

**MBA
USP
ESALQ**

Database Structures, Types of Variables and Measurement Scales

Introduction to Machine Learning

Rafael de Freitas Souza



Data Science

Set of programming techniques focused on collection, treatment, manipulation, organization, analysis, extraction of information and data presentation, in the form of reports or graphics, in order to support the decision-making process.

Data Science techniques to be approached during the Module:

Introduction to *data wrangling*;

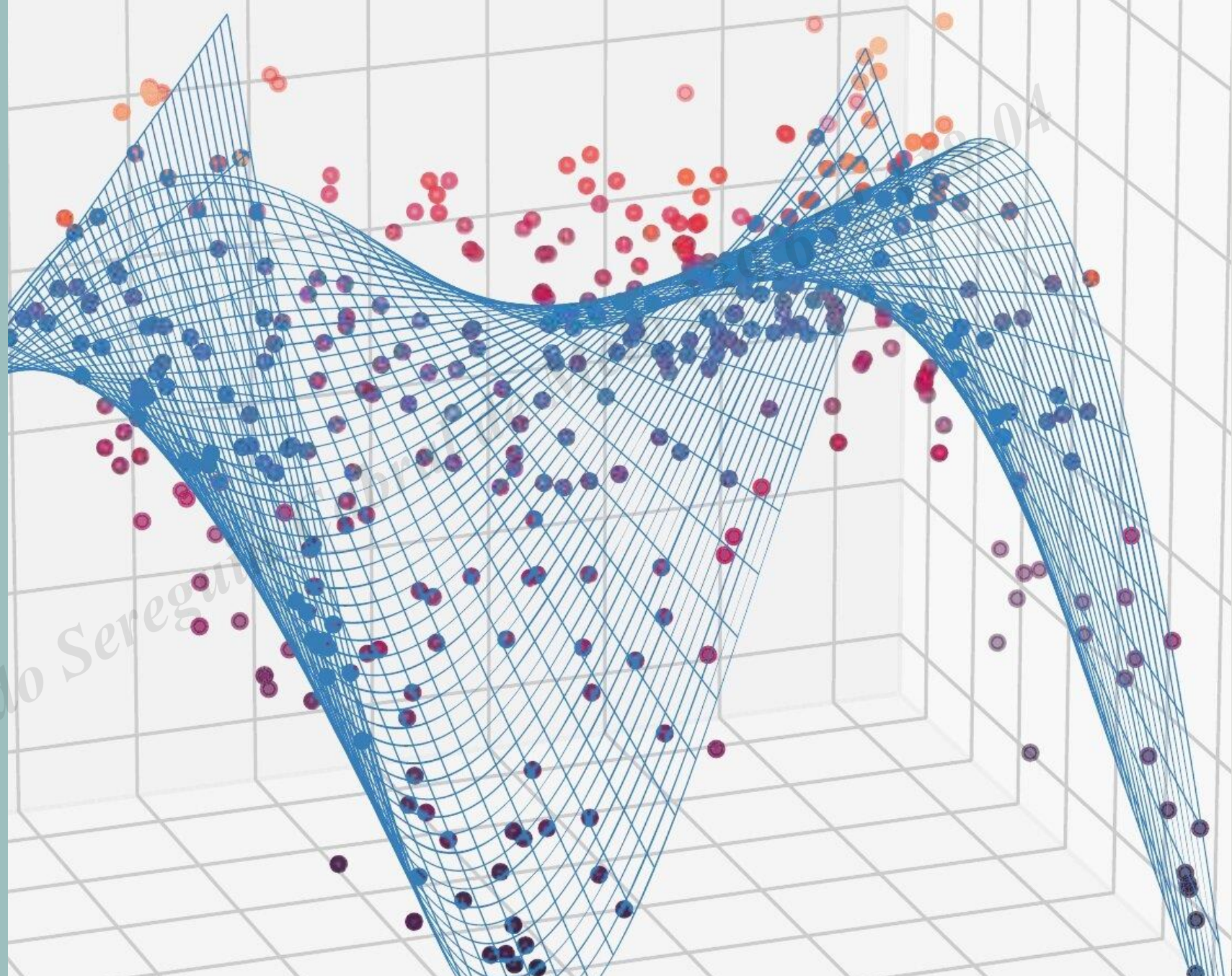
Construction and structuring *datasets*;

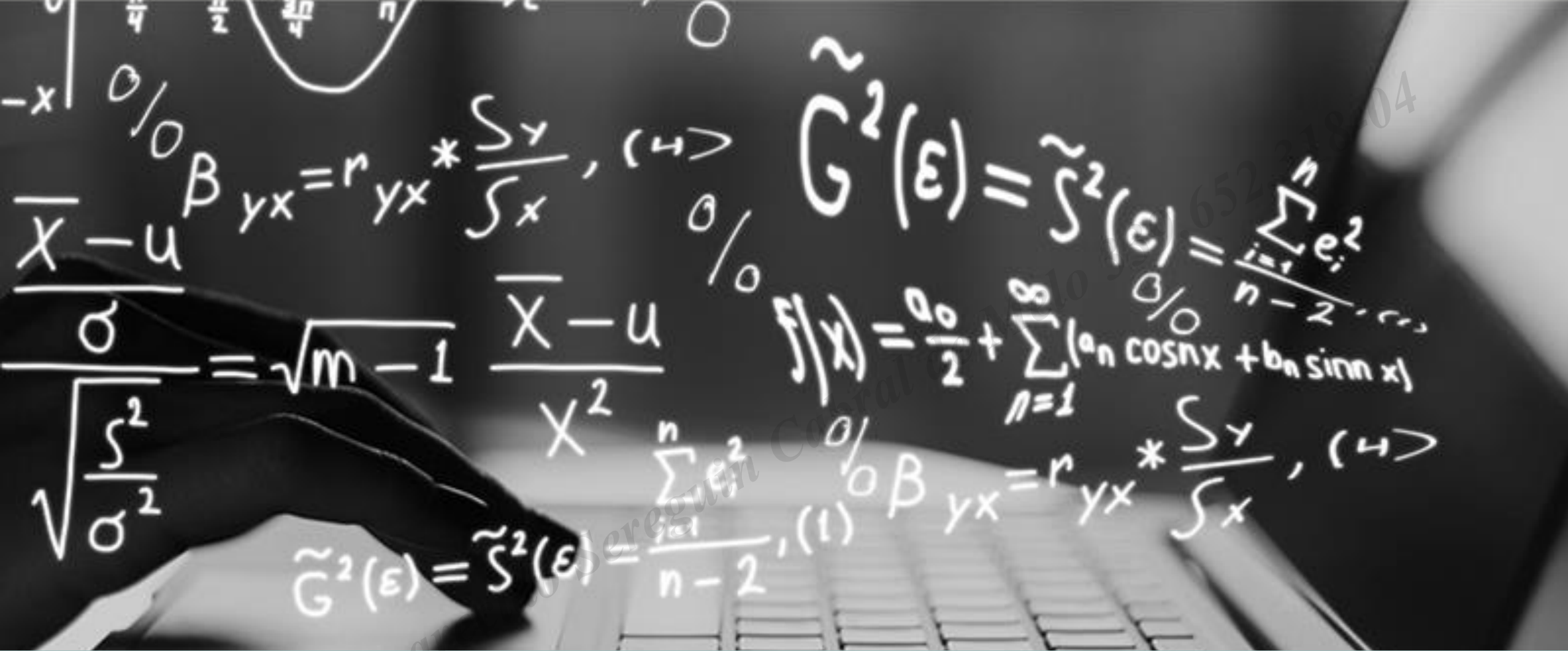
Scales of variables measurement;

Introduction to R programming language;

Algorithms of *machine learning*, encompassing:

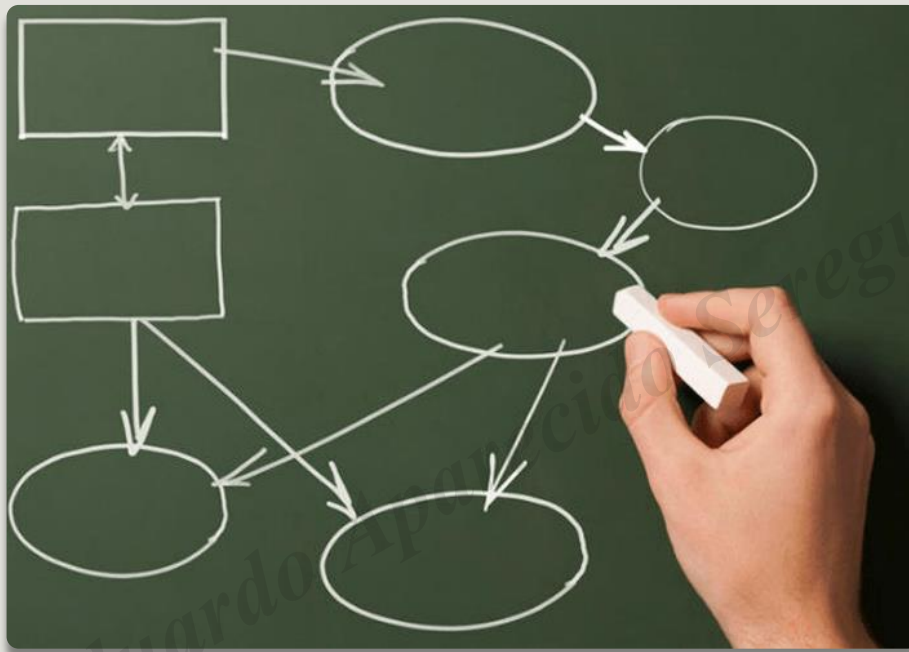
- Unsupervised Algorithms – exploratory techniques;
- Supervised algorithms – predictive techniques.





What are algorithms?

Algorithms – a concept:



Algorithms are explicit, literal, limited and systemic sequences of instructions and operations directed to the achievement of a preset objective data.

Basically, any known verb, as long as it denotes an action intended for humans, can be considered an algorithm.



Link: <https://www.youtube.com/watch?v=pdhqwbUWf4U>

Basic Taxonomy of Species of *Machine Learning* Algorithms

UNSUPERVISED

Machine learning techniques based on **unsupervised** algorithms (*unsupervised learning*) do not have the capacity of inference. They are dedicated to an exploratory, diagnostic analysis, of a studied phenomenon. Common examples: *Clusters* Analysis, Factorial Analysis of Main Components, Simple and Multiple Correspondence Analysis etc.

SUPERVISED

Machine learning techniques based on **supervised** algorithms (*supervised learning*) have the capacity of inference.

Therefore, they are dedicated to a confirmatory, predictive analysis of a given phenomenon studied. Common examples: Linear Regressions, Logistic Regressions, Decision Trees, *Random Forests*, Neural Networks, etc.

ere

Step 1: Unsupervised Algorithms or Supervised Algorithms?



The first step is the definition of the problem that you want to solve, whether academic or not.

- If there are objectives of **making inferences** for observations that are not present in the sample that was used for the training of the algorithm, **the ideal is the use of supervised algorithms**;
- If there are objectives of **making diagnoses**, without the intention of making inferences for observations that not present in the sample that was used for the algorithm training, **the ideal is the use of unsupervised algorithms of the unsupervised algorithms**.

Step 2: The Construction and Structure of a Database

As a general rule, the databases are structured in the following way: variables in columns and observations in lines.

The diagram illustrates the structure of a database. A blue box labeled 'Columns: variables' has arrows pointing to the column headers of the table. Another blue box labeled 'Lines: observations' has arrows pointing to the rows of the table.

id	suspicious words	unknown sender	presence of images	classification
1	yes	no	yes	spam
2	yes	yes	no	spam
3	yes	yes	no	spam
4	no	yes	yes	genuine
5	no	no	no	genuine
6	no	no	no	genuine

Step 3: What are the Measurement Scales of their Variables?

The incorrect definition of the scales of measurement of database variables is one of the main errors in the application of *machine learning* techniques. This error is irreparable, implying the restart of the entire modeling process, due to the biases created (e.g.: arbitrary weighting).

In short: are your variables only quantitative, only qualitative, or are both types present?



A black and white photograph of numerous wooden letter blocks scattered on a dark surface. In the center, the word "VARIABLE" is spelled out using ten blocks, arranged in a gentle upward curve. Other blocks with various letters like 'D', 'O', 'E', 'N', 'L', 'F', 'S', 'H', 'G', 'Y', 'I', 'M', 'X', 'A', 'C', 'T' are visible around the central word.

VARIABLE

What is a variable?

What is a variable?

Variables can be understood as a characteristic of a sample or population, which can be measured, or counted or categorized.

Good introduction examples are the height and/or weight of people, their income brackets, the color and/or model of cars they drive.

Individuals of a sample or population, not necessarily, need to be people in their physical sense. They can be objects, districts, municipalities, organizations, groups, cells, molecules, stars, etc. Thus, the characteristics of the individuals mentioned would be considered their variables.

For the course, we will establish the scale of measurement of variables in two: i) qualitative variables; and ii) quantitative variables.

Qualitative Variables

- They are also known as latent variables or categorical variables. They are variables that can not be measured; so only, categorized or counted.
- As they cannot be measured, they do not allow the calculation of descriptive statistics of position – e.g.: the mean and median.
- On the other hand, we can establish frequency tables for its categories.
- They are divided into nominal categories and ordinal categories.

Quantitative Variables

- Also known as metric variables, unlike qualitative variables, quantitative variables can be measured, and have, of course, a respective unit of measure.
- They allow the calculation of the mean and median, for example.
- They are divided into continuous variables and discrete variables.



Unsupervised Algorithms

Grouping Analysis

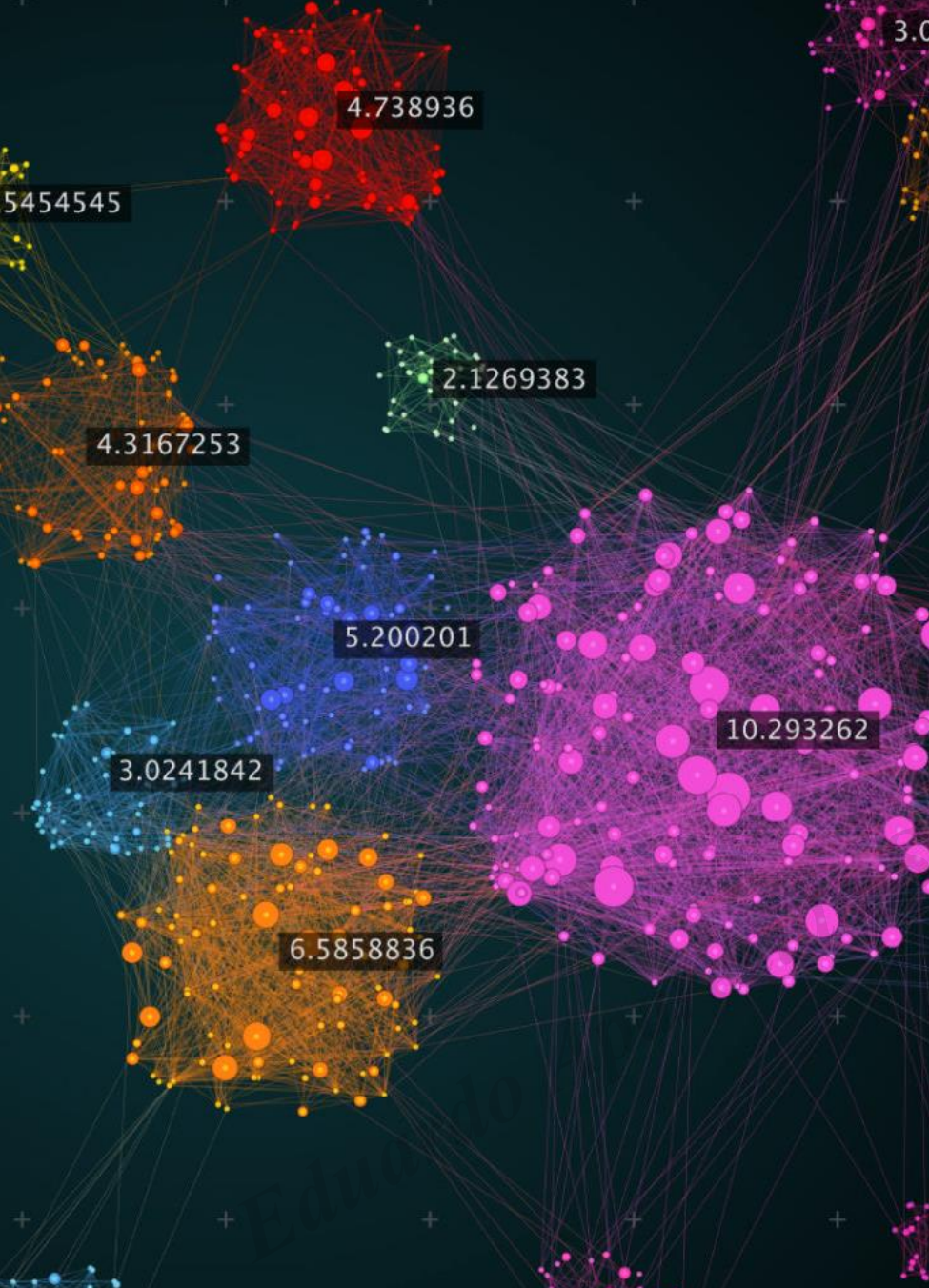
- Metric variables (distance measurement);
- Binary variables (similarity measurement).

Factor Analysis by Principal Components

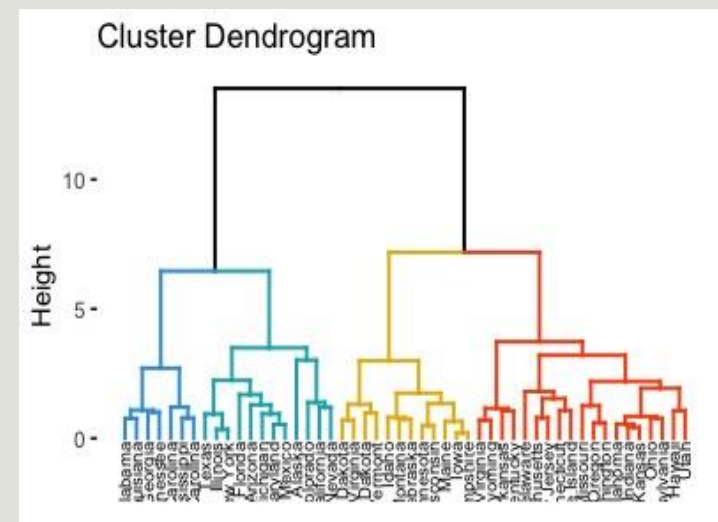
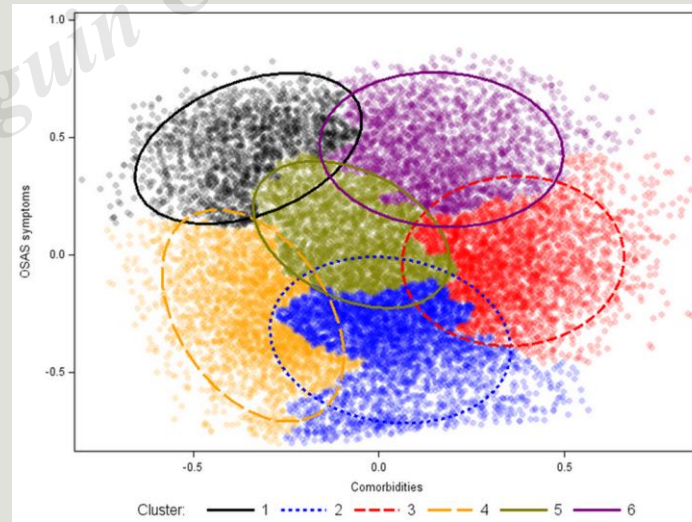
- Metric variables.

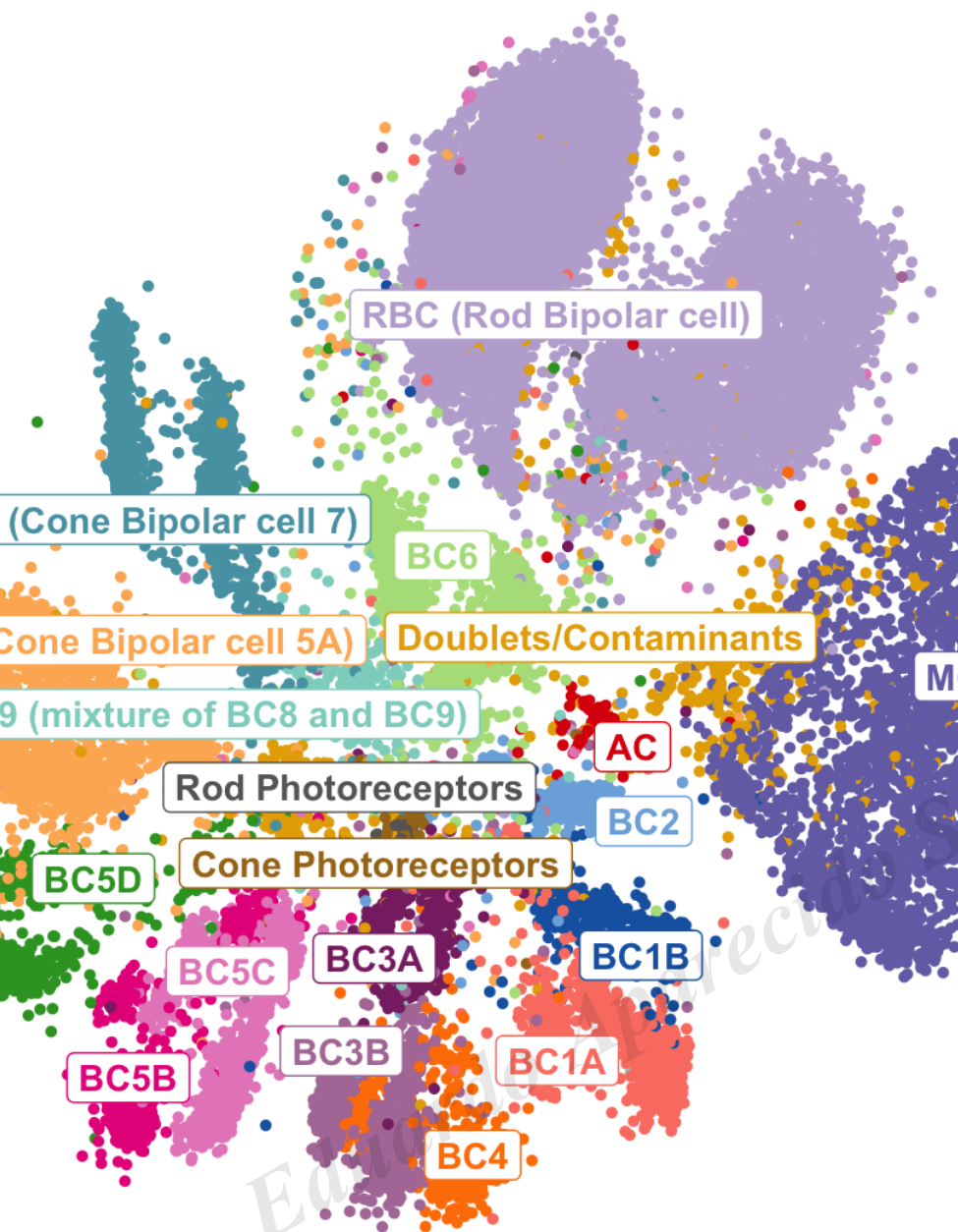
Simple and Multiple Correspondence Analyses

- Categorical variables.

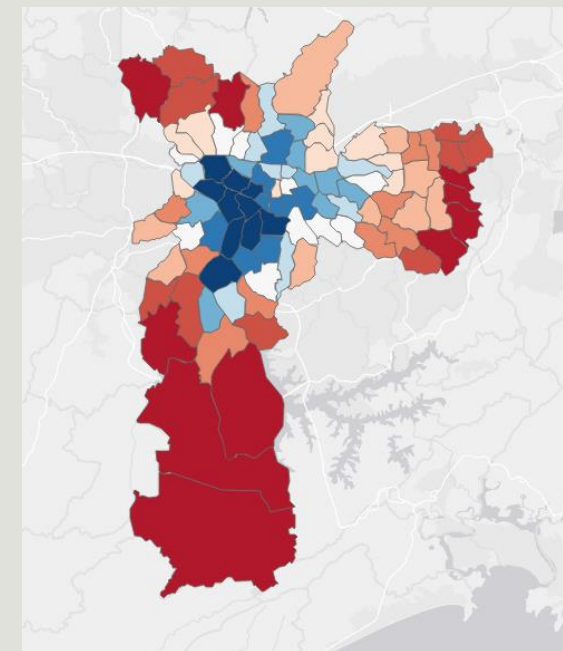
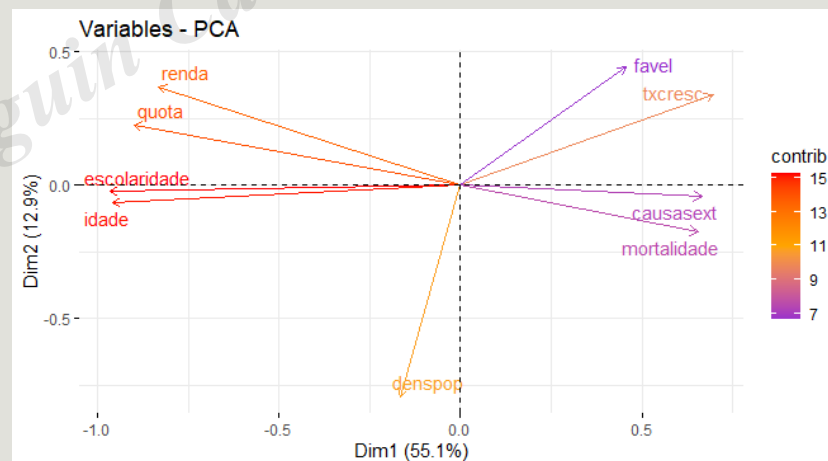


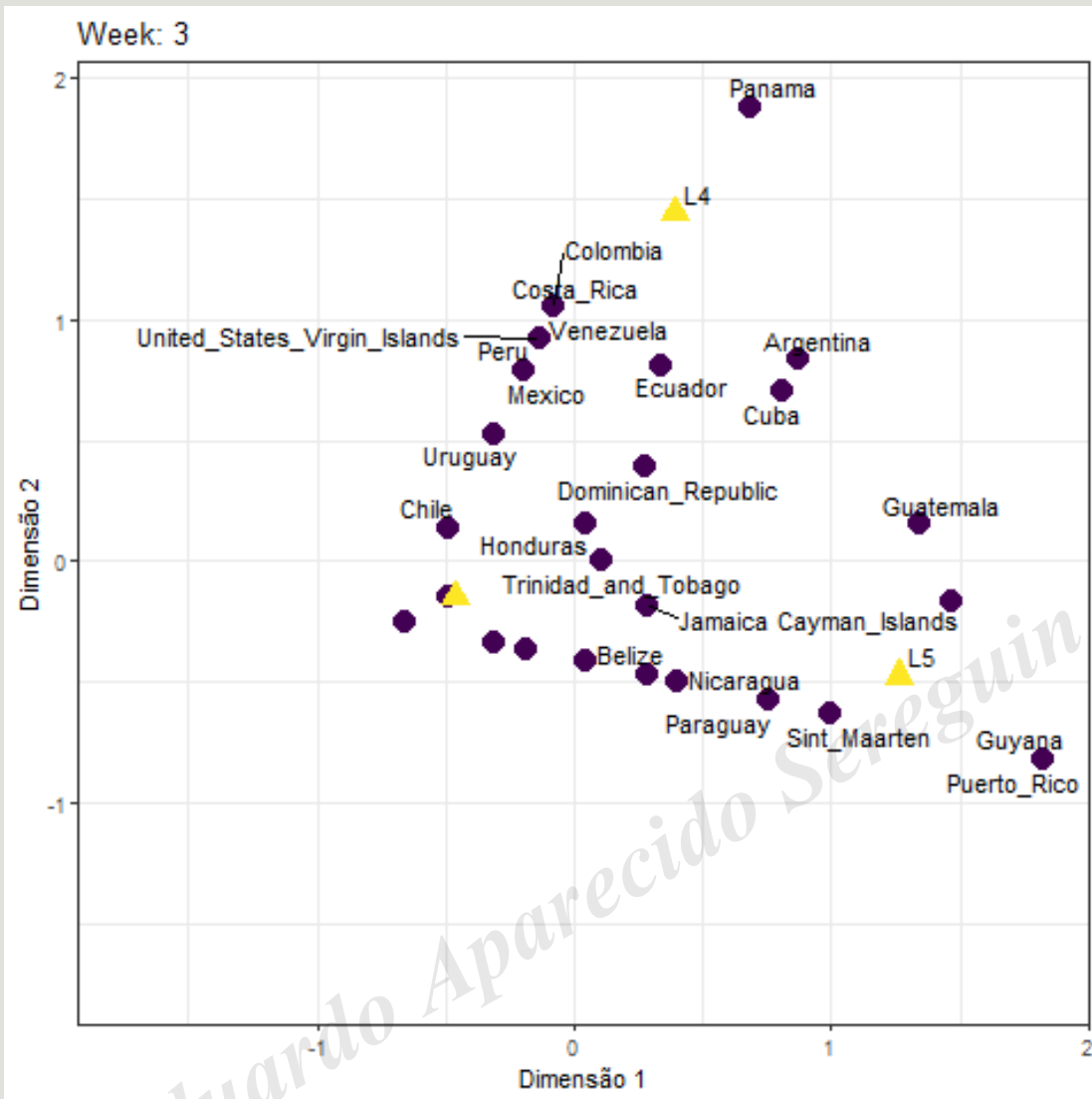
Grouping Analysis





Factor Analysis – PCA





Simple and Multiple Correspondence Analysis

Supervised Algorithms

Generalized Linear Models

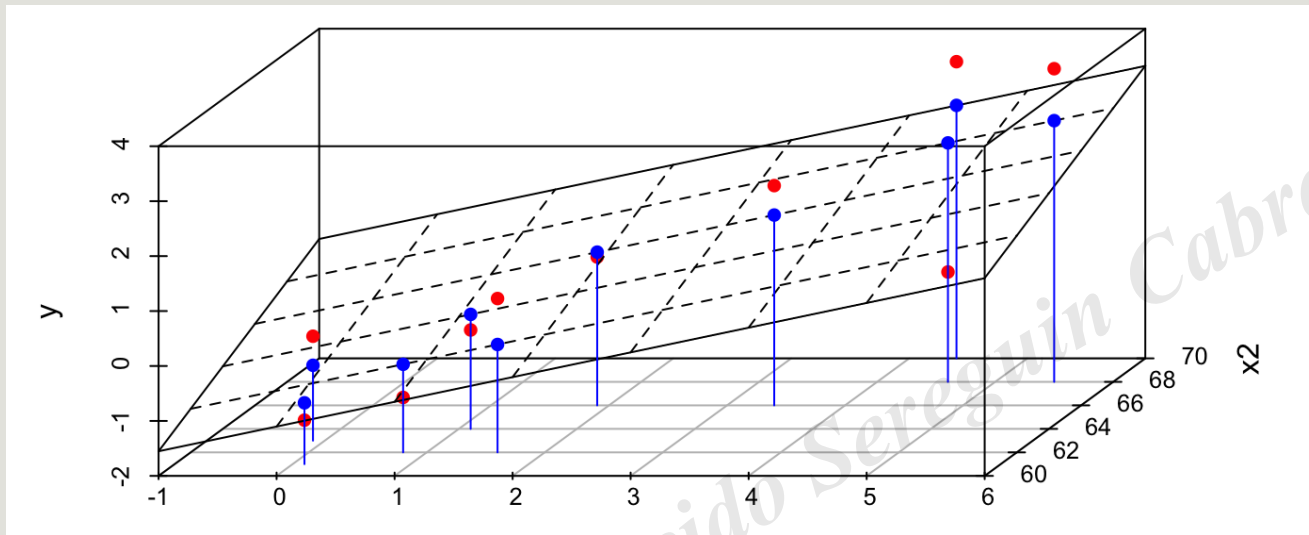
- Simple and Multiple Linear Regression;
- Binary and Multinomial Logistic Regression;
- Regression for Count Data;
- Zero-inflated Regression for Count Data.

Generalized Linear Mixed Models

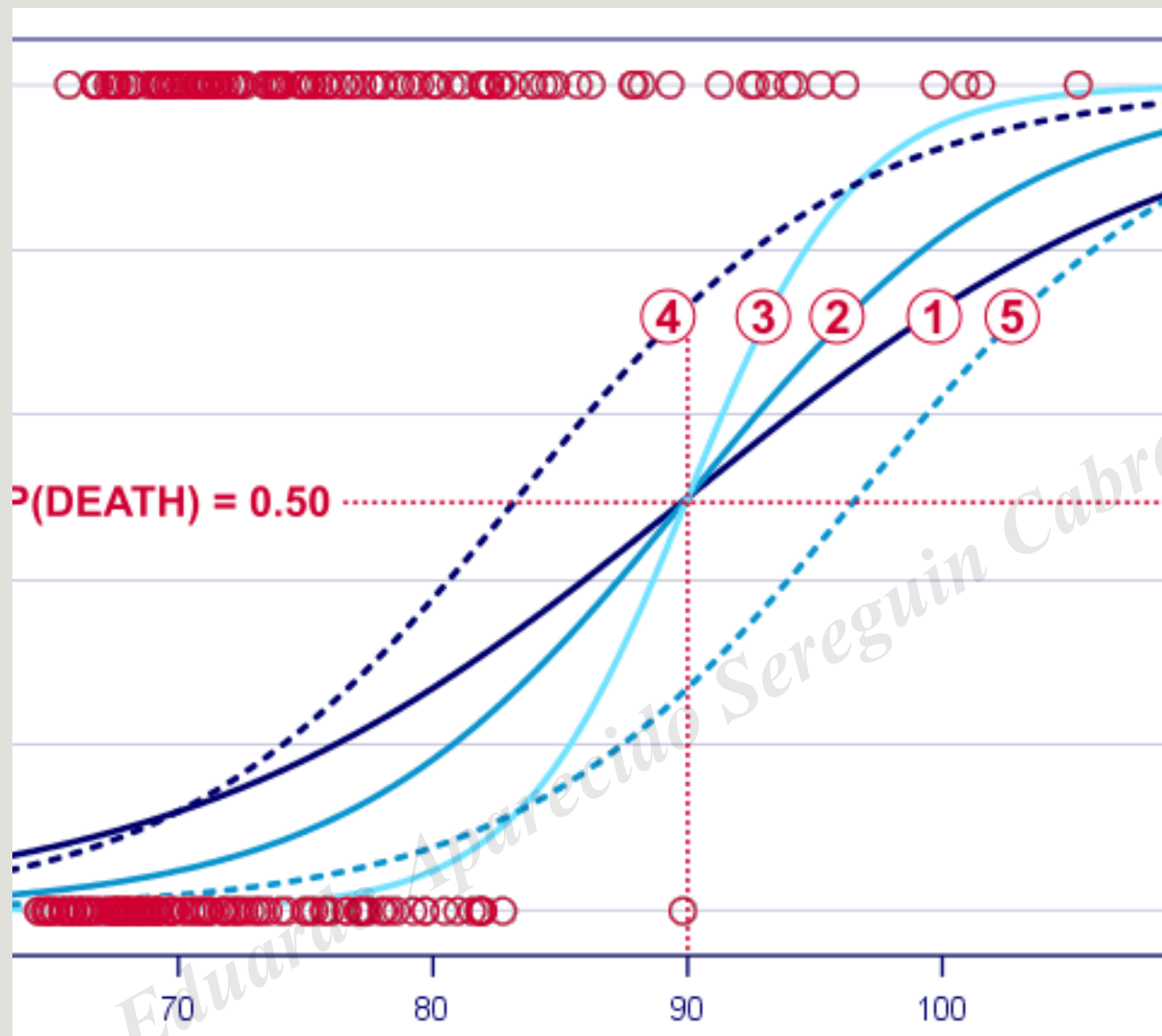
- Multilevel Linear Regression of 2 and 3 Levels.

Ensemble Models and Neural Networks

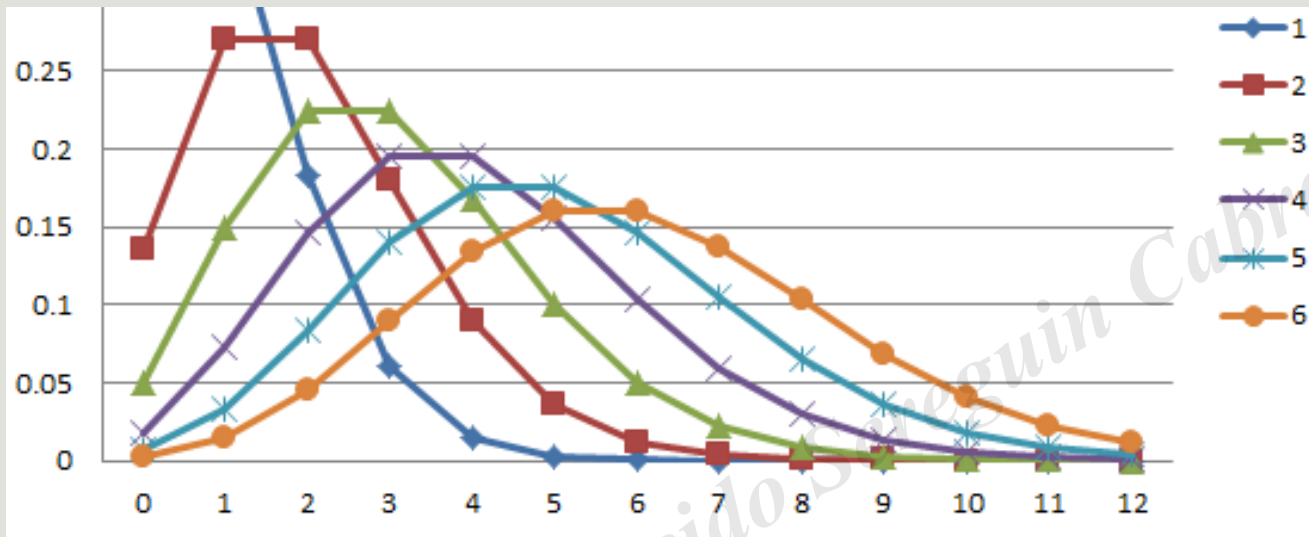
- Decision Trees of Classification or Regression;
- *Boosting*;
- *Bagging*;
- *Random Forests*;
- Artificial Neural Networks.



Linear
Regression

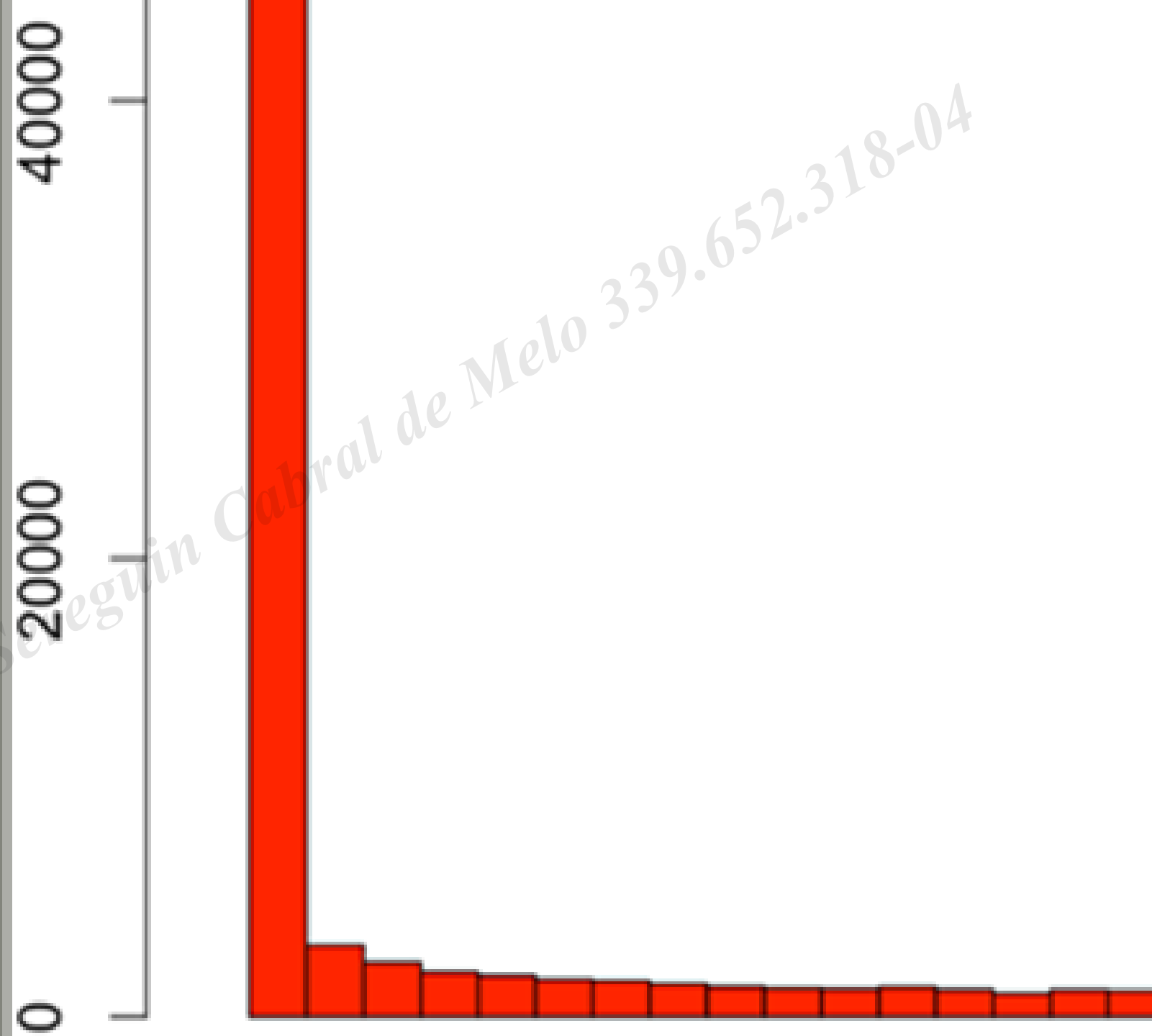


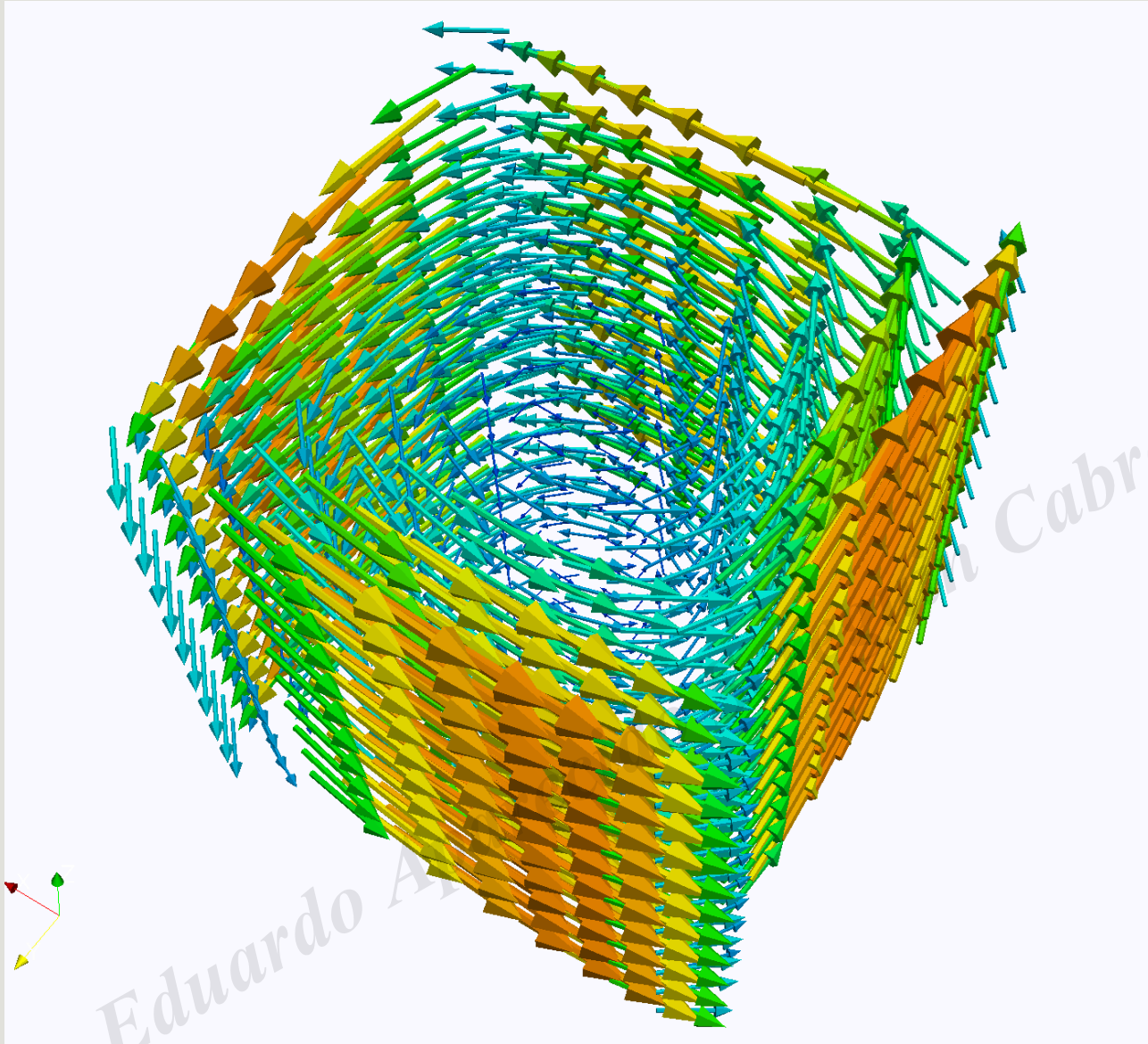
Logistic
Regression



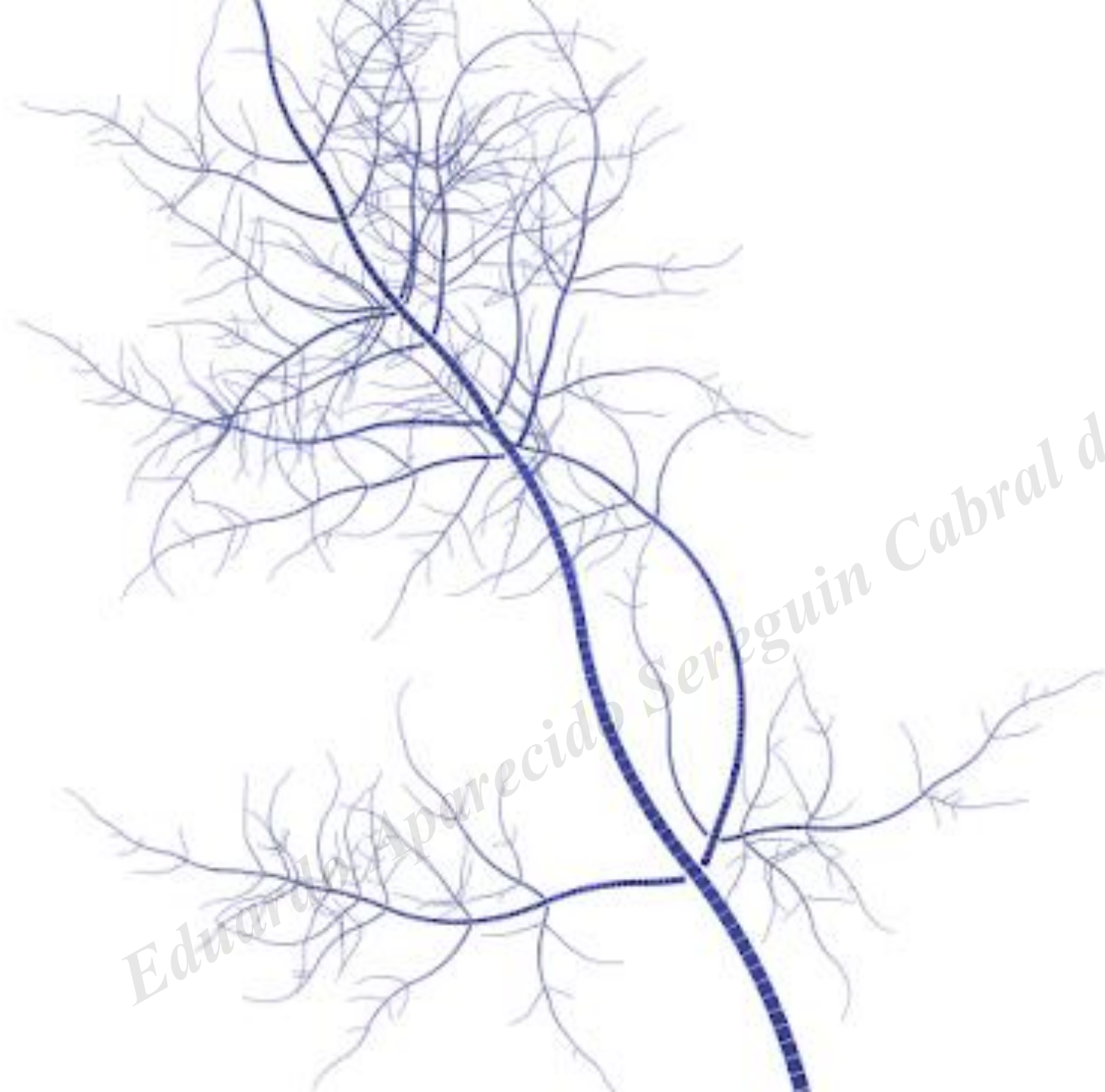
Regression for
Count Data

Zero-inflated
Regression for
Count Data.



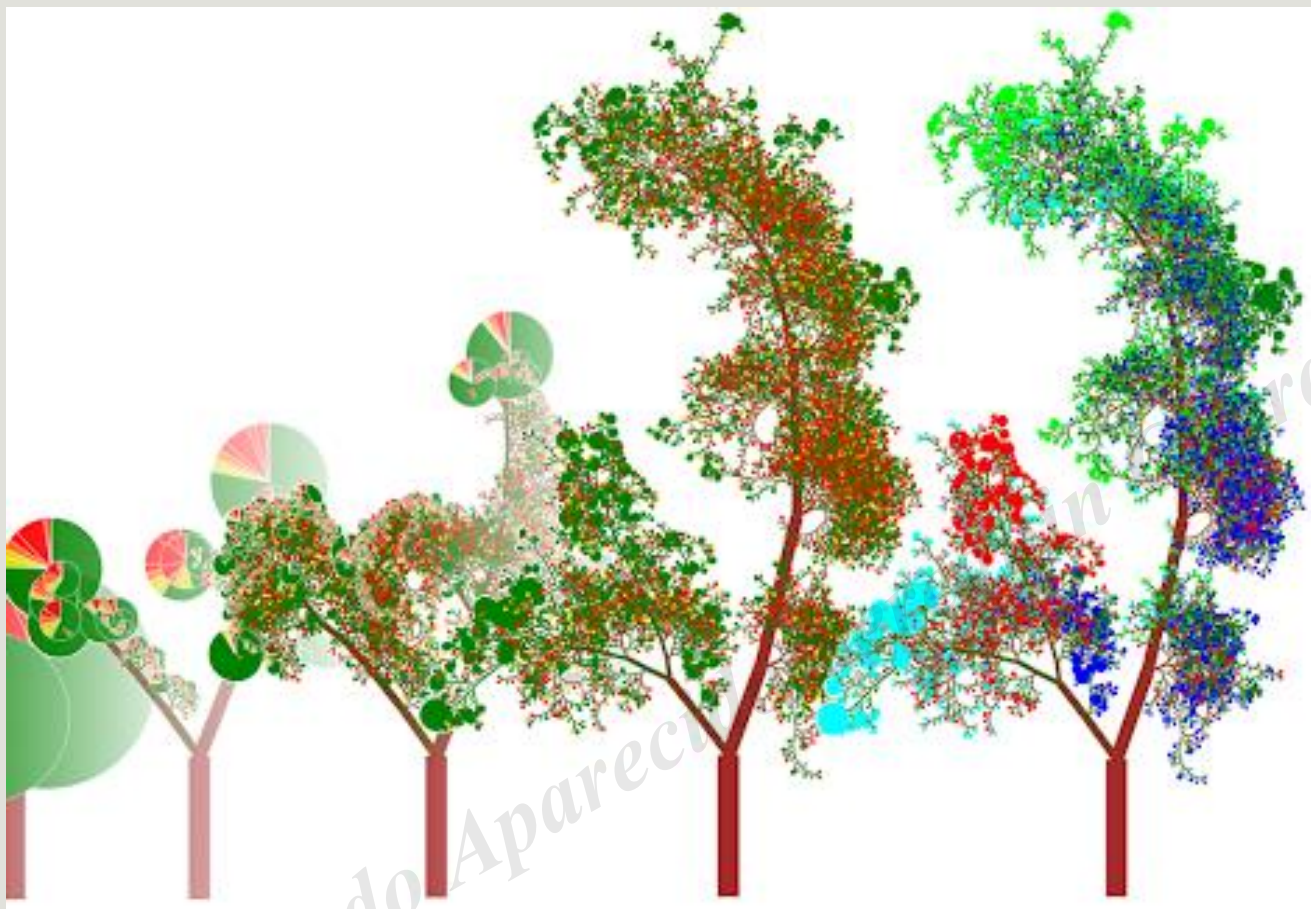


Multilevel Regression



Decision Trees

Eduardo Aparecido Sereguin Cabral de Melo 339.652.318-04



Random Forests

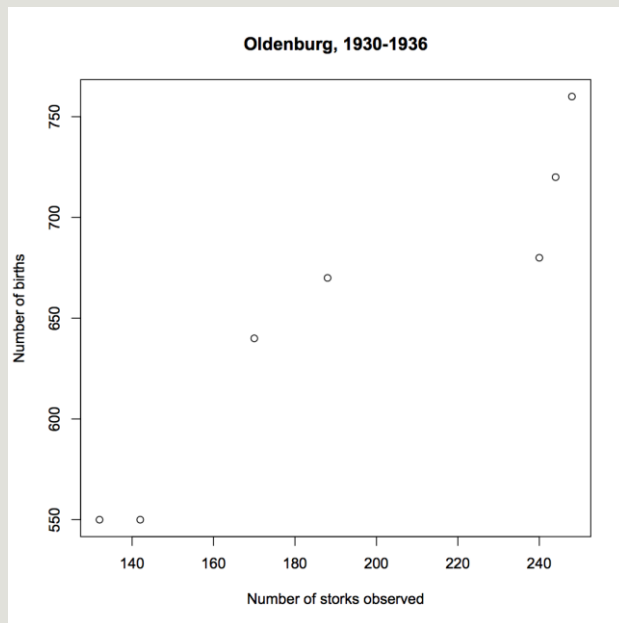


Melo 339.652.318-04

Neural Networks



Cause and Effect Relationship

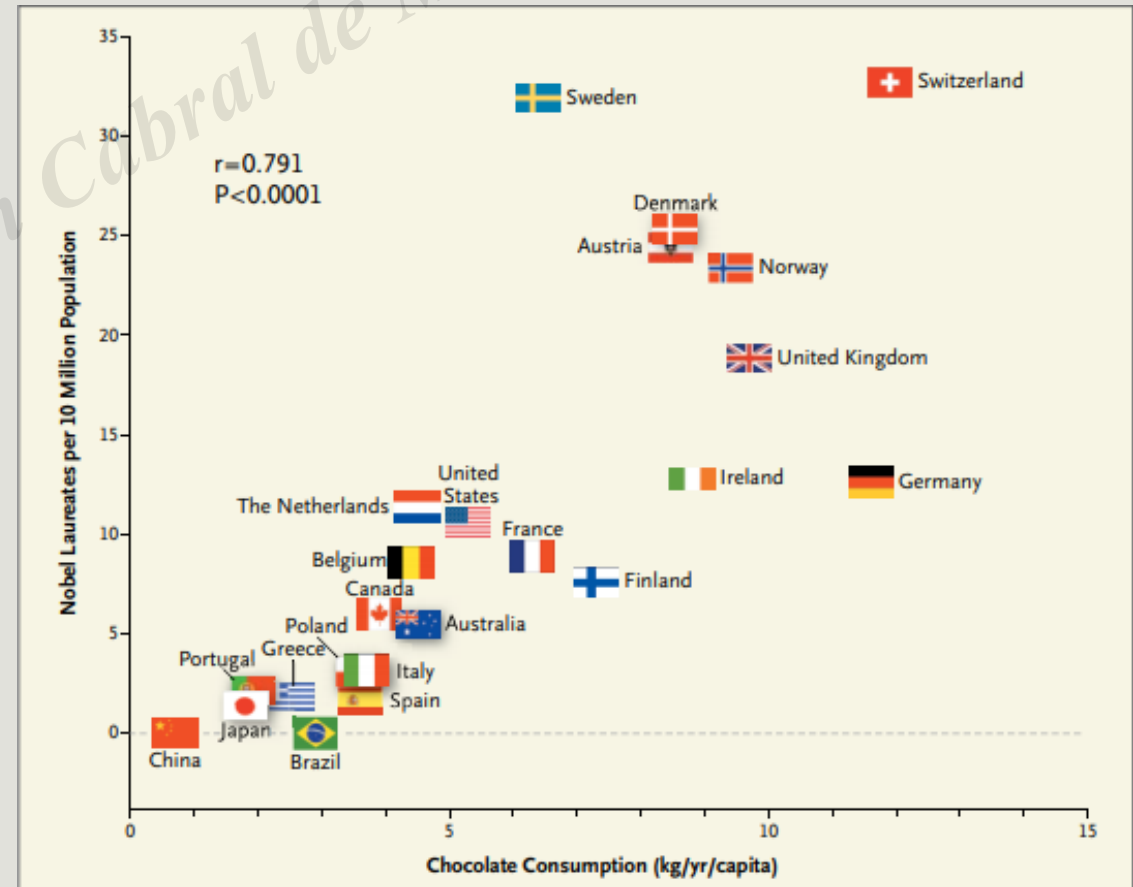


Didactic Example - 01

Data adapted by ox, Hunter, e Hunter de G. Fischer, Ornithologische Monatsberichte, vol. 44, no. 2, Jahrgang, 1936, Berlin e vol. 48, No. 1, Jahrgang, 1940, Berlin, e Statistisches Jahrbuch Deutscher Gemeinden, 27-33, Jahrgang, 1932-1938.

Didactic Example - 02

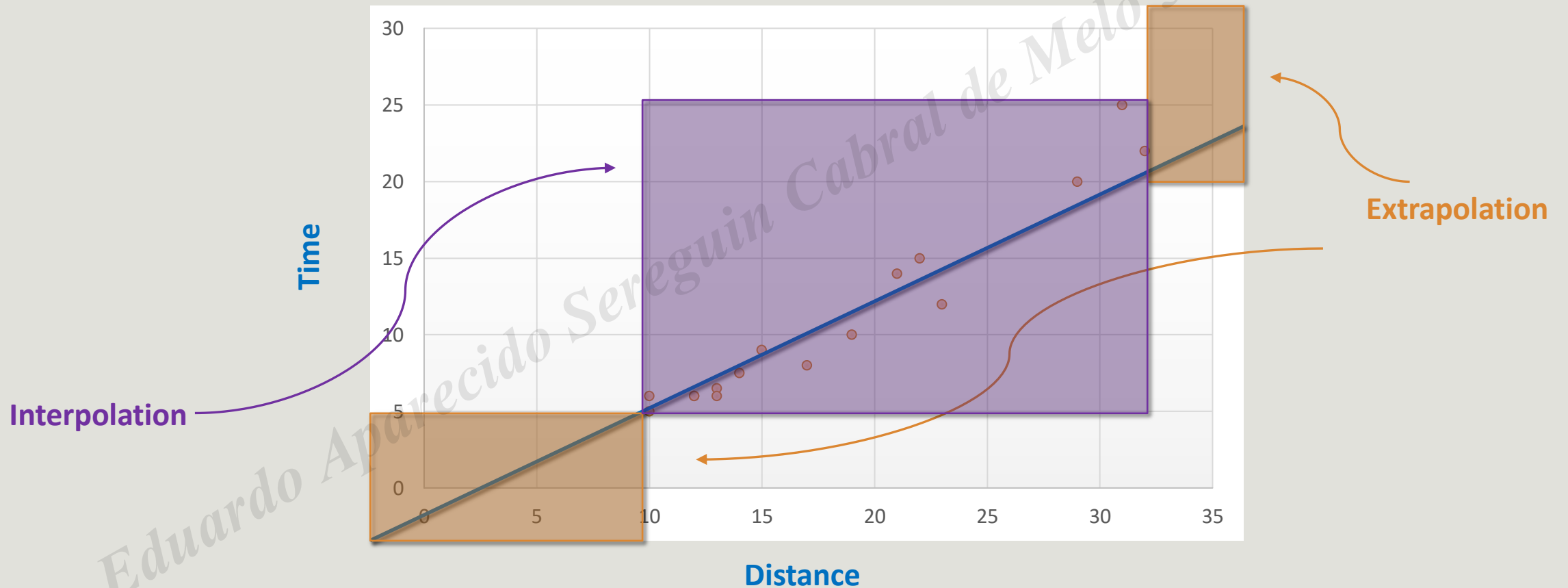
Data of Messerli, F. (2012). Chocolate Consumption, Cognitive Function, and Nobel Laureates. The New England Journal of Medicine, 367, pp. 1562-1564





Interpolation x Extrapolation

Interpolation and Extrapolation





Introduction to the R Programming Language



Why Learn a Programming language?

- Learn to program is very important when we want to understand the data! Plus: it is important to ensure that we can interpret the data so we can transform them into information!
- If you work or want to work with data, programming is an extremely relevant skill

A fair question would be :

"I already work or wish to work with data, and I already have access to them via Windows, OSx, Android, iOS, Chrome, Mozilla, Edge, Safari..."

Still, do I need to learn to program?"

The Limitations of a *Graphic User Interface* (GUI)



Reproducibility

Automation

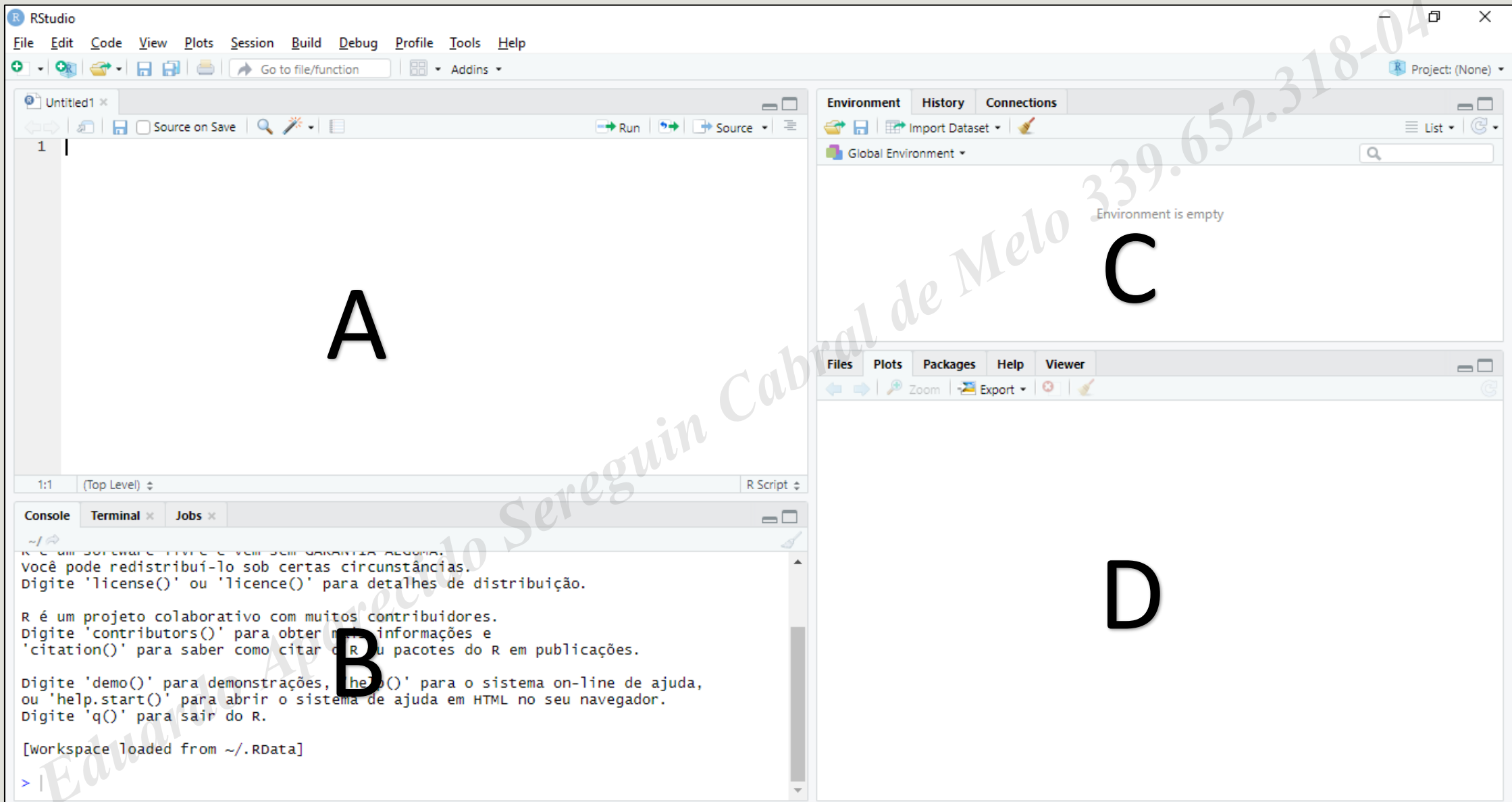
Communication

Why the R?



About the R

- The R language emerged in 1995, derived from the S language, and is object-oriented.
- It has several packages (over 17 thousand) with advantage for the application of the Advanced Statistics and a vast support community, in addition to strong capacities focused on the Data Science.
- Comprehensive R Archive Network (CRAN) is the R language repository in which each user can contribute to new packages (collections of functions in R with a compiled code). These packages can be easily installed with a command line.
- Recommended readings for those who are not yet familiar with the R language and wish to delve deeper:
 - Hands-On Programming with R (Grolemund, 2014);
 - R for Data Science (Wickham & Grolemund, 2016);
 - Ciência de Dados com R (Oliveira, Guerra & McDonnell, 2018).



Objects, Functions and Arguments

There are authors who define R objects as being a variable. For the course, we will understand that variables correspond to characteristics of a sample or population.

- Objects are simple ways of accessing something that was saved in the machine memory. It can be a value, a word, one or more variables, an URL, a sample or population database, a list of different things containing information and different sizes, a graph, a graph, a graph, an image, a new command, etc. **In R EVERYTHING is an object!** Each of these objects has a class!
- Functions correspond to actions, to orders directed to the machine;
- Arguments correspond to refinement or a better direction of the actions or orders proposed by the functions.

Creating an Object in the R

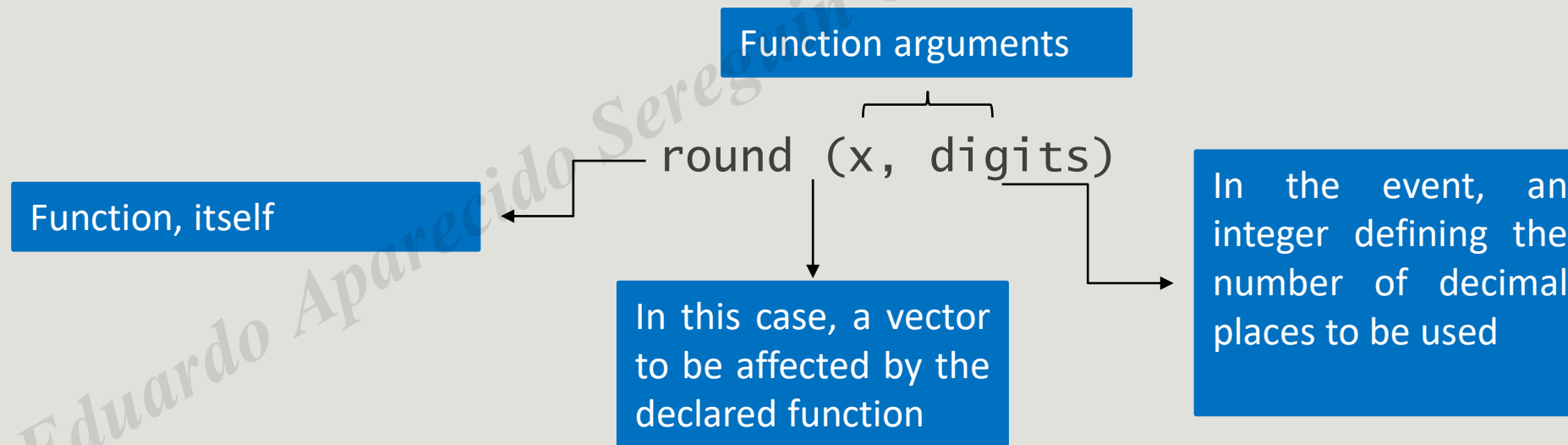
- There are two ways of creating an object in the R language:
 - Using the symbol =; or
 - Using the symbol <- called the assignment operator (prefer this form, reserving the symbol = for the value assignments of function arguments or for mathematical operations). The assignment sign can be quickly declared by pressing the 'Alt' and '-' keys together.
- The names of objects established in R must follow certain rules:
 1. They must not start with numbers and neither by points;
 2. It is not desirable that they contain spaces, although the situation is possible with the use of symbols ``;
 3. Their nomenclatures also do not accept special characters, such as !, ~, \$, @, +, --, /*.
 4. It is recommended to avoid appointing the objects with the same name of functions already established;
 5. It is recommended not to use accents and, whenever possible, avoid capital letters, as the language is *case sensitive*.

Main Introductory Functions of the Course

Function	Serves to:
args()	Verify the arguments of a given function in R
round()	Round numbers
sample()	Create samples
class()	Verify the classes of R objects
View()	View objects in spreadsheet form
head()	View the first observations of a database
tail()	View the latest observations of a database
str()	Observe the structure of a database
length()	Observe the length of a vector or of a data list
dim()	Discover the dimensions of an object
nrow()	Count the number of lines of a database
ncol()	Count the number of columns in a database
rm()	Remove an object from the work environment
install.packages()	Install packages
library()	Load packages

Using functions in R

To use a function in R, we must know its functional form, that is, we must, as a rule, declare the arguments inherent to it. Example of using the `round()` function:



Packages in R Language

- The R language has thousands of packages directed to the most diverse areas of knowledge, and most is not installed in our computers. To install a package, we must command:

```
install.packages("package name here")
```

- The installation of a package is not enough for its use. Thus, in each new open section of RStudio, we must call them in the following way:

```
library(package name here)
```

Creating and Excluding Variables in a *Dataset*

- Creating a variable:

```
basededados$nova_variável <-  
  NA  
or  
mtcars["nova_variável"] <- NA
```

- Excluding a variable:

```
basededados$nova_variável <- NULL
```

Extracting Values from a Dataset

- To extract a column from a *dataset*, use the operator \$:

basededados\$nova_variável

- More precisely, values can also be extracted from the *dataset* with the operator [,]:

Statement of
which line you
want to access

database[,]

Statement of
which column
you want to
access

Functions `if`, `else` and `ifelse`:

```
if(logical test){
```

if the answer of the logical test is TRUE, do this

```
} else {
```

if the answer of the logical test is FALSE, do this other thing

```
}
```

```
ifelse(logical test,
```

yes = if the answer of the logical test is TRUE, do this ,

no = if the answer of the logical test is FALSE, do this other thing)

Functions `for`, `while` and `repeat`

```
y <- 10
```

```
for(i in 1:5){  
  print(y + i)  
}
```

For each `i` (could be any symbol or word), present in the sequence of 1 to 5, print the value of the sum between `y` and `i`

Eduardo Aparecido Sereguin Cabral de Melo 339.652.318-04

Functions `for`, `while` and `repeat`

```
z <- 0
```

```
while (z < 10) {  
  print(z)  
  z <- z + 1  
}
```

While `z` is smaller than 10, print `z` and then update the value of `z` by adding its value in a unit

Eduardo Aparecido Sereguin Cabral de Melo 339.652.318-04

Functions `for`, `while` and `repeat`

```
w <- 3
```

```
repeat{  
  print(w)  
  w <- w + 2  
  if(w > 18) break()  
}
```

Repeat the steps below:

- Print the value of `w`;
- Update the value of `w`, adding it in two units;
- If `w` becomes greater than 18, stop everything.

Visualization of Data with ggplot2

- The most basic syntax of the ggplot2, for the creation of a chart from a *data frame*, is the following:

```
ggplot(data = database here) +  
geom_geometry chosen here(aes(main elements of chart here))
```

Eduardo Aparecido Seregini Cabral de Melo 339.652.318-04

Obrigado!



Rafael de Freitas Souza

[Linkedin](#)