MBA USP ESALQ

Other Machine Learning Models II – bagging and boosting

João F. Serrajordia R. de Mello

You will need...

Preparations

- Open R
- Import libraries
- Something to take your notes

Agenda

Regression trees

Bagging – Random Forest

Boosting – Gradient Boosting

Grid Search CV

Regression trees

Regression trees

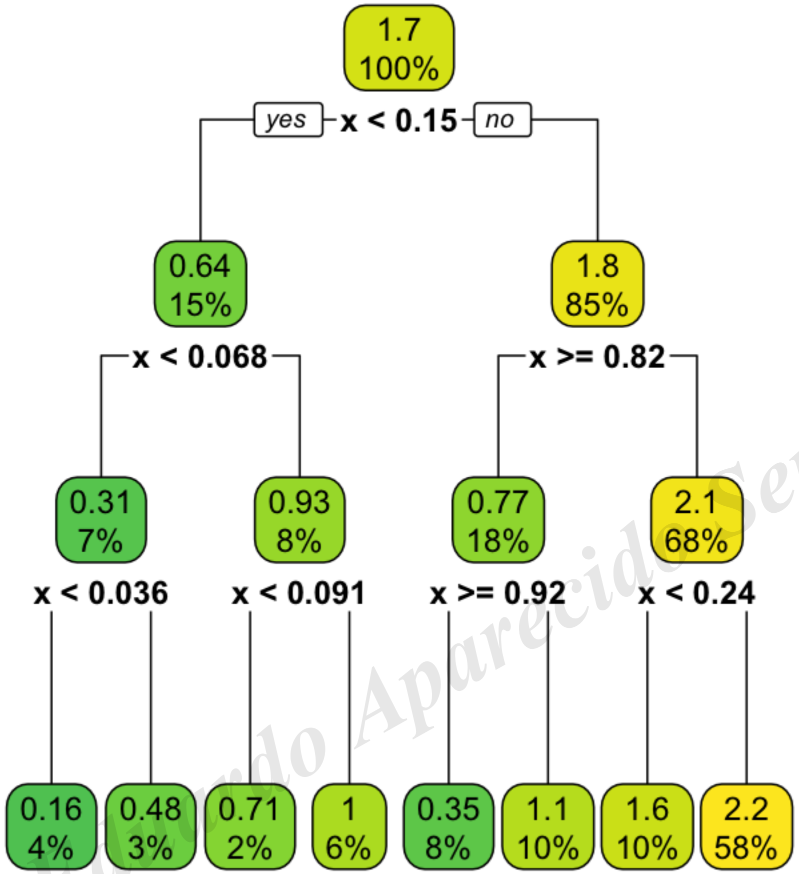
They are very similar to classification trees

The criterion of impurity is what changes.

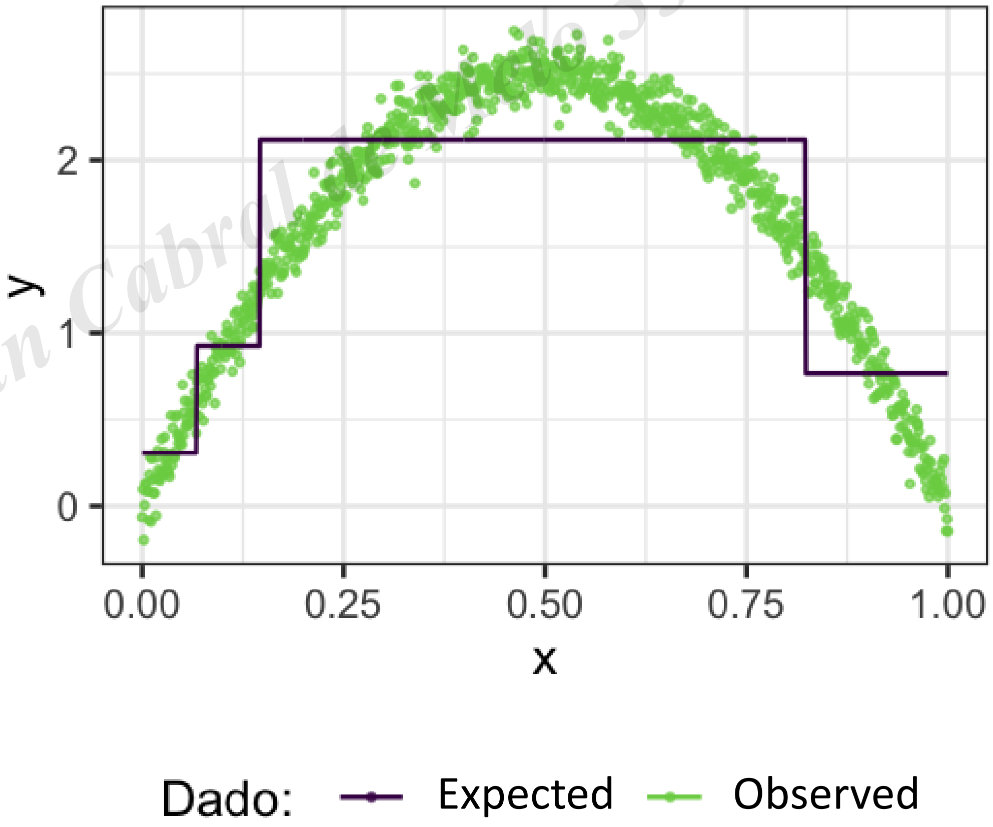
$$SQE = \sum_{i=1}^{N} (y_i - \hat{y}_i)^2$$

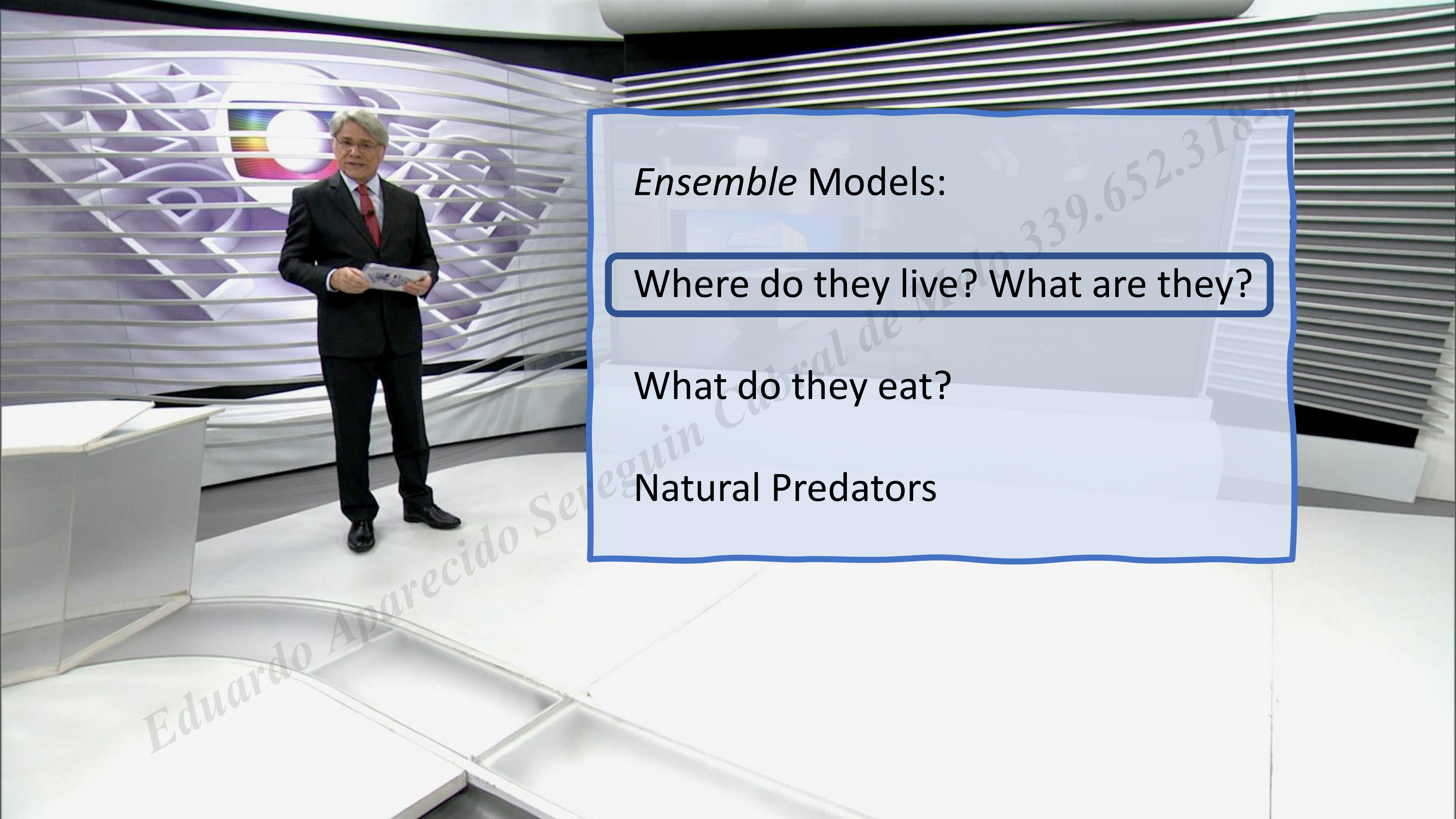
https://view.genial.ly/62094a9c11415700197ff277

Regression trees



Observed vs expected values





Ensemble Models:

Where do they live? What are they?

What do they eat?

Natural Predators

Predictive and classification problems

What is the efficacy of a vaccine?

Will the customer pay the loan?

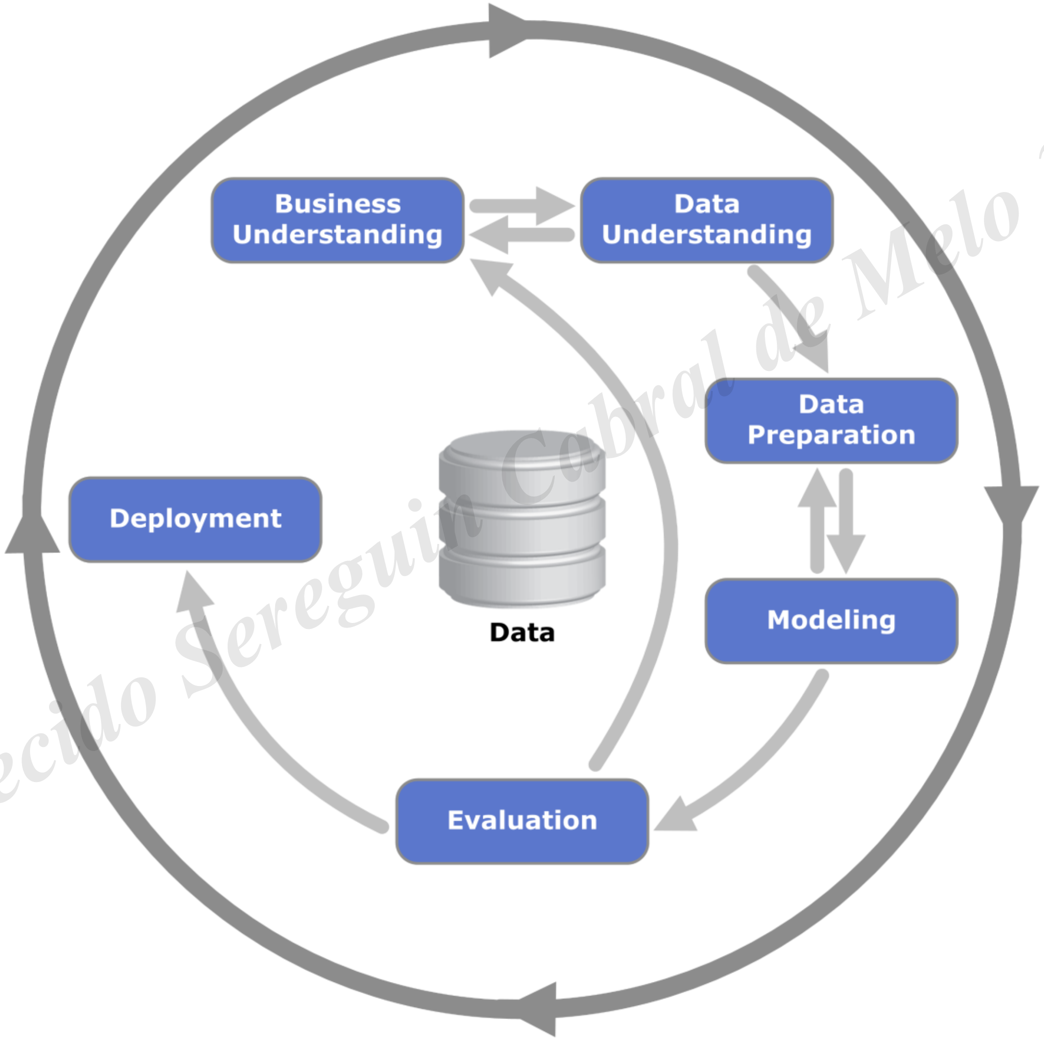
How much oil is in the well?

Will the customer buy my product?

What is the person doing?

How green is this vehicle?

CRISP-DM



Source: https://www.the-modeling-agency.com/crisp-dm.pdf

Algorithms classification

Supervised

- Regression
- GLM
- GLMM
- Support vector machines
- Naive Bayes
- K-nearest neighbors
- Neural Networks
- Decision Trees

Unsupervised

- K-Means
- Hierarchical methods
- Gaussian Mixture
- DBScan
- Mini-Batch-K-Means

We are here!

Algorithms classification

Continuous response

- Regression
- GLM
- GLMM
- Support vector machines
- K-nearest neighbors
- Neural Networks
- Regression Trees

Discrete response

- Logistic Regression
- Classification trees
- Neural Networks
- GLM
- GLMM

We are here!

Algorithms classification

Machine Learning Methods

- Decision Trees
- Bagging
- Boosting
- K-NN
- Neural Networks
- Support Vector Machines

Machine Learning Statistics Methods

- Regression
- GLM
- GLMM
- ANOVA

We are here!

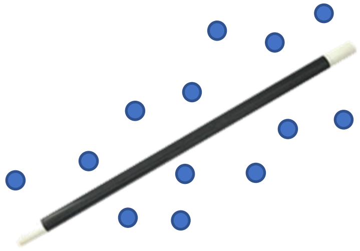
Ensemble

An ensemble is any combination of existing models. The main types are:

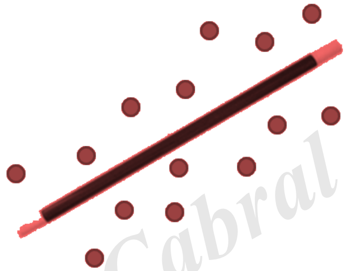
- *Bagging*
- *Boosting*
- *Stacking*

Ensemble - aggregation

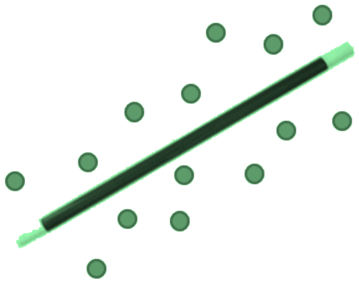
Model 1



Model 2



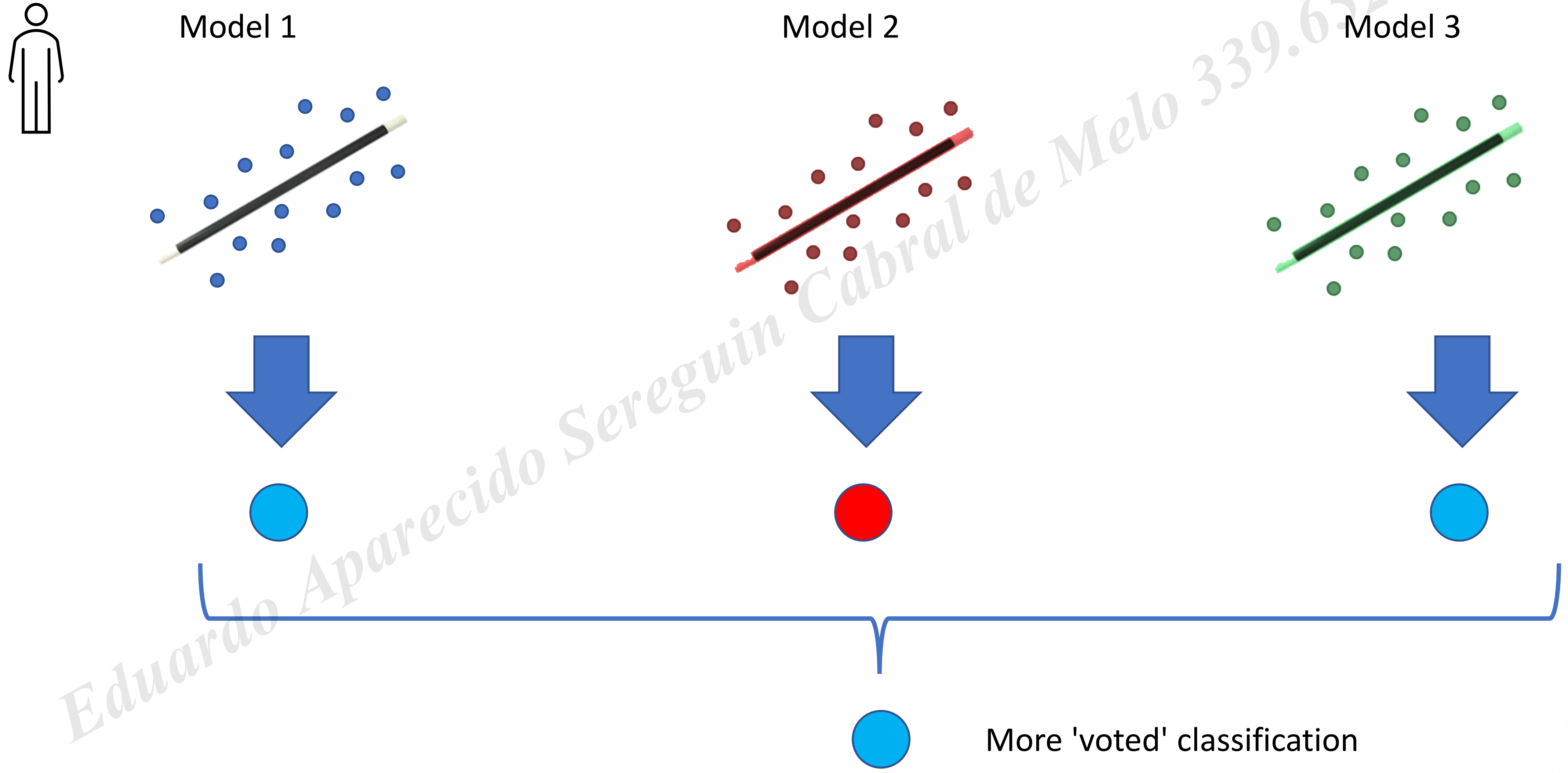
Model 3



An *aggregation* is a combination (in general a simple average) of the predictions of two or more previously constructed models.

Objective: even if each model is a "*weak learner*", the combination can be a *“Strong learner*" or a better predictor than each of the integrant.

Ensemble – Hard Voting



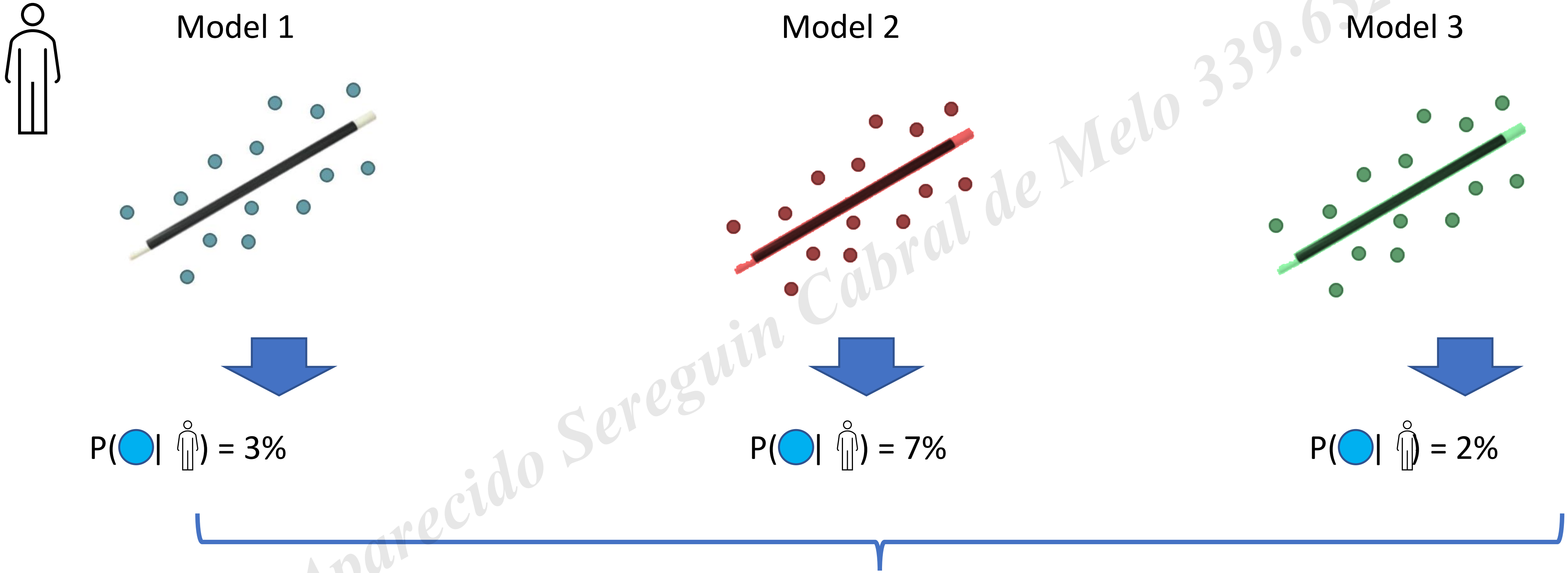
Model 1

Model 2

Model 3

More 'voted' classification

Ensemble - aggregation



Model 1

Model 2

Model 3

P(●| 🧍) = 3%

P(●| 🧍) = 7%

P(●| 🧍) = 2%

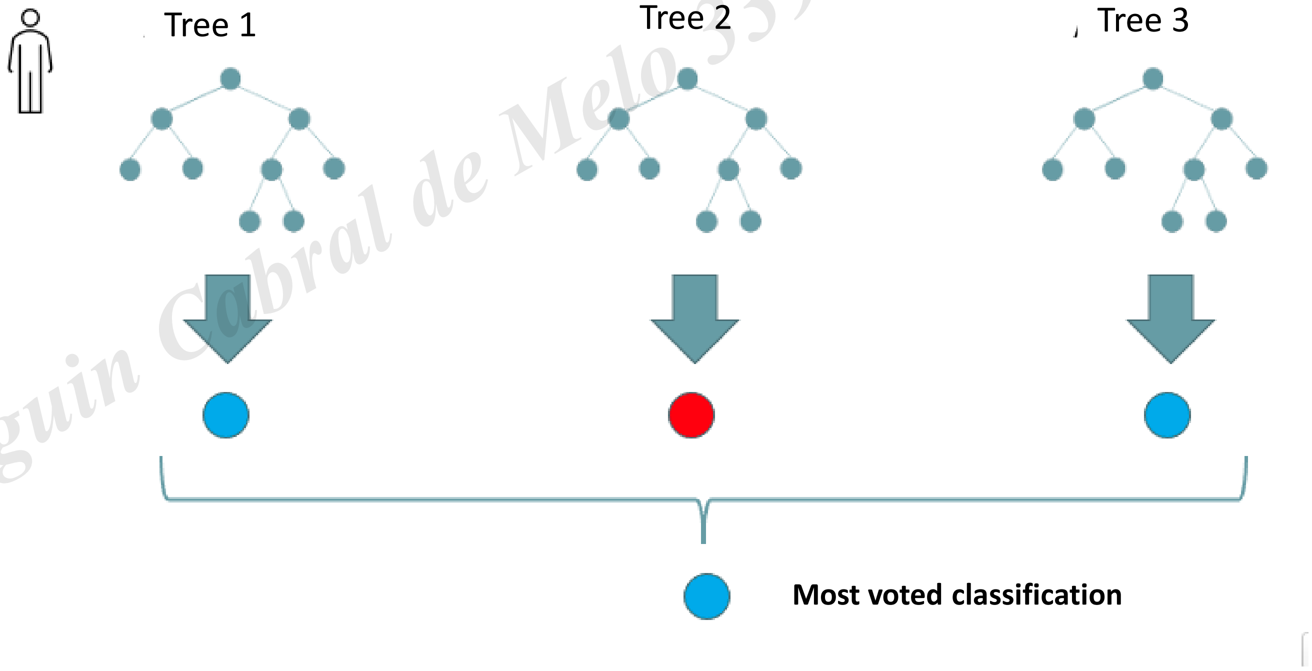
P(●| 🧍) = 4%

A simple aggregation method but also powerful consists of obtaining the average of several predictions.

Ensemble - aggregation

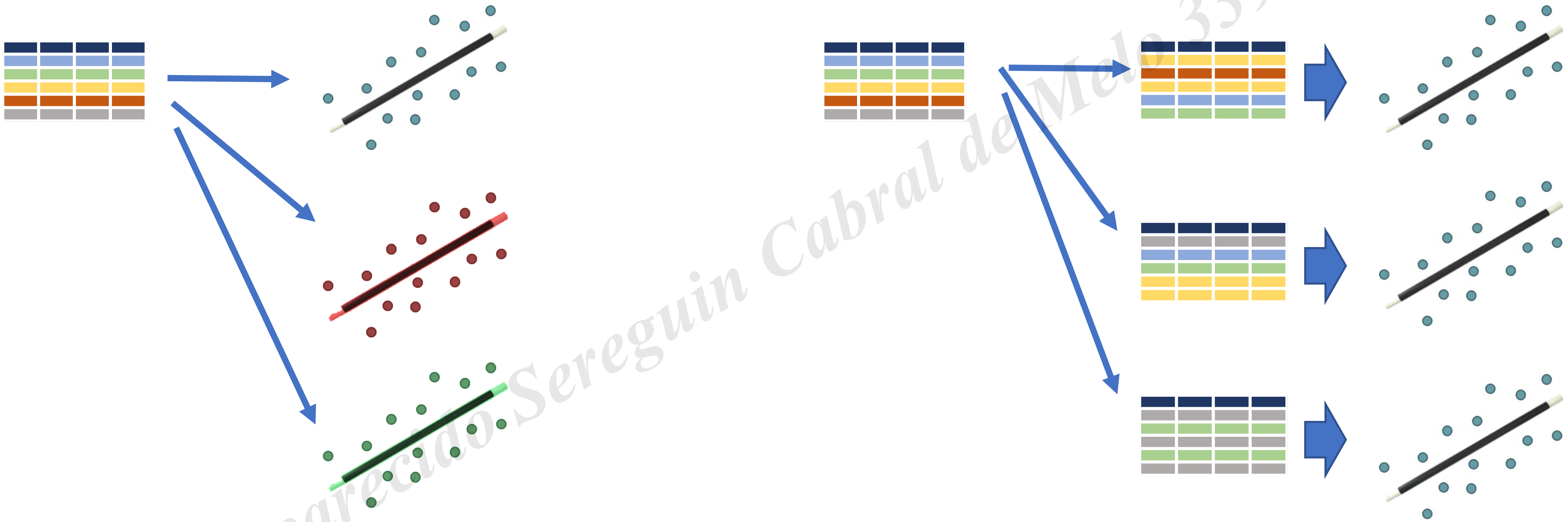
We want to add models that are:

- *Useful*
- *Have the same objective*
- *Different*



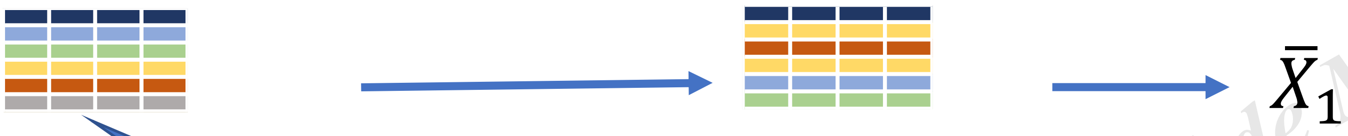
We want different predictors, but that they “indicate” the same response variable. An idea would be to generate predictors with some random 'disturbance'.

Bootstrapping to evaluate the average



And what happens if we change the base using the same algorithm instead of changing the algorithm?

Bootstrapping to evaluate the average

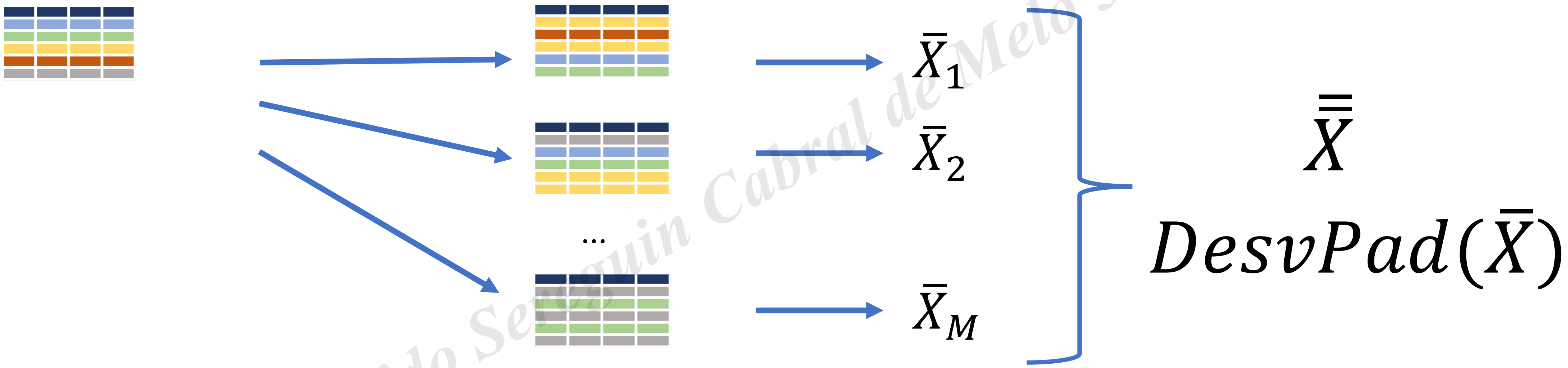


We have a set of size N data

We want to estimate the standard error of a parameter, for example, the average.

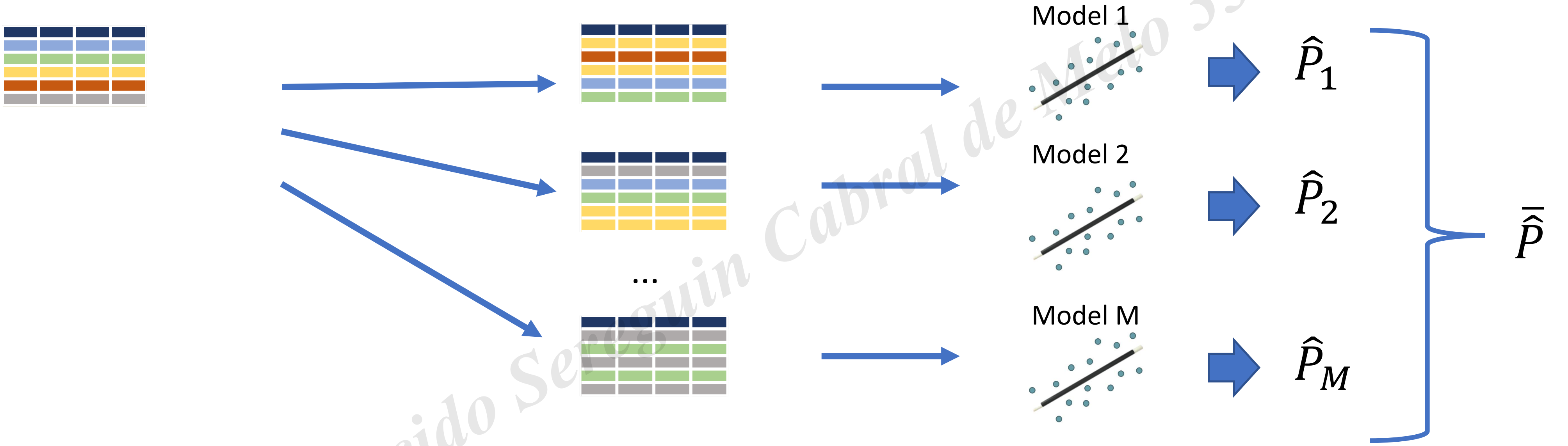
1) Remove a random sample of size N from the base
2) Calculate the parameter, store information

Bootstrapping to evaluate the average



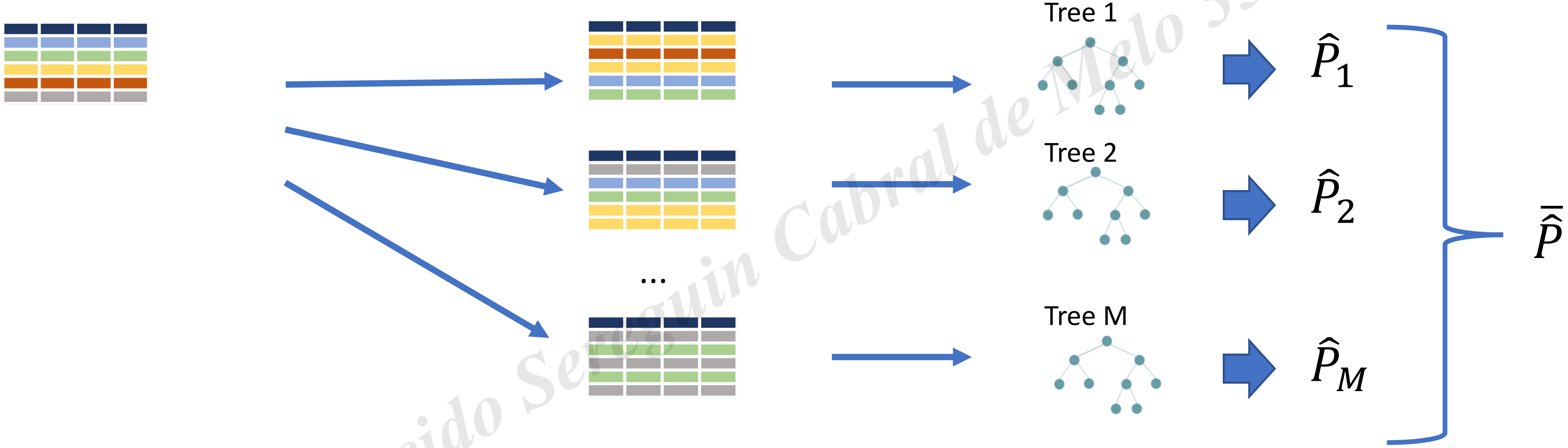
3) We repeat this M times (let's say... M=10,000 times)
4) We can calculate the average and standard error of the estimator

Bootstrap – aggregation (bagging)



Bagging is an *aggregation* of the same *algorithm* in *bootstrap* samples

Bootstrap – aggregation (bagging)



Bagging with trees is the famous *Random Forest*

Random Forest

Bagging and *Pasting*

Bagging

1. Remove a random sample **with** the replacement of size N
2. Build the model in this sample
3. Repeat 1 and 2 M times

Pasting

1. </g>Remove a random sample <g id="1" ctype="x-bold;" equiv-text="&lt;run1&gt;">WITHOUT reposition of size Q&lt;N
2. Build the model in this sample
3. Repeat 1 and 2 M times

The most famous *bagging* is *Random Forest*, which is made with trees, hence the name.

Characteristics

Bagging

1. Parallel wheel
2. It also classifies in parallel
3. It usually has good performance without great adjustments

If it were a car, I would say that it is a GMC Hummer H3.

Questions that I had when I learned it.

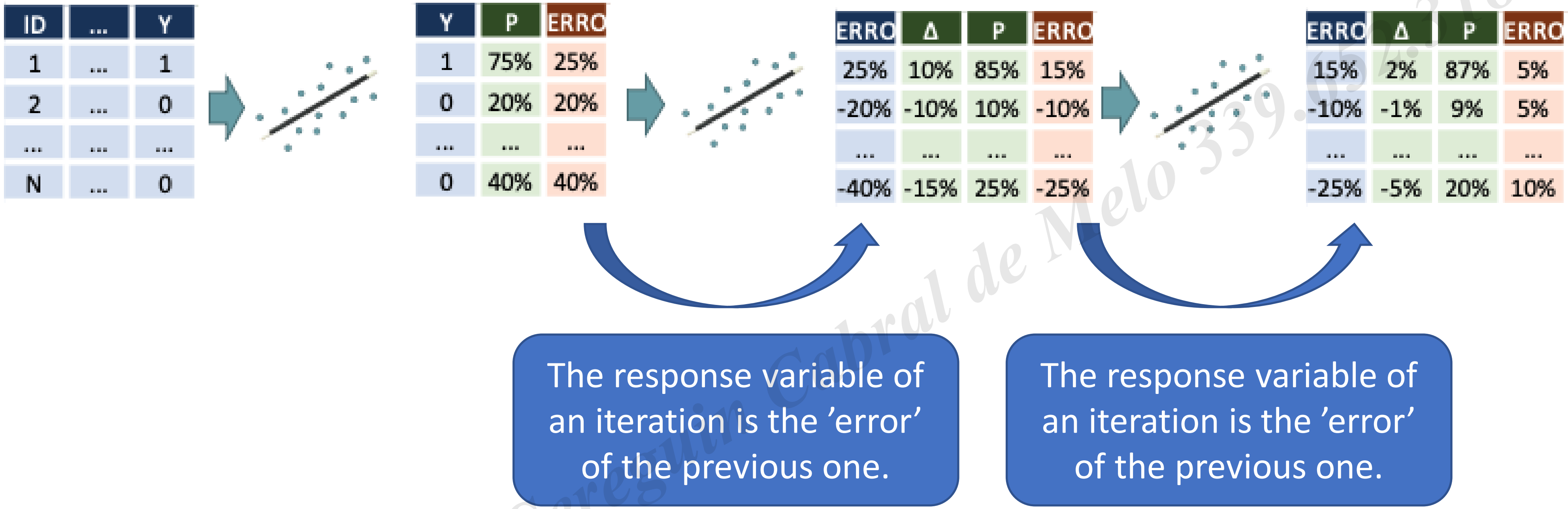
Random Forest

1. Is performing 500 trees the *default?*
2. Does it take loads of time to train?
3. And to apply the rule? Do I have to apply all of this rules? Does it take a long time?
4. Does the algorithm keep all of these trees?

If it were a car, I would say that it is a GMC Hummer H3.

Boosting

Sequential correction of errors



ID	...	Y
1	...	1
2	...	0
...	...	...
N	...	0

Y	P	ERRO
1	75%	25%
0	20%	20%
...	...	...
0	40%	40%

ERRO	Δ	P	ERRO
25%	10%	85%	15%
-20%	-10%	10%	-10%
...	...	...	...
-40%	-15%	25%	-25%

ERRO	Δ	P	ERRO
15%	2%	87%	5%
-10%	-1%	9%	5%
...	...	...	...
-25%	-5%	20%	10%

The response variable of an iteration is the 'error' of the previous one.

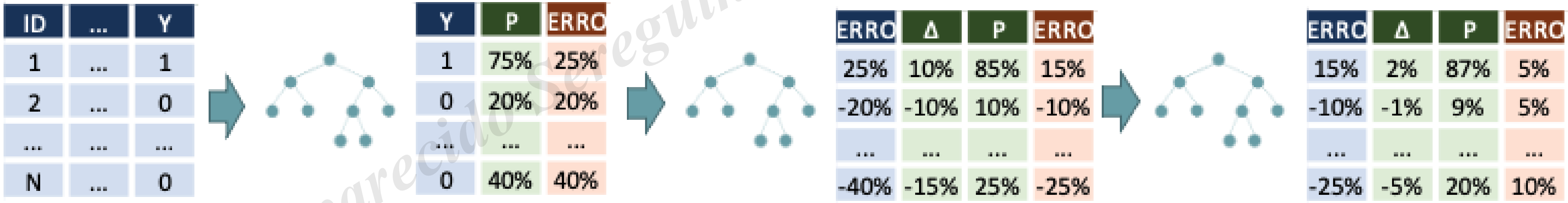
The response variable of an iteration is the 'error' of the previous one.

Boosting

- *Boosting* methods are sequential models that try to improve the error of the previous model

Gradient Boosting

- *Gradient Boosting* is a variation based on trees with some hyperparameters that control the algorithm



ID	...	Y
1	...	1
2	...	0
...	...	...
N	...	0

Y	P	ERRO
1	75%	25%
0	20%	20%
...	...	...
0	40%	40%

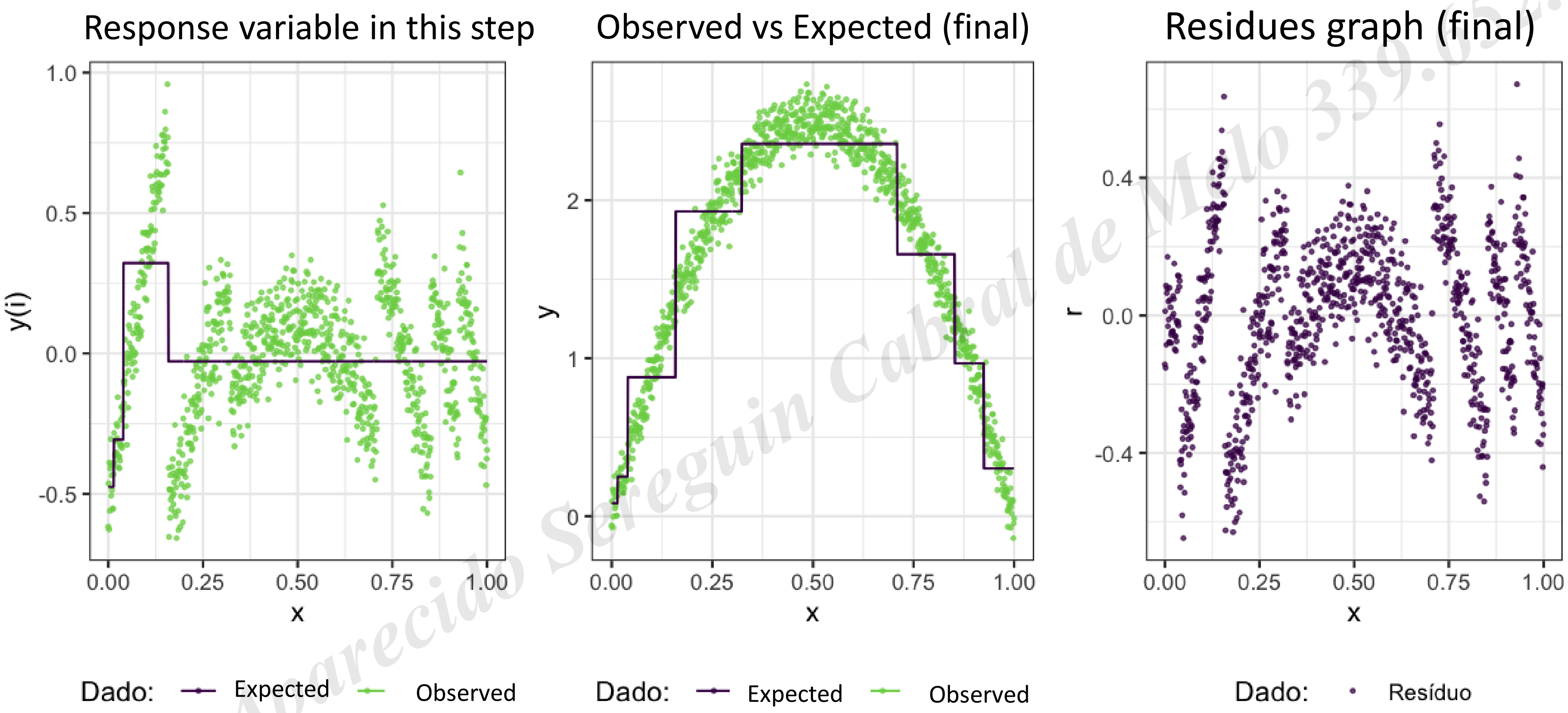
ERRO	Δ	P	ERRO
25%	10%	85%	15%
-20%	-10%	10%	-10%
...	...	...	...
-40%	-15%	25%	-25%

ERRO	Δ	P	ERRO
15%	2%	87%	5%
-10%	-1%	9%	5%
...	...	...	...
-25%	-5%	20%	10%

Learning rate

"Extend the string too much and it breaks, let it very loose, and the instrument can not be played"

Learning rate



Response variable in this step

Observed vs Expected (final)

Residues graph (final)

Learning Rate decreases the impact of each iteration it usually requires more iterations

but it helps to achieve better results

XGBoosting

- Short name for Extreme Gradient Boosting
- It is an implementation of Gradient Boosting
- It has interfaces for R and Python
- It got famous for being used by competition winners
- Created by Tianqi Chen

What do I do with my new superpowers?

- Practice suggestions in addition to the class:
  - Try to classify human activity by accelerometer and gyroscope of cellphones https://archive.ics.uci.edu/ml/datasets/human+activity+recognition+using+smartphones
  - Identify heart disease https://archive.ics.uci.edu/ml/datasets/Heart+Disease

Conclusions

- Trees are only the beginning
- There are INFINITE ways of combining models, these are the most famous ones
- These models are difficult to interpret
- *Cross-validation* replaces the *stepwise*
- *PRACTICE!*

That's it for today ;)

linkedin.com/in/joao-serrajordia