Data-Driven Problem Solving in Mechanical Engineering

Decision Tree

Masoud Masoumi

ME 364 - Spring 2022

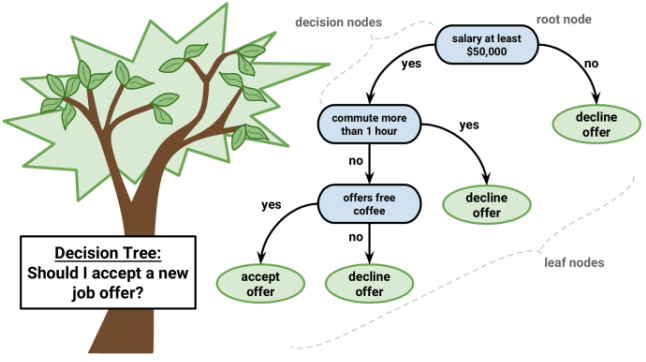
Department of Mechanical Engineering
The Cooper Union for the Advancement of Science and Art

April, 2022

Introduction

A decision tree is a flowchart-like tree structure where:

- Each internal node (decision) denotes a test on an feature
- Each branch represents an outcome of the test
- Each leaf node (or terminal node) holds a class label
- The topmost node is the root node



Ref for figure: https://medium.com/analytics-vidhya/mathematics-behind-decision-tree-73ee2ef82164

History

Late 1970s and early 1980s, J. Ross Quinlan, developed a decision tree algorithm known as ID3 (Iterative Dichotomiser)

Quinlan later presented C4.5 (a successor of ID3), which became a benchmark to which newer supervised learning algorithms are often compared.

In 1984, a group of statisticians (L. Breiman, J. Friedman, R. Olshen, and C. Stone) introduced Classification and Regression Trees (CART).

Most algorithms for decision tree including ID3, C4.5, and CART follow a top-down approach.

(a) They start with a training set and their associated class labels.

(b) The training set is recursively partitioned into smaller subsets as the tree is being built.

Algorithm

Tree is constructed in a top-down recursive divide-and-conquer manner as follows:

- At start, all the training examples are at the root

- Features are categorical (if continuous-valued, they are discretized in advance)

- Examples are partitioned recursively based on test features

- Test features are selected on the basis of a heuristic or statistical measure (e.g., **information gain**)

So when do we stop?

(a) All samples for a given node belong to the same class

(b) There are no remaining feature for further partitioning

(c) There are no samples left

Information Gain

Let p_i be the probability that an arbitrary sample in set D belongs to class C_i.

Expected information (entropy) needed to classify a sample in set D:

$$\text{info}(D) = -\sum_{i=1}^{m} p_i log_2(p_i)$$

Information needed (after using feature A to split set D into V partitions) to classify set D

$$\text{info}_A(D) = \sum_{i=1}^{v} \frac{|D_i|}{|D|} \times info(D_i)$$

Information gained by branching on feature A

$$Gain(A) = \text{info}(D) - \text{info}_A(D)$$

Example

age	income	student	credit_rating	buys_computer
youth	high	no	fair	no
youth	high	no	excellent	no
mid_age	high	no	fair	yes
senior	medium	no	fair	yes
senior	low	yes	fair	yes
senior	low	yes	excellent	no
mid_age	low	yes	excellent	yes
youth	medium	no	fair	no
youth	low	yes	fair	yes
senior	medium	yes	fair	yes
youth	medium	yes	excellent	yes
mid_age	medium	no	excellent	yes
mid_age	high	yes	fair	yes
senior	medium	no	excellent	no

$$Info(D) = I(9,5) = -\frac{9}{14}\log_2(\frac{9}{14}) - \frac{5}{14}\log_2(\frac{5}{14}) = 0.940$$

$$Info_{age}(D) = \frac{5}{14} I(2,3) + \frac{4}{14} I(4,0) + \frac{5}{14} I(3,2) = 0.694$$

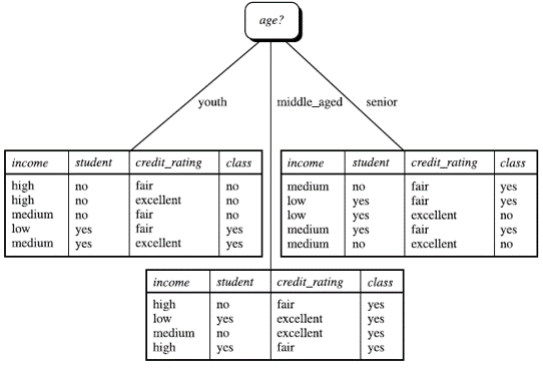
$\frac{5}{14} I(2,3)$ means "age =youth" has 5 out of 14 samples, with 2 yes's and 3 no's.

$$Info_{age}(D) = \frac{5}{14} \times \left(-\frac{2}{5}\log_2\frac{2}{5} - \frac{3}{5}\log_2\frac{3}{5}\right) + \frac{4}{14} \times \left(-\frac{4}{4}\log_2\frac{4}{4}\right) + \frac{5}{14} \times \left(-\frac{3}{5}\log_2\frac{3}{5} - \frac{2}{5}\log_2\frac{2}{5}\right) = 0.694 \text{ bits.}$$

$$Gain(age) = Info(D) - Info_{age}(D) = 0.246$$

Example

Similarly, we can find the gain for
Income: 0.029, **Student**: 0.151, **Credit Rating**: 0.048



income	*student*	*credit_rating*	*class*
high	no	fair	no
high	no	excellent	no
medium	no	fair	no
low	yes	fair	yes
medium	yes	excellent	yes

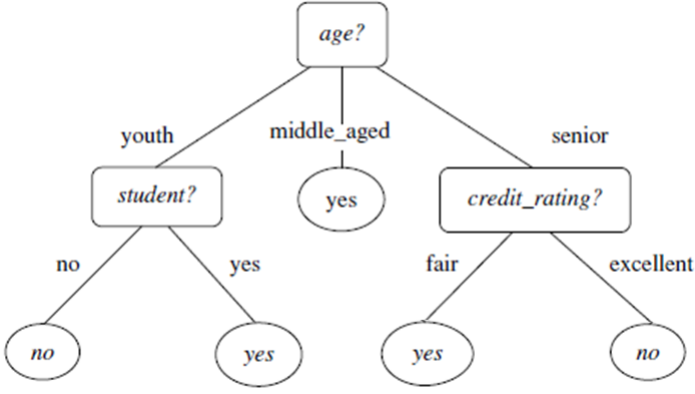
income	*student*	*credit_rating*	*class*
medium	no	fair	yes
low	yes	fair	yes
low	yes	excellent	no
medium	yes	fair	yes
medium	no	excellent	no

income	*student*	*credit_rating*	*class*
high	no	fair	yes
low	yes	excellent	yes
medium	no	excellent	yes
high	yes	fair	yes

Notice that the samples falling into the partition for age = middle_aged all belong to the same class. Because they all belong to class “yes”, a leaf should therefore be created at the end of this branch and labeled “yes”.

Example

For the other two partitions, you repeat the process. For each partition, you find the entropy and the information gain based on each feature in the partition and then select the feature with the highest information gain.



Ref for example: Data mining: concepts and techniques by Han, Jiawei, Jian Pei, and Micheline Kamber