Lecture 9

H.A. 2

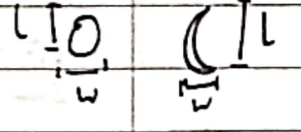
Gini index

(machine learning decision tree)

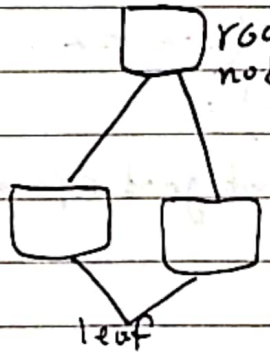
??? Didn't complete?

Covid weight	blood preasure	age	Swine covid
			yes
			no

apple banana



← starting with ~~a~~ w is the best option (IDK why)



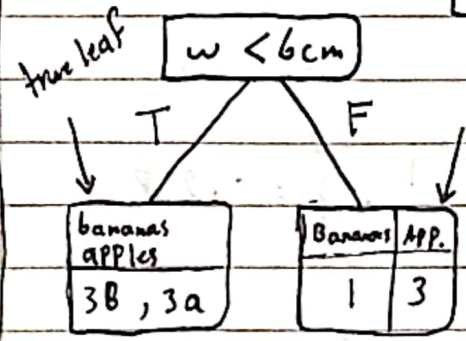
← start: w has the best variable to start with

Decide how the tree looks like

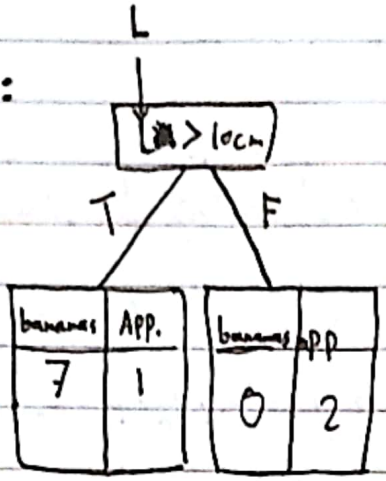
Choice 1

test on 10 known fruits.

Lowest Gini index is the one we choose



Choice 2:



test on 10 known fruits

The lines inside the leaf are made by me Niklas did not have them.

Gini index: min 0 ⟷ max 0.5

0 → good 0.5 → bad

reflects on how well did we seperate the fruit

↓

'Gini Impurity' order doesn't matter

Choice 1 true leaf: $1 - \left(\frac{\#\text{bananas}}{\#\text{bananas and apples}}\right)^2 - \left(\frac{\#\text{apples}}{\#\text{bananas \& apples}}\right)^2$ # → number

$= 1 - \left(\frac{3}{3+3}\right)^2 - \left(\frac{3}{3+3}\right)^2 =$

$= 1 - \frac{1}{4} - \frac{1}{4} = \underline{0.5}$ → not good seperation

Solve this as a formula

Choice 1 false leaf: $1 - \left(\frac{1}{1+3}\right)^2 - \left(\frac{3}{1+3}\right)^2 = \underline{0.375}$ → better seperation

choice 2 true leaf: $1 - \left(\frac{7}{7+1}\right)^2 - \left(\frac{1}{7+1}\right)^2 = 0.218$

choice 2 false leaf: $1 - \left(\frac{0}{0+2}\right)^2 - \left(\frac{2}{0+2}\right)^2 = 0$ ← best seperation

Ибрахим

number

this is the gini index for choice 1

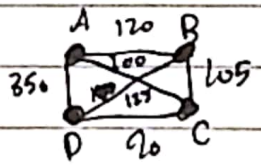
Gini Index: $\frac{\#\text{ of fruits in the true leaf}}{\text{total \# of fruits}} \times 0.5 + \frac{\#\text{ false leaf}}{\text{total \# of fruits}} \times 0.375$ (importance)

$= \frac{6}{10} \cdot 0.5 + \frac{4}{10} \cdot 0.375 = 0.45$ ← Index for choice 1

close 0.5, so not GOOD!

choice 2 → Gini Index: weighted average $\frac{8}{10} \cdot 0.218 + \frac{2}{10} \cdot 0 = 0.175$ ↓ better choice

Dijkstra's Algo.

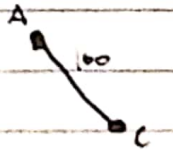


if one edge makes a loop then that's incorrect

Look at slides on canvas

this is important as well might come on H.A.
we also have to follow the instructions exactly.
Not just find the shortest way.

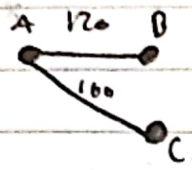
Shortest dist. from A to C = 100



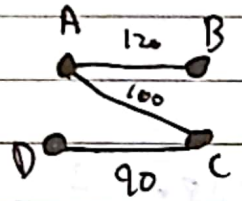
A, AC, C

From A to B = 120

A, AB, B



From A to D = 190 ← shortest path back to the starting vertex



A, AC, C, CD, D

check the distance all the way to the starting vertex ← Dijkstra's Algo.