CLASSIFICATION TREES

INTRODUCTION TO THE ALGORITHM

CLASSIFICATION TREES ARE ALGORITHMS THAT ARE USED TO CLASSIFY THE DATA OF A SPECIFIC DATASET INTO CATEGORIES, AND WE OFTEN USE THEM TO MAKE PREDICTIONS. THEY DIVIDE THE DATA INTO CATEGORIES BASED ON THEIR FEATURES, BUT THEY NEED TO BE TRAINED, THEREFORE THERE IS AN ADDITIONAL ALGORITHM THAT ALLOWS YOU TO BUILD THE TREE IN ORDER TO EFFECTIVELY DIVIDE THE DATASET.
THIS IS A VERY POWERFUL ALGORITHM BECAUSE IT ALLOWS YOU TO USE DIFFERENT TYPES OF FEATURES AT SAME TIME, IT MANAGES VERY WELL ALSO MISSING FEATURES LIKE WE WILL SEE LATER. HOWEVER, THIS ALGORITHM IS NOT PERFECT, IT HAS SOME CRITICAL ISSUES SUCH AS THE FACT THAT IT TENDS TO OVERFIT THE TRAINING DATASET AND IS DIFFICULT TO USE WITH LARGE DATASETS.

LET'S START

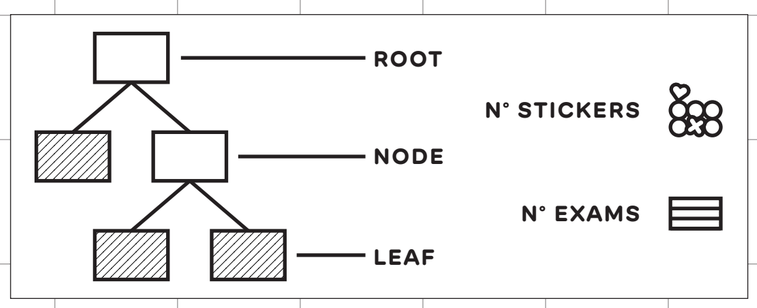
Today is a good day: you woke up early and decided to go to the Politecnico library in order to be productive.
However, the library is very crowded and you don't know where to sit, but one thing is for sure: **you will not sit next to anyone who is not from your course**. We are confident that with the number of the stickers students have and the exams that they have yet to take may help us classify them. Here classification trees come to the rescue.

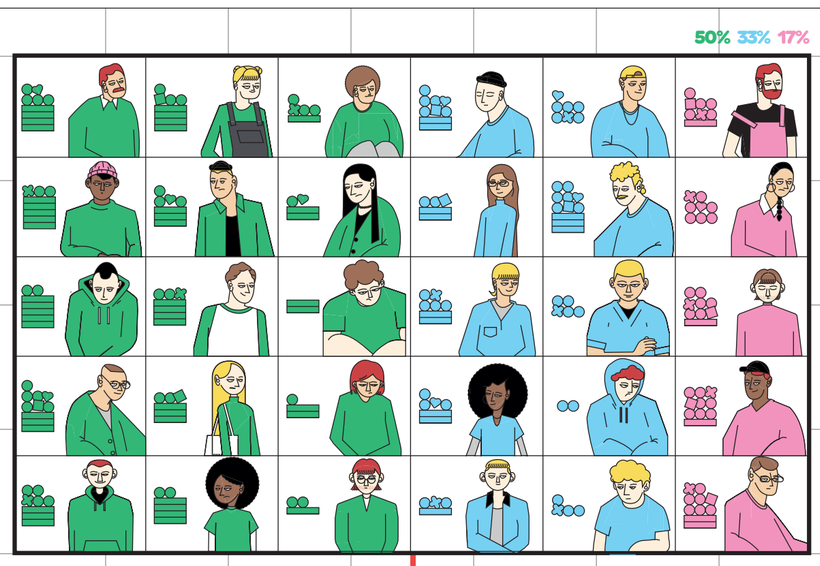
THE DATA! 1

As we said before, we have to **train the algorithm**, so we think about a sample of our friends and gather their data on a table. Since we are only considering **two features** and they are both numerical, we can represent them on a cartesian plane with the n° of stickers on the x-axis and the n° of exams on the y-axis.

S	7	0	1	2	2	2	2	3	3	3	4	4	4	5	5	8	2	3	3	4	4	5	5	7	7	7	8	8	8	9
E	3	2	2	1	2	4	5	3	4	5	1	3	4	3	4	3	0	1	2	0	2	0	2	0	2	1	0	1	2	1

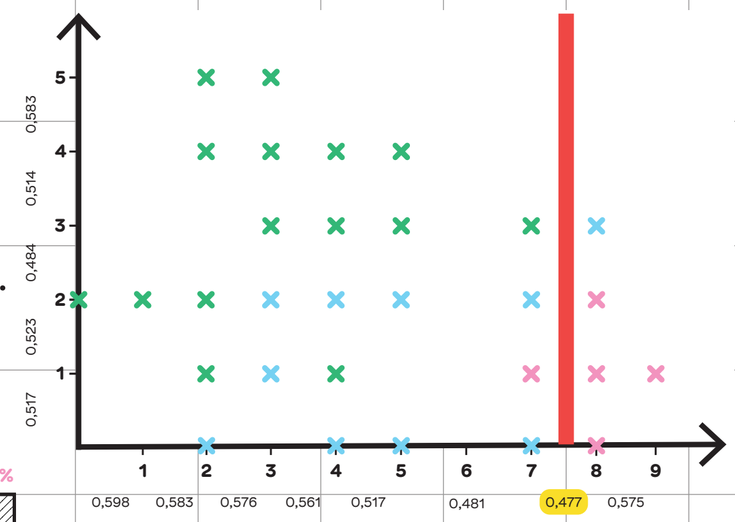
ENGINEERS ARCHITECTS DESIGNERS





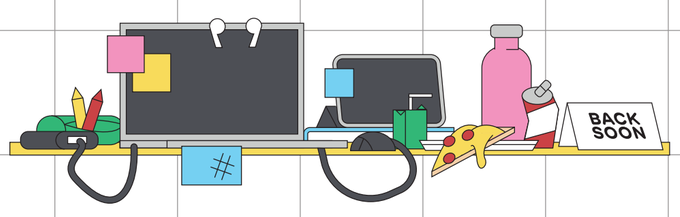
LET'S VISUALIZE THE THING! 3

In the cartesian plane we visualize how we make the first choice. Moving the cursors (there are two of them, but they are independent from each other) you'll discover the heterogeneity index associated to the division they are indicating.



ASK THE RIGHT QUESTION! 2

To achieve the right answer, we need to ask the right question, or rather the one that most efficiently splits the dataset. To do so we could use various methods: in this case we will use the **Gini impurity index**, a number from 0 to 1 that indicates the purity of the two halves made by the split. Once we have the two indices we need to calculate their **weighted average**, We call this number **heterogeneity index**. We repeat the process for each possible split, and then choose the one with the lowest result to start from. This process will give us the **smallest tree possible**.

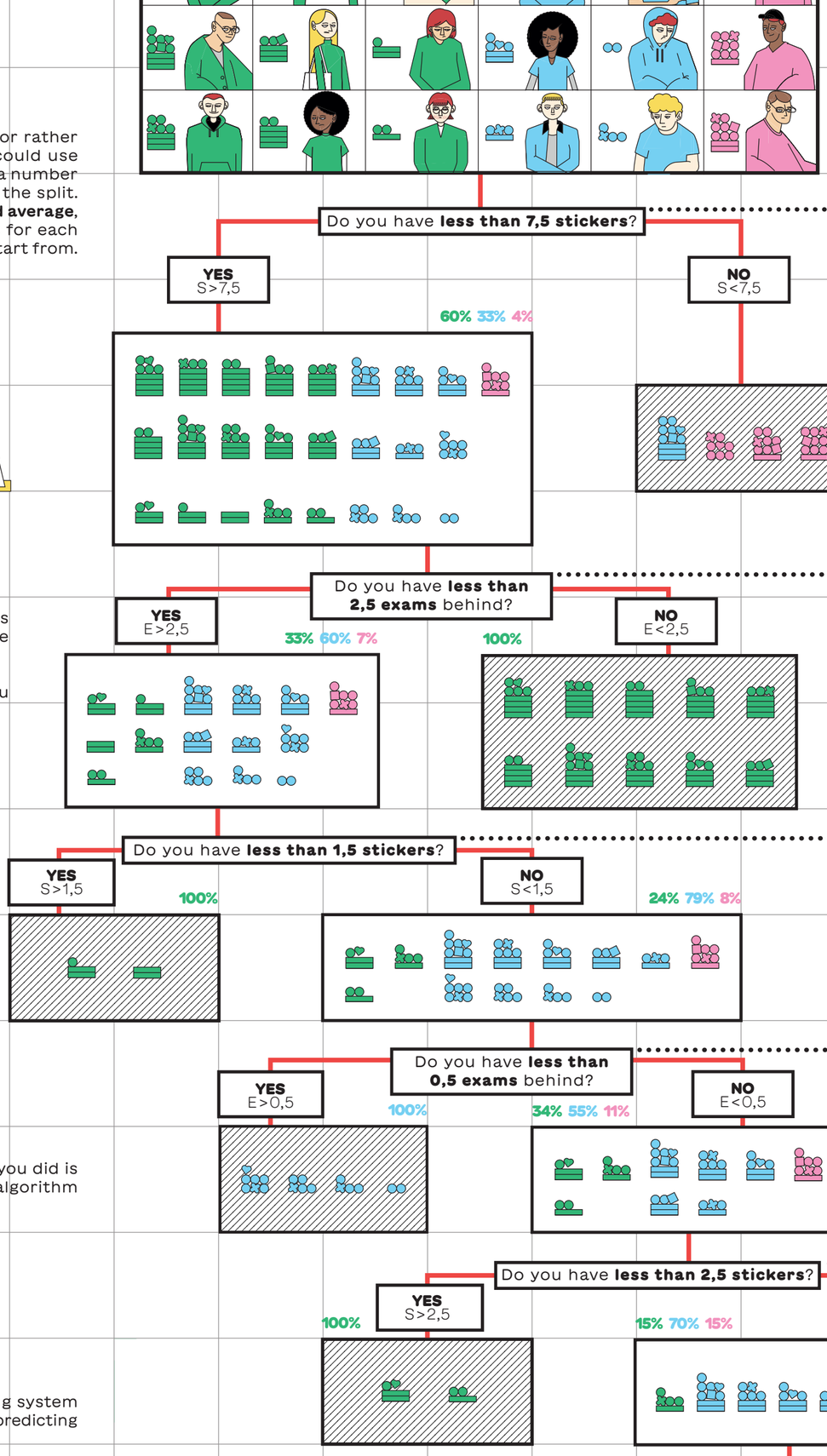


AGAIN AND AGAIN 4

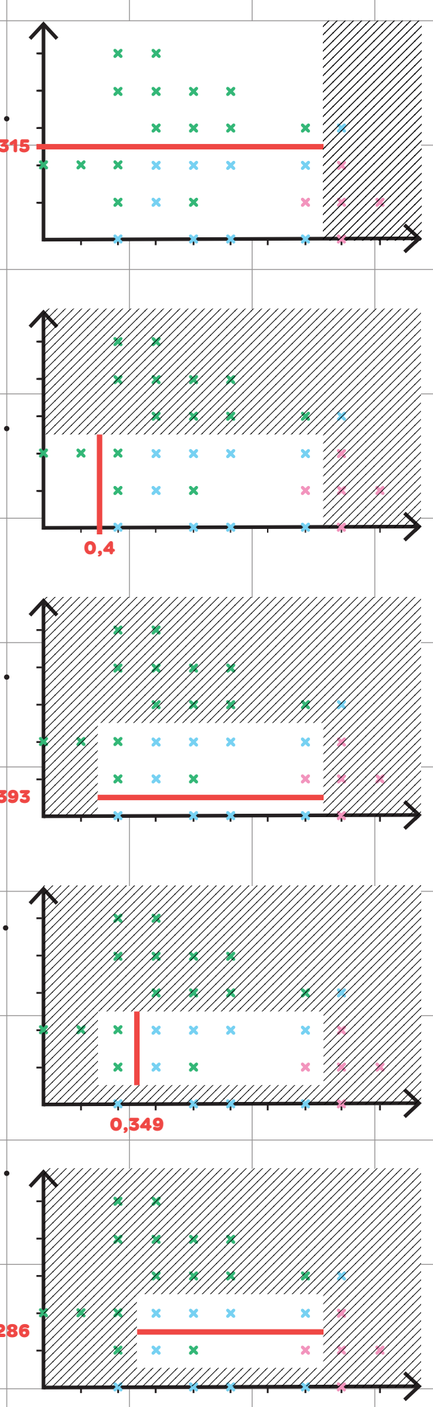
Now, you repeat the process for each half. Again. And again. Unless you got to a **homogeneous leaf**. In this case, your job there is done and you can only consider the remaining half.

The graphs here show the most efficient split we identified, as you did by yourself with the slider above.

MISSING INFORMATION?
But what if you can't see how many stickers someone has? **You can still classify them**, that's the beauty of classification trees! Indeed each branch reports the probability of occurrence of each class. It might not be the most accurate, but it offers a **quick solution** that you can base your decisionon.
Empty places get occupied quickly. **Hurry up**!



HEY! HAVE YOU NOTICED THIS?
This leaf is not homogenous because we applied **pre-pruning**, a process we use to **avoid unnecessary complexity**. It requires to define in advance the criteria that, if unsatisfied, will stop the division. In our case the criteria was: keep splitting if you have at least six records in a branch.



TASK COMPLETED, LET'S TEST IT! 5

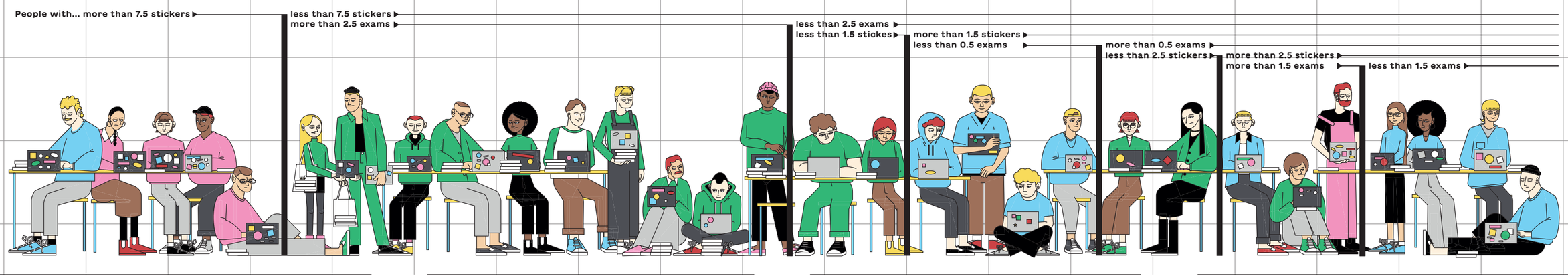
Once we completed the tree it's good practice to check if what you did is working. Let's think of four more of your friends and **test** if the algorithm you just built categorizes them well.

S	2	2	7	5	3	6	3
E	2	1	1	2	2	1	3

Oh no, the algorithm got them wrong. **So inefficient!**
This is caused by **overfitting**, a phenomenon in which the learning system fits the given training data so tightly that it would be inaccurate in predicting the outcomes of new data, **slide to discover the solution**.

HAPPILY EVER AFTER 6

Now the algorithm is good to go and you can use it to classify people in the library. Here is a visualization to **recap the whole process**, step by step, following the order each person got sorted into their final category.



VISUAL EXPLANATIONS OF STATISTICAL METHODS

Classification Trees

AUTHORS

Giulio Alessandrini
Alexandra Chiojdeanu
Andrea Corsini
Greta Cozza
Miguel Gashi
Alessia Mattesini
Ana Muço

FACULTY

Michele Mauri
Ángeles Briones
Gabriele Colombo
Simone Vantini
Salvatore Zingale

TEACHING ASSISTANTS

Elena Aversa
Andrea Benedetti
Tommaso Elli
Beatrice Gobbo
Arianna Bellantuono

Final Synthesis Design Studio
Sect. C3

LM in Communication Design
A.A. 2022/2023

DEN-SITY GN+

POLITECNICO MILANO 1863
SCHOOL OF DESIGN