

Final Project

Cristian Camilo Zapata García 202010104010

Harold Steven Gonzales Ossa 202010016101

Julián Andrés Mazo Zapata 202010110010

| Estructuras de datos y algoritmos 1 |

# **Doubts of the members**

* What would be the most efficient method for reading data, CSV or pandas?
* What csv files should all be read?
* Under what criteria do we define the data?

**Load the dataset**

**Second part**

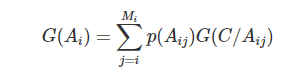
**Main algorithm – CART Algorithm.**

The main algorithm that we are going to use to do the data analysis is the one called CART (Classification And Regression Trees) which uses the “Gini impurity” method (not Gini coefficient) to be able to learn based on decision trees.

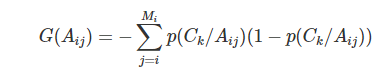
-The classification trees predict categories of objects

-The regression trees predict values continuous.

The CART algorithm generates binary decision trees (Which means that every node is divided in two branches), in each iteration the variable is selected predictive and the breakpoint that best reduce ‘impurity’; it uses the Gini index to calculate the measure of impurity



Being G(C/Aij) equals to



\*Aij is the attribute use to branch the tree

\*J is the number of classes

\*Mi are the different values that the attribute Ai has

\*p(Aij) constitutes the probability that A takes its j-th value

\*p(Ck/Aij) represents the probability that an example is of class Ck when its attribute Ai takes its j-th value.

**Alternative #2 – CHAID Algorithm.**

Another choice that could be implemented instead of CART algorithm is CHAID tree decision algorithm. This one is the oldest decision tree algorithm in the history. It uses the chi-square metric to find significance of a feature (A higher value means a higher significance). It works for classification problems, so it would be useful for this project.

The formula of chi-square testing is:

Where the actual value means times that a decision was taken, and the expected value means times that a decision is expected to be took. The formula of expected value for any statement is:

The root of the decision tree will be the statement that have the most chi-square value. Then, the data will be sorted and rearranged accord the specifications of the root, and the branches will be the statements with the most chi-square value of the new arrange of data. When a branch just arrives to a definitive decision (example, in the case that a statement just have a yes decision or a not decision), we stop making more branches for that branch (becoming a leaf).

When we done arriving to leaves, the tree will be finished and ready for make the data classification.

**Alternative #3 – C4.5 Algorithm.**

Algorithm C4.5 is our third option, C4.5 is an extension of ID3, it is an algorithm applied to generate decision trees, it can be employed for classification, an interesting factor to take into account.C4.5 build trees using the concept of information entropy. At each node in the tree, C4.5 chooses an attribute of the data that most efficiently divides the set of samples into subsets enriched in one class or another. Its criterion is the normalized one for information gain (entropy difference) that results in the selection of an attribute to divide the data. The attribute with the highest normalized information gain is preferred as the decision parameter. The C4.5 algorithm recursively divides into smaller sub lists. This algorithm has a few base cases which are:

* All the samples in the list belong to the same class. When this happens, you just create a leaf node for the decision tree by saying to choose that class.
* Neither feature provides any information gain. In this case, C4.5 creates a decision node higher up the tree using the expected value of the class.
* Previously unseen class instance found. Again, C4.5 creates a decision node higher up the tree with the expected value.

References:

-https://sefiks.com/2018/08/27/a-step-by-step-cart-decision-tree-example/

-https://sefiks.com/2020/03/18/a-step-by-step-chaid-decision-tree-example/

-https://sefiks.com/2018/05/13/a-step-by-step-c4-5-decision-tree-example/