Machine Learning

Decision Tree & Random Forest

March Brag

cesarbrma91@gmail.com

@MarchBragagnini

Universidad Católica San Pablo

Centro de Investigación e Innovación en Ciencia Computación

Outline

1. Introduction.
2. Decision Tree.
3. Decision Tree Example.
4. Decision Tree Learning Algorithm.
5. Metrics for Decision Tree.
6. Gini Purity
7. Advantages & When to Use Decision Trees.
8. Code.
9. References.

Introduction

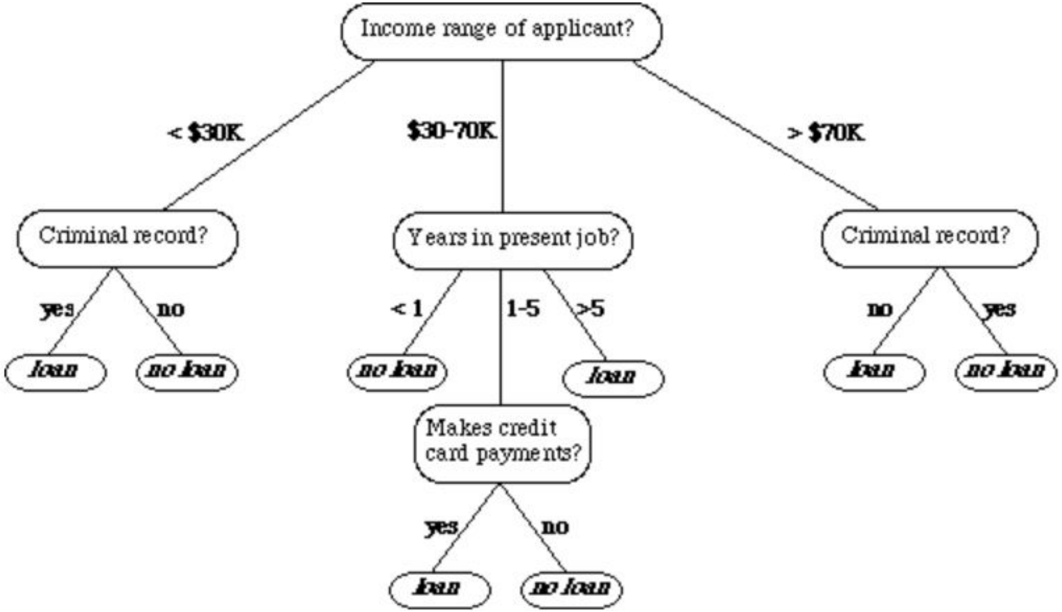
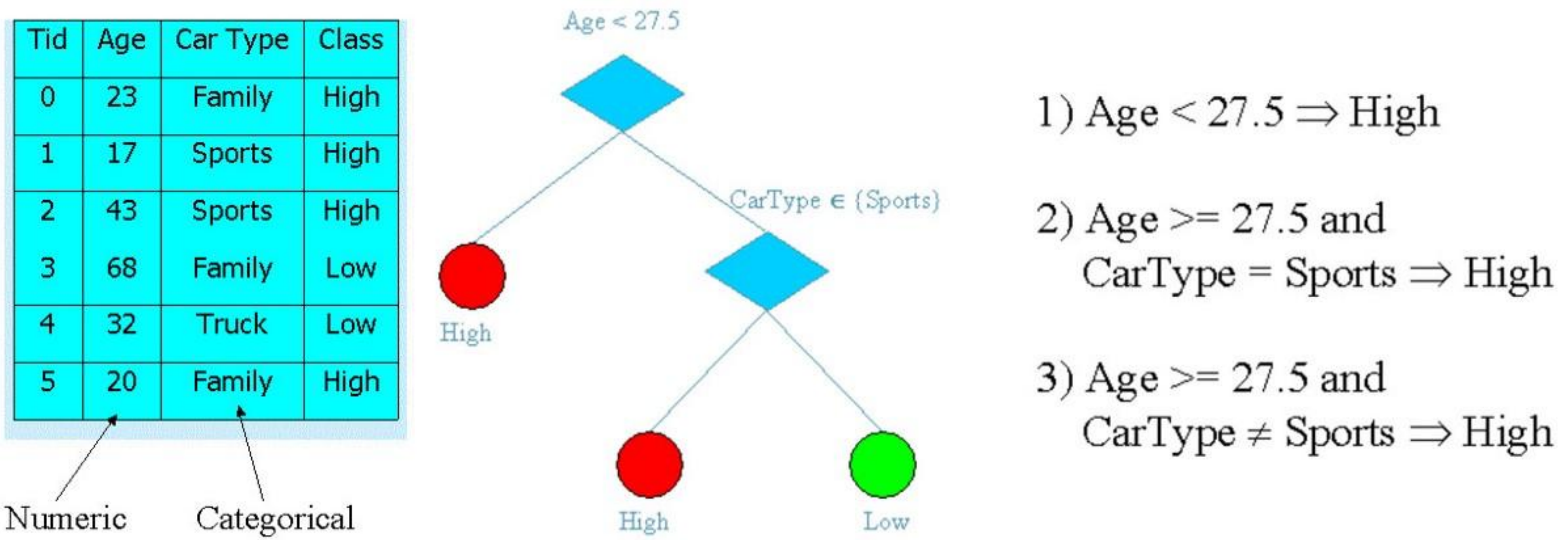


Figure 01 : If we are classifying *bank loan* application for a customer, the decision tree may look like this

[ ] https://medium.com/deep-math-machine-learning-ai/chapter-4-decision-trees-algorithms-b93975f7a1f1

Introduction



Tid	Age	Car Type	Class
0	23	Family	High
1	17	Sports	High
2	43	Sports	High
3	68	Family	Low
4	32	Truck	Low
5	20	Family	High

Figure 03: If we are classifying *bank loan* application for a customer, the decision tree may look like this

[ ] https://medium.com/deep-math-machine-learning-ai/chapter-4-decision-trees-algorithms-b93975f7a1f1

Decision Tree

A decision tree uses a tree structure to represent a number of possible decision paths and an outcome for each path.

- C: “I am thinking of an animal”
- P: Does it have more than five legs ?
- C: No
- P: Is it delicious ?
- C: No
- P: Does it appear on the back of the Australian five-cent coin ?
- C: Yes
- D: Is it an echidna ?
- C: Yes, it is !

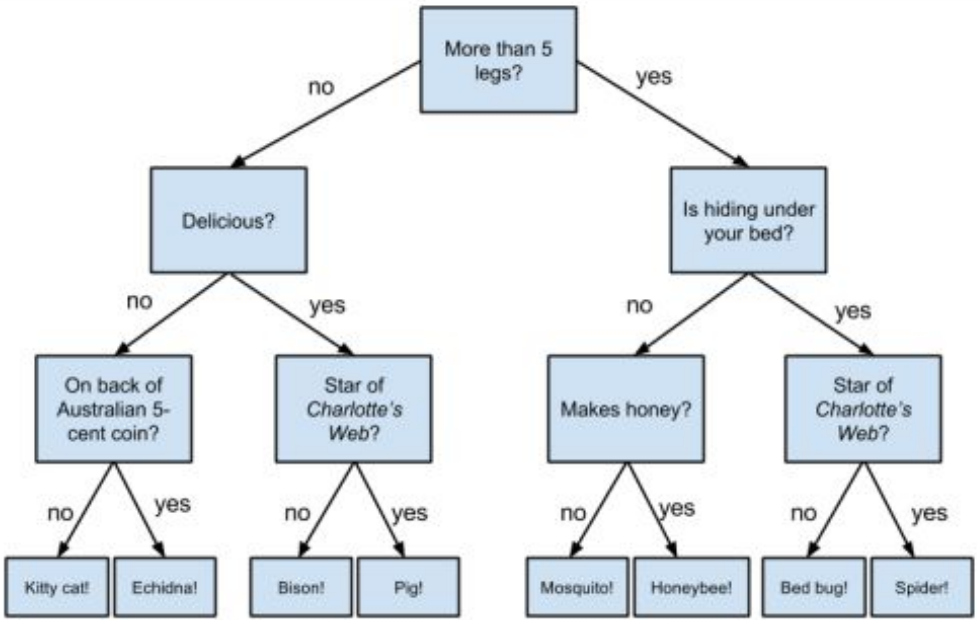


Figure 02: Example of Decision Tree.

[ ] Data Science from Scratch First Principles with Python, Joel Grus

Decision Tree

1. **Alternate**: whether there is a suitable alternative restaurant nearby.
2. **Bar**: whether the restaurant has a comfortable bar area to wait in.
3. **Fri/Sat**: true on Fridays and Saturdays.
4. **Hungry**: whether we are hungry.
5. **Patrons**: how many people are in the restaurant (values are None, Some, and Full).
6. **Price**: the restaurant's price range ($, $$, $$$).
7. **Raining**: whether it is raining outside.
8. **Reservation**: whether we made a reservation.
9. **Type**: the kind of restaurant (French, Italian, Thai, or burger).
10. **WaitEstimate**: the wait estimated by the host (0-10 minutes, 10-30, 30-60, >60).

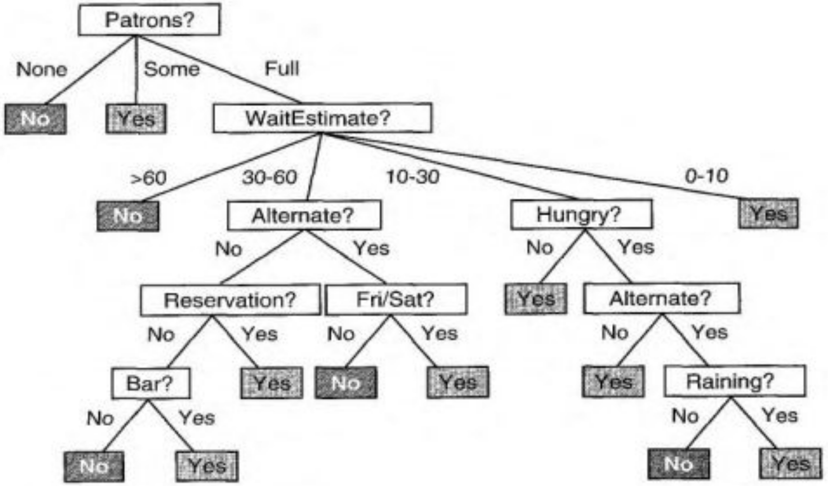


Figure 04: A decision tree for deciding whether to wait for a table.

[ ] Artificial Intelligence: A Modern Approach - Russel, Norvig

Decision Tree Example

Example	Attributes										Goal
	Alt	*Bar*	*Fri*	*Hun*	*Pat*	*Price*	*Rain*	*Res*	*Type*	*Est*	*WillWait*
X_1	*Yes*	*No*	*No*	*Yes*	*Some*	*$$$*	*No*	*Yes*	*French*	*0–10*	*Yes*
X_2	*Yes*	*No*	*No*	*Yes*	*Full*	*$*	*No*	*No*	*Thai*	*3040*	*No*
X_3	*No*	*Yes*	*No*	*No*	*Some*	*$*	*No*	*No*	*Burger*	*0–10*	*Yes*
X_4	*Yes*	*No*	*Yes*	*Yes*	*Full*	*$*	*Yes*	*No*	*Thai*	*10–30*	*Yes*
X_5	*Yes*	*No*	*Yes*	*No*	*Full*	*$$$*	*No*	*Yes*	*French*	*>60*	*No*
X_6	*No*	*Yes*	*No*	*Yes*	*Some*	*$$*	*Yes*	*Yes*	*Italian*	*0–10*	*Yes*
X_7	*No*	*Yes*	*No*	*No*	*None*	*$*	*Yes*	*No*	*Burger*	*0–10*	*No*
X_8	*No*	*No*	*No*	*Yes*	*Some*	*$$*	*Yes*	*Yes*	*Thai*	*0–10*	*Yes*
X_9	*No*	*Yes*	*Yes*	*No*	*Full*	*$*	*Yes*	*No*	*Burger*	*>60*	*No*
X_{10}	*Yes*	*Yes*	*Yes*	*Yes*	*Full*	*$$$*	*No*	*Yes*	*Italian*	*10–30*	*No*
X_{11}	*No*	*No*	*No*	*No*	*None*	*$*	*No*	*No*	*Thai*	*0–10*	*No*
X_{12}	*Yes*	*Yes*	*Yes*	*Yes*	*Full*	*$*	*No*	*No*	*Burger*	*30–60*	*Yes*

Table 01: Examples for the restaurant domain. The positive examples are the ones in which the goal WillWait is true (X1,X3,...); the negative examples are the ones in which it is false (X2,X5 ,...)

[ ] Artificial Intelligence: A Modern Approach - Russel, Norvig

Decision Tree Example

Is a good or a bad decision tree?

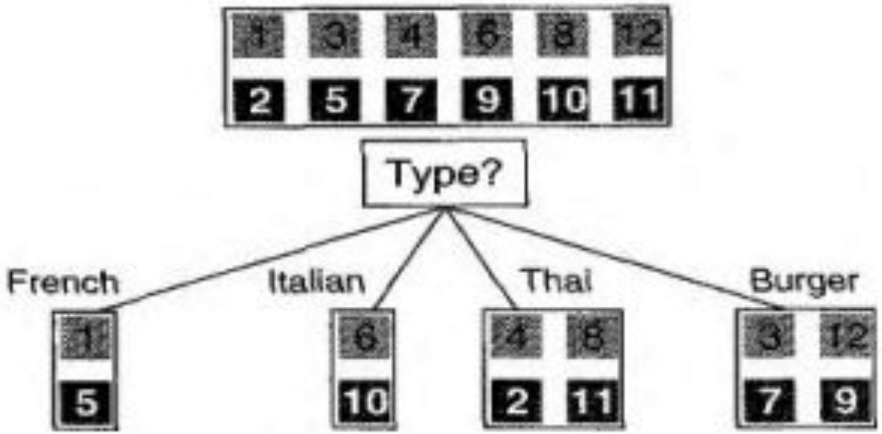


Figure 04: A part of decision tree for deciding whether to wait for a table.

[ ] Artificial Intelligence: A Modern Approach - Russel, Norvig

Decision Tree Example

Is a good or a bad decision tree?

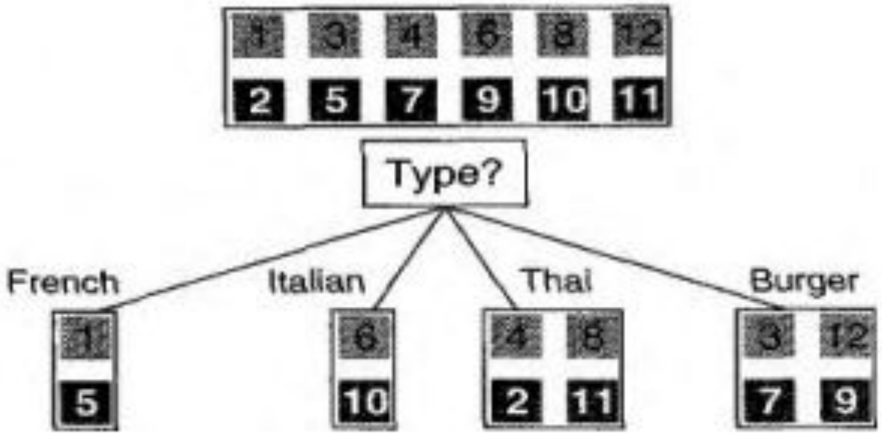


Figure 04: A part of decision tree for deciding whether to wait for a table.

It is bad split, obviously.

[ ] Artificial Intelligence: A Modern Approach - Russel, Norvig

Decision Tree Example

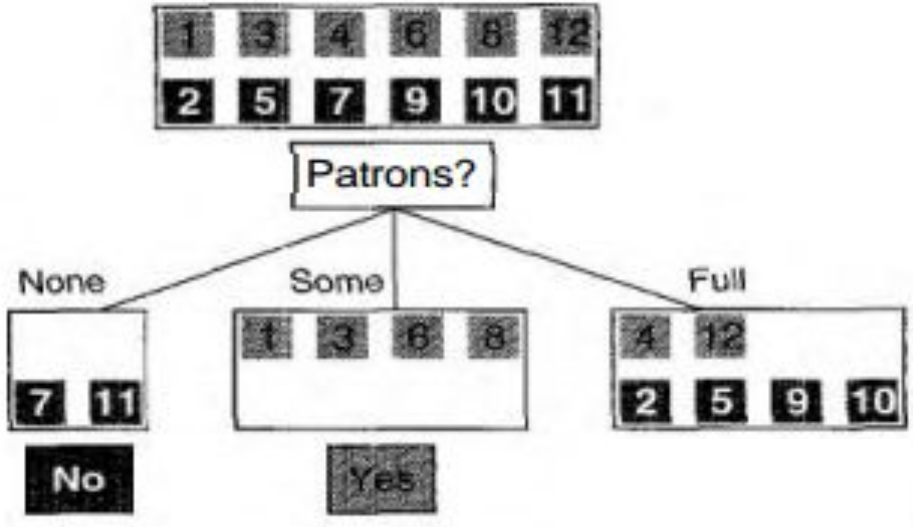


Figure 05 (a): A part of other decision tree for deciding whether to wait for a table.

[ ] **Artificial Intelligence: A Modern Approach - Russel, Norvig**

Decision Tree Example

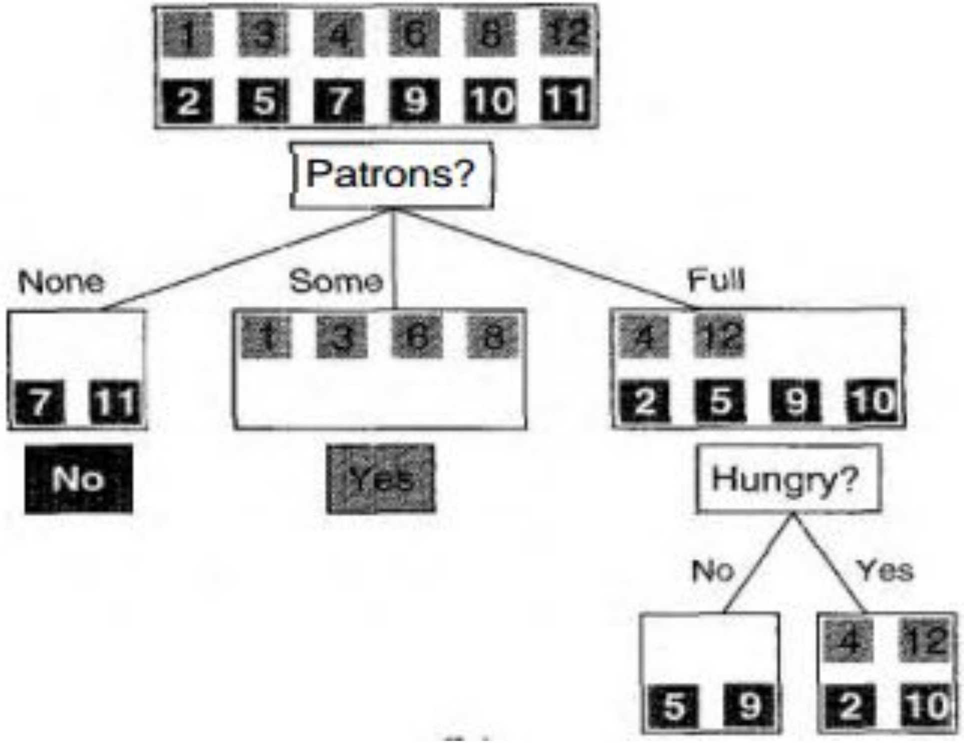


Figure 05 (b): A part of other decision tree for deciding whether to wait for a table.

[ ] Artificial Intelligence: A Modern Approach - Russel, Norvig

Decision Tree Example

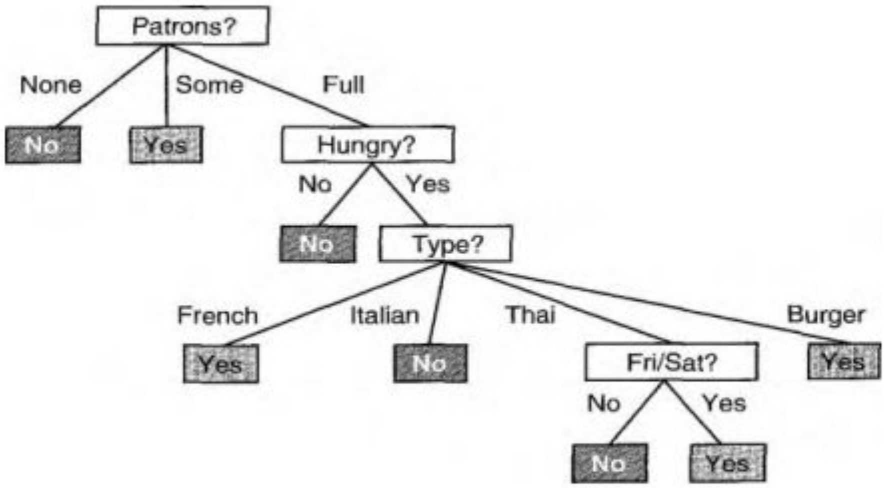


Figure 05 (c): A good decision tree for deciding whether to wait for a table.

[ ] Artificial Intelligence: A Modern Approach - Russel, Norvig

Decision Tree Learning Algorithm

function DECISION-TREE-LEARNING(*examples*, *attribs*, *default*) **returns** a decision tree
  inputs: *examples*, set of examples
      attrzbs, set of attributes
      default, default value for the goal predicate

  if *examples* is empty **then return** *default*
  else if all *examples* have the same classification **then return** the classification
  else if *attrzbs* is empty **then return** MAJORITY-VALUE(*examples*)
  else
    best ← CHOOSE-ATTRIBUTE(*attribs*, *examples*)
    tree ← a new decision tree with root test *best*
    m ← MAJORITY-VALUE(*examples*)
    for each value v_i of *best* **do**
      $examples_i$ ← {elements of *examples* with *best* $= v_i$}
      subtree ← DECISION-TREE-LEARNING($examples_i$, *attribs* − *best*, *m*)
      add a branch to *tree* with label v_i and subtree *subtree*
    return *tree*

Algorithm 01: Decision Tree Learning Algorithm.

[ ] Artificial Intelligence: A Modern Approach - Russel, Norvig

Metrics for Decision Tree

1 Gini Purity -> CART(Classification and Regression Trees)

2 Entropy Function and Information Gain -> ID3(Iterative Dichotomiser 3)

[ ] https://medium.com/deep-math-machine-learning-ai/chapter-4-decision-trees-algorithms-b93975f7a1f1

Gini Purity

The goal in building a decision tree is to create the smallest possible tree in which each leaf node contains training data from only one class. In evaluating possible splits, it is useful to have a way of measuring the purity of a node. **The purity d**escribes how close the node is to containing data from only one class

$$\phi(\mathbf{p}) = \sum_i p_i(1 - p_i)$$

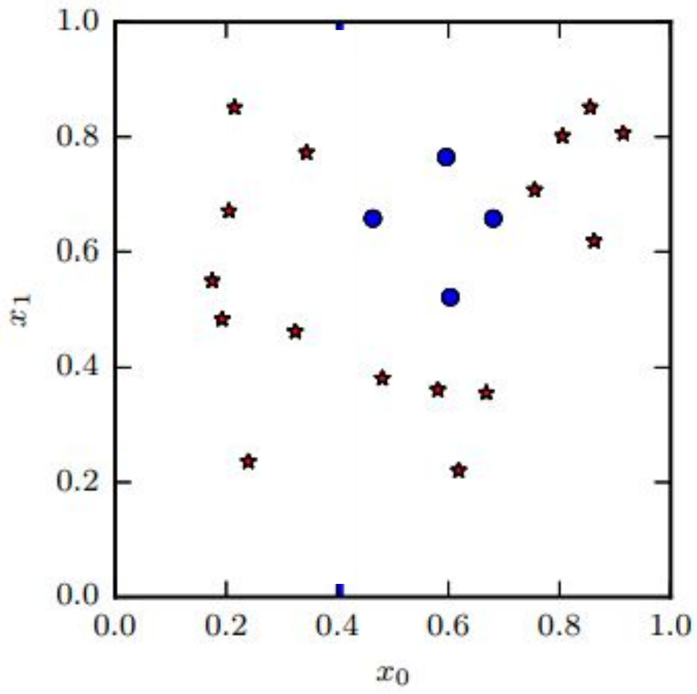
The decision tree construction algorithm proceeds by recursively splitting the training data

$$\Theta(s,t) = \phi(\mathbf{p}) - P_L\phi(\mathbf{p_L}) - P_R\phi(\mathbf{p_R})$$

[ ] https://w3.cs.jmu.edu/spragunr/CS444/activities/trees_bagging/decision_trees.pdf

Gini Purity

1. Calculate the Gini purity.
2. Execute the recursive tree construction algorithm.
3. Classify the following three points using your decisic tree.
   a. (0.4, 1.0)
   b. (0.6, 1.0)
   c. (0.6, 0.0)



[ ] https://w3.cs.jmu.edu/spragunr/CS444/activities/trees_bagging/decision_trees.pdf

Gini Purity

1. **Calculate the Gini purity.**

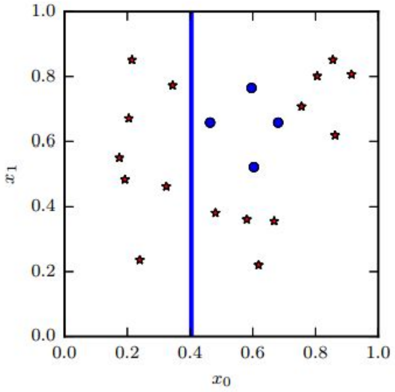
$$\phi(\mathbf{p}) = .2 \times .8 + .8 \times .2 = .32$$

[ ] https://w3.cs.jmu.edu/spragunr/CS444/activities/trees_bagging/decision_trees.pdf

Gini Purity

2. Execute the recursive tree construction algorithm.

Split 1



The impurity on the left is:

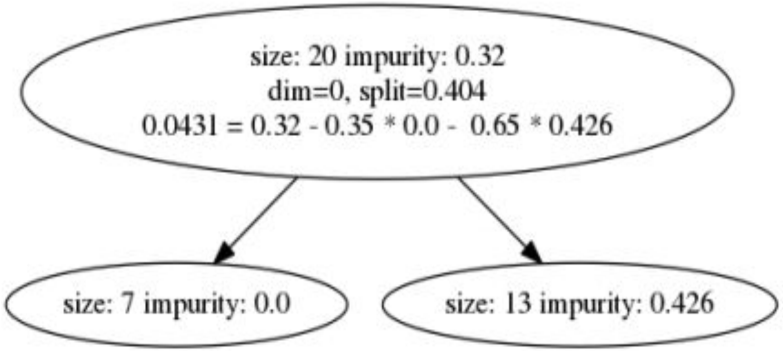
$\phi(\mathbf{p_L}) = 0 \times 1 + 1 \times 0 = 0$

The impurity on the right is:

$\phi(\mathbf{p_R}) = 4/13 \times 9/13 + 9/13 \times 4/13 = .426$

This makes the goodness-of-split:

$$\begin{aligned}\Theta(s,t) &= .32 - P_L \times 0 + P_R \times .426 \\ &= .32 - .35 \times 0 + .65 \times .426 = .0431\end{aligned}$$

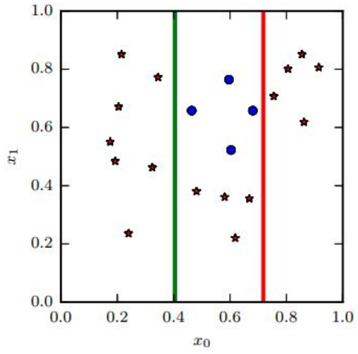


[ ] https://w3.cs.jmu.edu/spragunr/CS444/activities/trees_bagging/decision_trees.pdf

Gini Purity

2. Execute the recursive tree construction algorithm.

Split 2



The impurity on the left is:

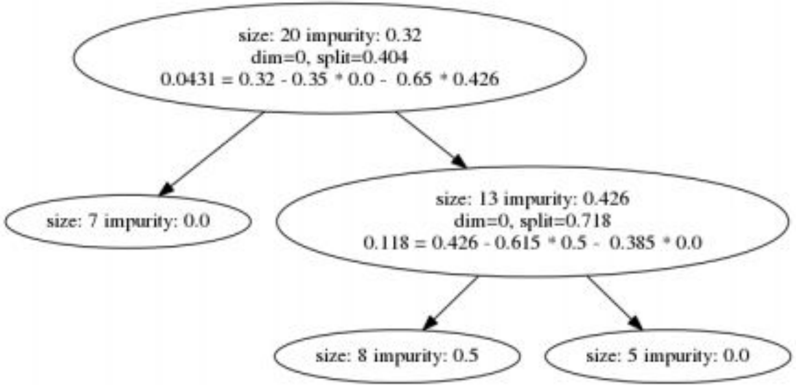
$\phi(\mathbf{p_L}) = 4/8 \times 4/8 + 4/8 \times 4/8 = .5$

The impurity on the right is:

$\phi(\mathbf{p_R}) = 0 \times 1 + 1 \times 0 = 0$

This makes the goodness-of-split:

$\Theta(s, t) = .426 - 8/13 \times .5 + 5/13 \times 0 = .118$

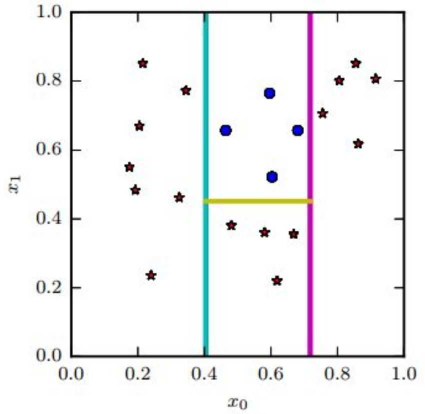


[ ] https://w3.cs.jmu.edu/spragunr/CS444/activities/trees_bagging/decision_trees.pdf

Gini Purity

2. Execute the recursive tree construction algorithm.

Split 3



The impurity on the left is:

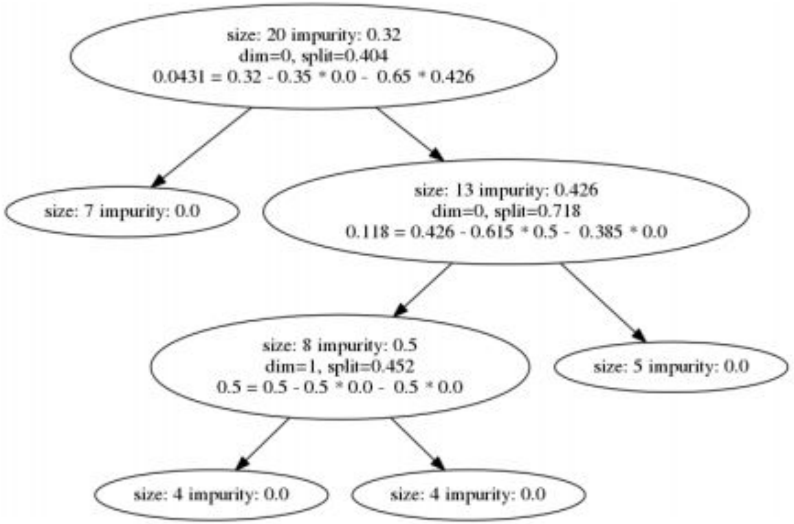
$\phi(\mathbf{p_L}) = 1 \times 0 + 0 \times 1 = 0$

The impurity on the right is:

$\phi(\mathbf{p_R}) = 0 \times 1 + 1 \times 0 = 0$

This makes the goodness-of-split:

$$\Theta(s,t) = .5 - 4/8 \times 0 + 4/8 \times 0 = .5$$



[ ] https://w3.cs.jmu.edu/spragunr/CS444/activities/trees_bagging/decision_trees.pdf

Advantages to use Decision Trees

Advantages

- Decision trees often mimic the human level thinking.
- Decision trees actually make you see the logic for the data to interpret (not like black box algorithms).
- The decision tree representation seems to be very natural for humans; indeed, many "How To" manuals (e.g., for car repair).

[ ] Chapter 4: Decision Trees Algorithms - Medium
[ ] Artificial Intelligence: A Modern Approach - Russel, Norvig

Random Forest

Random forest are an example of an ensemble learner built on decision trees.

[ ] https://medium.com/deep-math-machine-learning-ai/chapter-4-decision-trees-algorithms-b93975f7a1f1

Advantages & When to Use Random Forest

Advantages

- Both training and prediction are very fast, because of the simplicity of the underlying decision trees. In addition, both tasks can be straightforwardly parallelized, because the individual trees are entirely independent entities.
- The multiple trees allow for a probabilistic classification: a majority vote among estimators gives an estimate of the probability.
- The nonparametric model is extremely flexible, and can thus perform well on tasks that are underfit by other estimators.

[ ] Python Data Science Handbook, Jake Vanderplas

Code

sklearn.tree.DecisionTreeClassifier

class sklearn.tree.**DecisionTreeClassifier**(*criterion='gini'*,...)

https://github.com/marbramen/100DaysOfMLCode/blob/master/Day%2080%20-%20Decision%20Tree%20%26%20Random%20Forest.ipynb

References

- https://medium.com/@SeattleDataGuy/what-is-a-decision-tree-algorithm-4531749d2a17
- https://medium.com/deep-math-machine-learning-ai/chapter-4-decision-trees-algorithms-b93975f7a1f1
- Artificial Intelligence: A Modern Approach - Russel, Norvig.
- https://w3.cs.jmu.edu/spragunr/CS444/activities/trees_bagging/decision_trees.pdf
- http://lsirwww.epfl.ch/courses/dis/2007ws/exercises/week13/Exercise11-Classification-Solution.pdf
- https://profs.info.uaic.ro/~ciortuz/ML.ex-book/SLIDES/ML.ex-book.SLIDES.DT.pdf

Machine Learning

Decision Tree & Random Forest

March Brag

cesarbrma91@gmail.com

@MarchBragagnini

Universidad Católica San Pablo

Centro de Investigación e Innovación en Ciencia Computación