outputs**

Functions	Description
fviz_eig (or fviz_eigenvalue)	Visualize eigenvalues.
fviz_pca	Graph of PCA results.
fviz_ca	Graph of CA results.
fviz_mca	Graph of MCA results.
fviz_mfa	Graph of MFA results.
fviz_famd	Graph of FAMD results.
fviz_hmfa	Graph of HMFA results.
fviz_ellipses	Plot ellipses around groups.
fviz_cos2	Visualize element cos2. [3]
fviz_contrib	Visualize element contributions. [4]

- **Extracting data from principal component method outputs**. The following functions extract all the results (coordinates, squared cosine, contributions) for the active individuals/variables from the analysis outputs.

Functions	Description
get_eigenvalue	Access to the dimension eigenvalues.
get_pca	Access to PCA outputs.
get_ca	Access to CA outputs.
get_mca	Access to MCA outputs.
get_mfa	Access to MFA outputs.
get_famd	Access to MFA outputs.
get_hmfa	Access to HMFA outputs.
facto_summarize	Summarize the analysis.

- **Clustering analysis and visualization**

[3] Cos2: quality of representation of the row/column variables on the principal component maps.
[4] This is the contribution of row/column elements to the definition of the principal components.

Functions	Description
fviz_dend	Enhanced Visualization of Dendrogram.
fviz_cluster	Visualize Clustering Results.

Part II

Classical Methods

Chapter 3

Principal Component Analysis

3.1 Introduction

Principal component analysis (**PCA**) allows us to summarize and to visualize the information in a data set containing individuals/observations described by multiple inter-correlated quantitative variables. Each variable could be considered as a different dimension. If you have more than 3 variables in your data sets, it could be very difficult to visualize a multi-dimensional hyperspace.

Principal component analysis is used to extract the important information from a multivariate data table and to express this information as a set of few new variables called **principal components**. These new variables correspond to a linear combination of the originals. The number of principal components is less than or equal to the number of original variables.

The information in a given data set corresponds to the *total variation* it contains. The goal of PCA is to identify directions (or principal components) along which the variation in the data is maximal.

In other words, PCA reduces the dimensionality of a multivariate data to two or three principal components, that can be visualized graphically, with minimal loss of information.

In this chapter, we describe the basic idea of PCA and, demonstrate how to compute and visualize PCA using R software. Additionally, we'll show how to reveal the most important variables that explain the variations in a data set.

3.2 Basics

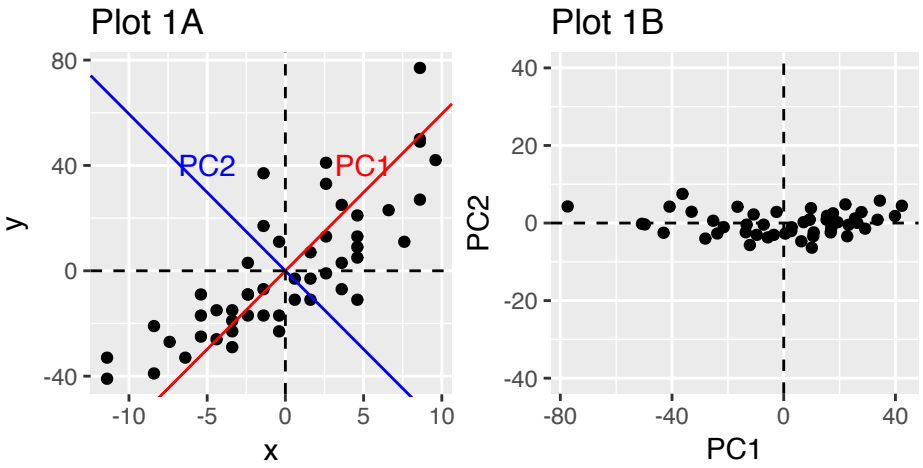
Understanding the details of PCA requires knowledge of linear algebra. Here, we'll explain only the basics with simple graphical representation of the data.

In the Plot 1A below, the data are represented in the X-Y coordinate system. The dimension reduction is achieved by identifying the principal directions, called principal components, in which the data varies.

PCA assumes that the directions with the largest variances are the most "important" (i.e, the most principal).

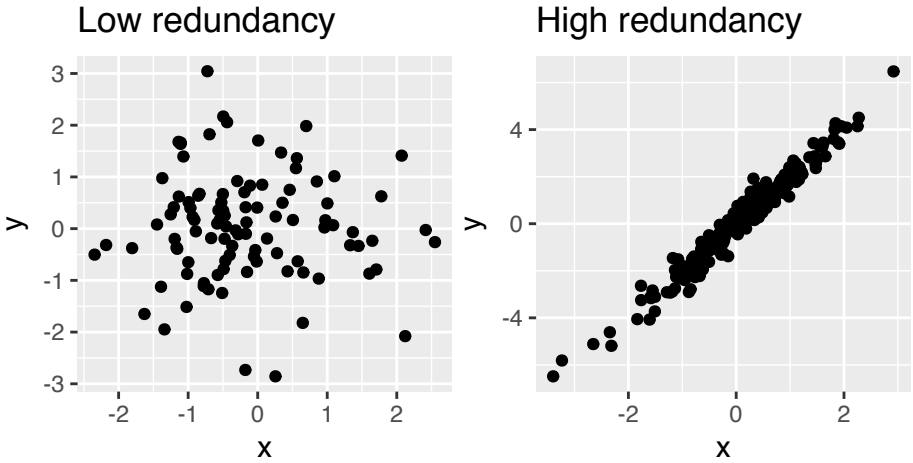
In the figure below, the *PC1 axis* is the **first principal direction** along which the samples show the largest variation. The *PC2 axis* is the **second most important direction** and it is **orthogonal** to the PC1 axis.

The dimensionality of our two-dimensional data can be reduced to a single dimension by projecting each sample onto the first principal component (Plot 1B)



Technically speaking, the amount of variance retained by each principal component is measured by the so-called **eigenvalue**.

Note that, the PCA method is particularly useful when the variables within the data set are highly correlated. Correlation indicates that there is redundancy in the data. Due to this redundancy, PCA can be used to reduce the original variables into a smaller number of new variables (= **principal components**) explaining most of the variance in the original variables.



Taken together, the main purpose of principal component analysis is to:

- identify hidden pattern in a data set,
- reduce the dimensionnality of the data by **removing the noise** and **redundancy** in the data,
- identify correlated variables

3.3 Computation

3.3.1 R packages

Several functions from different packages are available in the *R software* for computing PCA:

- *prcomp*() and *princomp*() [built-in R *stats* package],
- *PCA*() [*FactoMineR* package],
- *dudi.pca*() [*ade4* package],
- and *epPCA*() [*ExPosition* package]

No matter what function you decide to use, you can easily extract and visualize the results of PCA using R functions provided in the *factoextra* R package.

Here, we'll use the two packages FactoMineR (for the analysis) and factoextra (for ggplot2-based visualization).

Install the two packages as follow:

```
install.packages(c("FactoMineR", "factoextra"))
```

Load them in R, by typing this:

```
library("FactoMineR")
library("factoextra")
```

3.3.2 Data format

We'll use the demo data sets *decathlon2* from the *factoextra* package:

```
data(decathlon2)
# head(decathlon2)
```

As illustrated in Figure 3.1, the data used here describes athletes' performance during two sporting events (Desctar and OlympicG). It contains 27 individuals (athletes) described by 13 variables.

Note that, only some of these individuals and variables will be used to perform the principal component analysis. The coordinates of the remaining individuals and variables on the factor map will be predicted after the PCA.

In PCA terminology, our data contains :

- *Active individuals* (in light blue, rows 1:23) : Individuals that are used during the principal component analysis.
- *Supplementary individuals* (in dark blue, rows 24:27) : The coordinates of these individuals will be predicted using the PCA information and parameters obtained with active individuals/variables