UCIrvine | The Paul Merage School of Business

2022.11.24

BANA 273 Session 6

Classif n Trees

Prof. Vibs

The Paul Merage School of Business

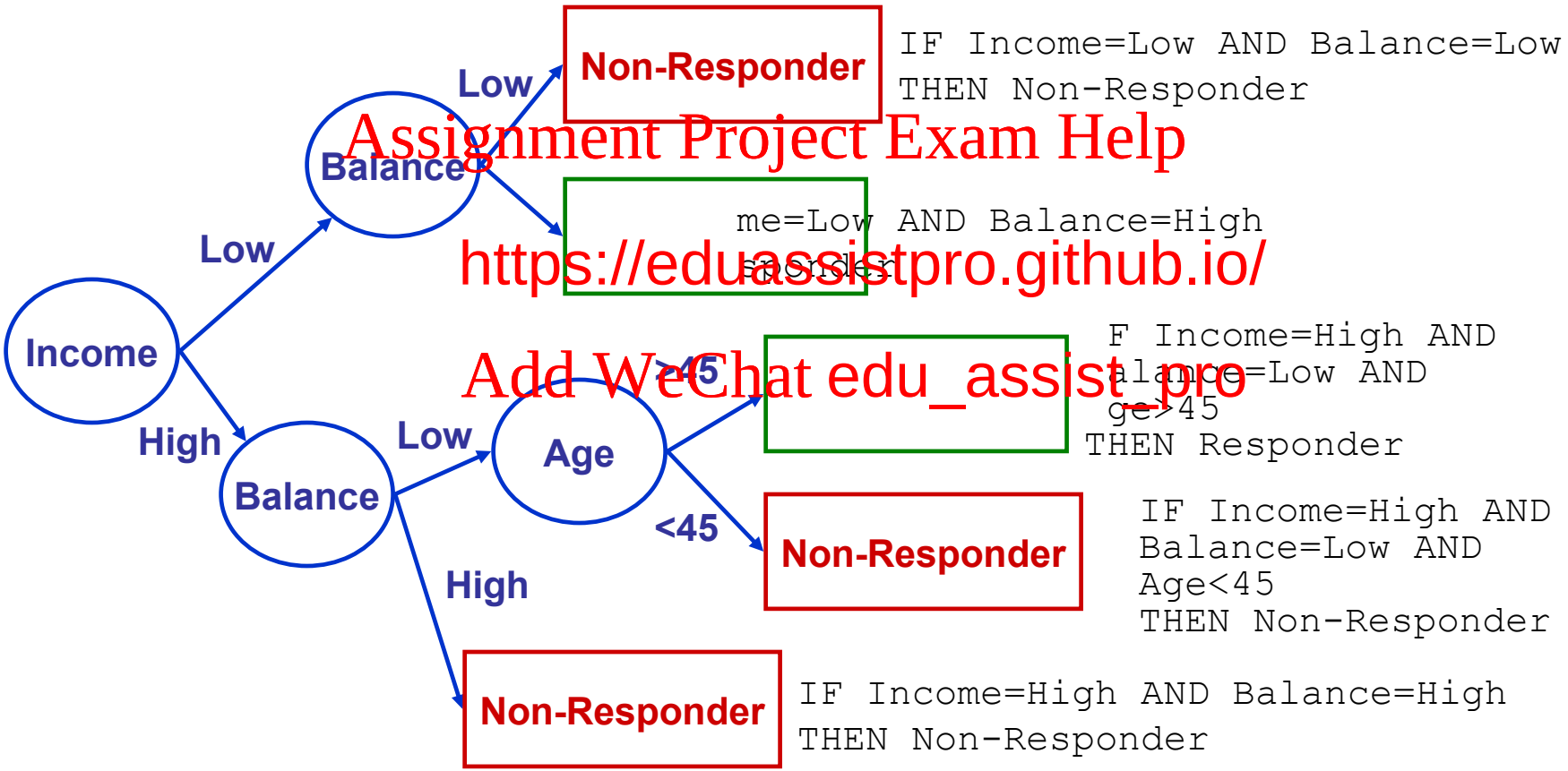
University of California, Irvine

Agenda

- Using Decision Tree for Classification
- Building Decision Trees
- Review Ass

Reading Rules off the Decision Tree

For each leaf in the tree, read the rule from the root to that leaf. You will arrive at a set of rules.



IF Income=Low AND Balance=Low
THEN Non-Responder

me=Low AND Balance=High
sponder

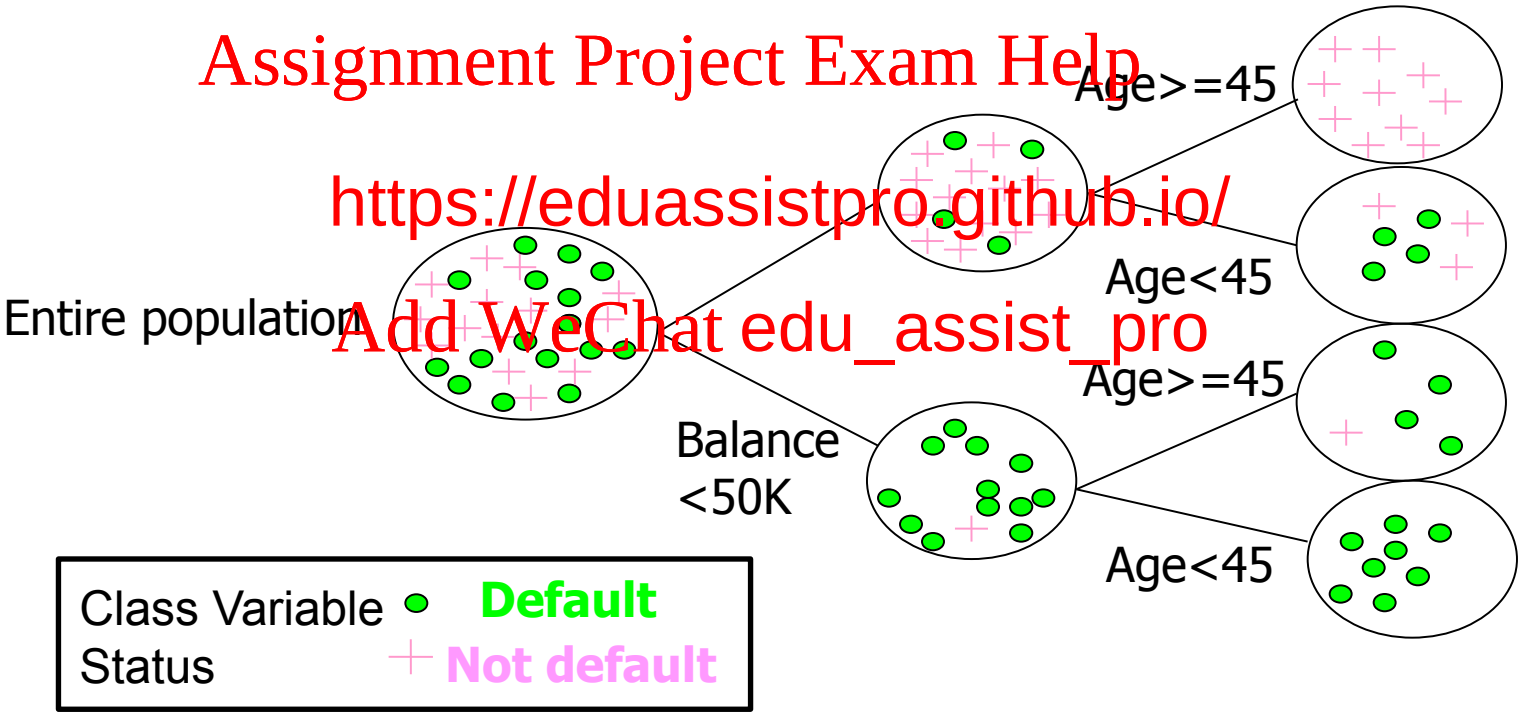
F Income=High AND
alance=Low AND
ge>45
THEN Responder

IF Income=High AND
Balance=Low AND
Age<45
THEN Non-Responder

IF Income=High AND Balance=High
THEN Non-Responder

Goal of Decision Tree Construction

- Partition the training instances into purer sub groups
  - pure: the instances in a sub-group mostly belong to the same class



- How to build a tree: How to split instances into purer sub-groups

Purity Measures

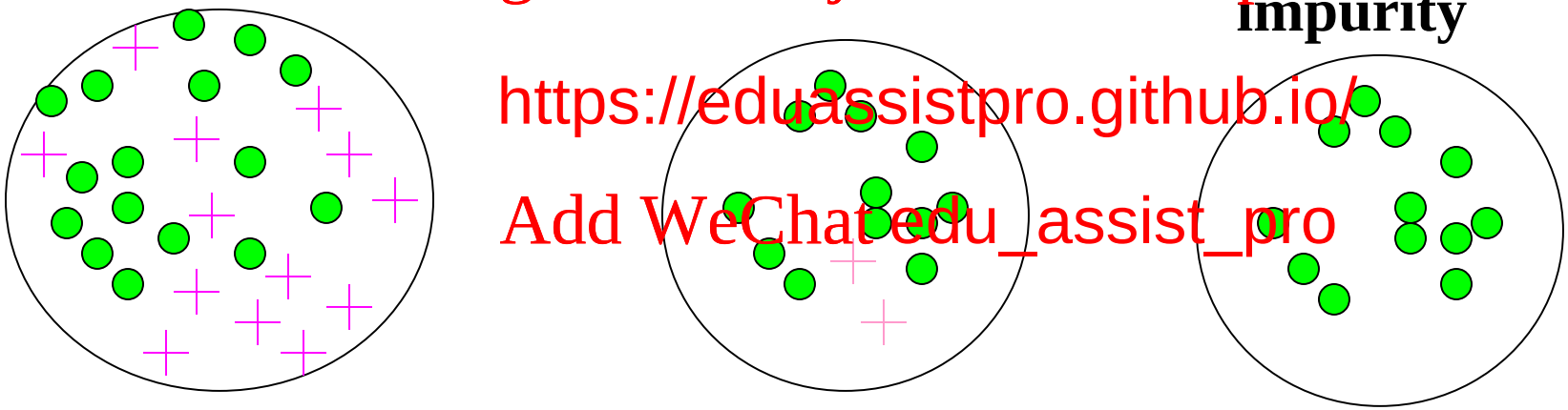
- Purity measures: Many available
  - Gini (population diversity)
  - Entropy (information gain)
  - Information
  - Chi-square T
- Most common one (from infor ity) is: *Information Gain*

Why do we want to identify pure sub groups?

- To classify a new instance, we can determine the leaf that the instance belongs to based on its attributes.
- If the leaf is v ulted) we can determine wit e new instance belongs to this class (i.e., the “ ss.)
- If the leaf is not very pure (e.g mixture of the two classes, Default and Not Default), our prediction for the new instance is more like a random guess.

Impurity

Very impure group **Less impure** **Minimum impurity**



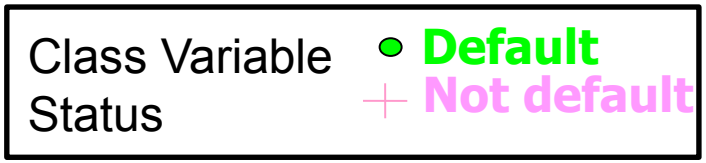
The figures above show distribution of the class variable

Class Variable Status — **Default** / **Not default**

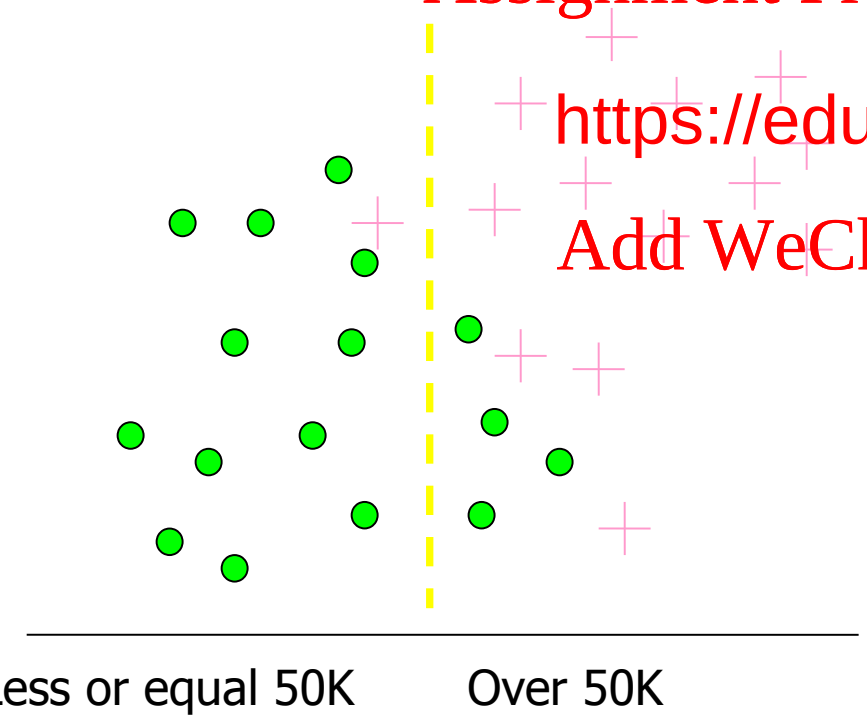
Example Split

Consider the two following splits.

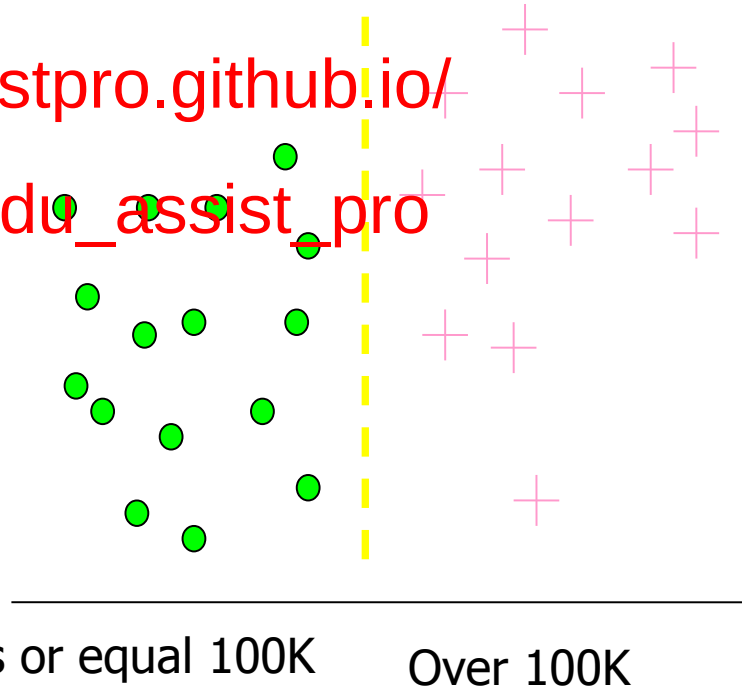
Which one is more informative?



Split over whether Balance exceeds 50K



Split over whether Income exceeds 100K



Decision Tree Construction

- A tree is constructed by recursively partitioning the examples.

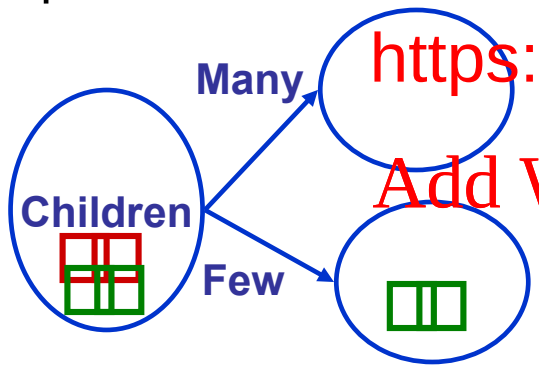
- With each par split into increasingly purer sub grou

- The key in building a tree: How to split

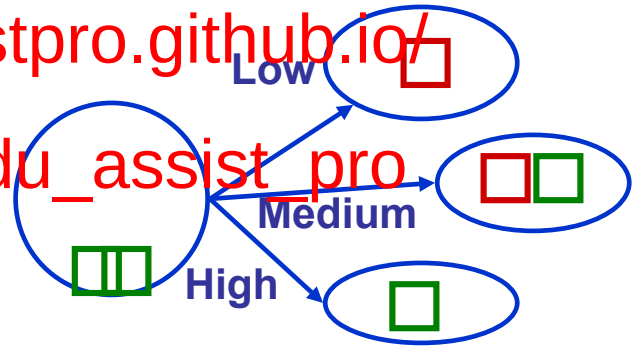
Choosing a Split

ApplicantID	City	Children	Income	Status
1	Philly	Many	Medium	DEFAULTS
2	Philly	Many	Low	DEFAULTS
3	Philly	Few	Medium	PAYS
4	Philly	Few	High	PAYS

Try split on Childre



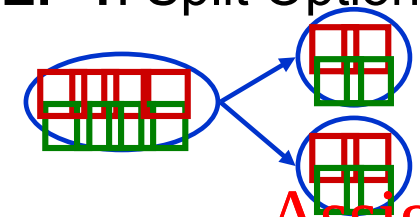
Income attribute:



Notice how the split on the Children attribute gives purer partitions. It is therefore chosen as the first split (and in this case the only split – because the two sub-groups are 100% pure).

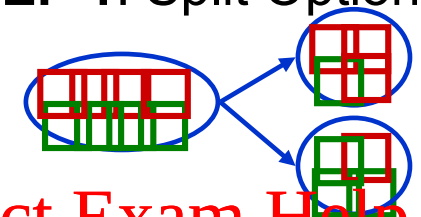
Recursive Steps in Building a Tree Example

STEP 1: Split Option A



Not good as sub-nodes are still very heterogeno

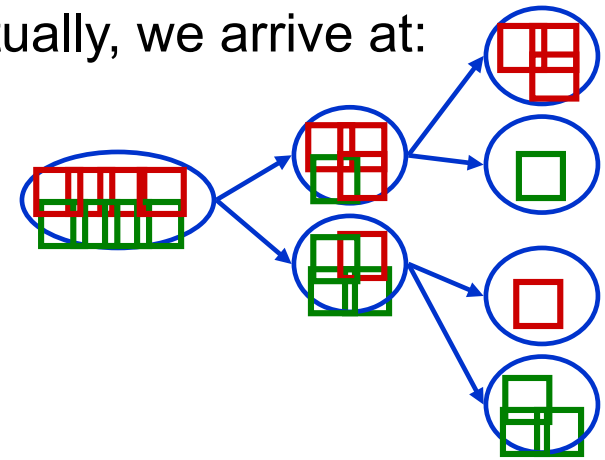
STEP 1: Split Option B



Better, as purity of sub-nodes

STEP 2: Choose S r split.

STEP 3: Try out splits on each of the f Split Option B. Eventually, we arrive at:



Notice how examples in a parent node are split between sub-nodes - i.e. notice how the training examples are partitioned into smaller and smaller subsets. Also, notice that sub-nodes are purer than parent nodes.

Example 1: Riding Mower

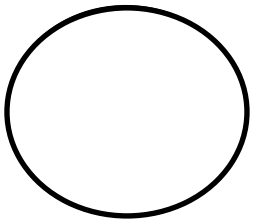
Scatterplot of Lot Size versus Income

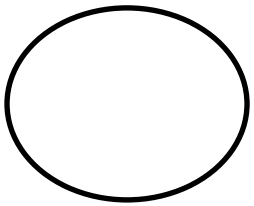
Splitting the Observations by Lot Size Value of 19

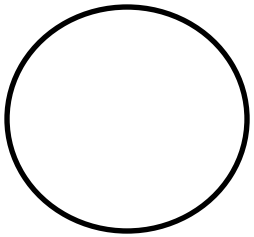
Tree Diagram: First Split

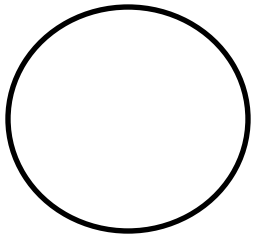
Second Split: Lot Size Value of 19K and then Income Value of 84.75K

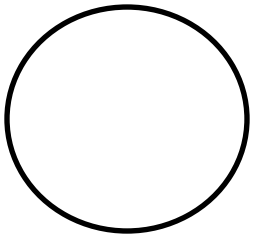
Tree Diagram: First Two Splits









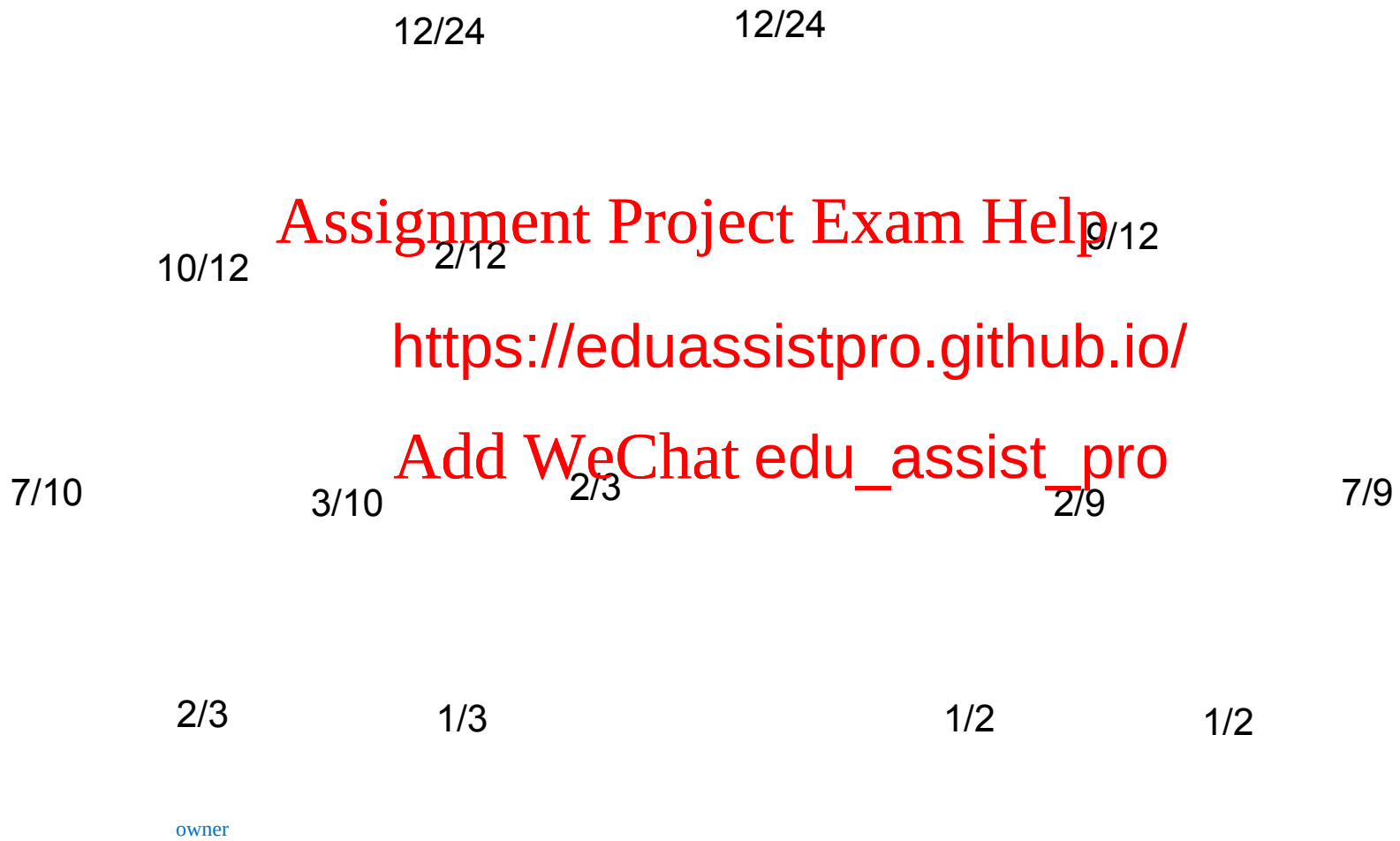


Final Partitioning

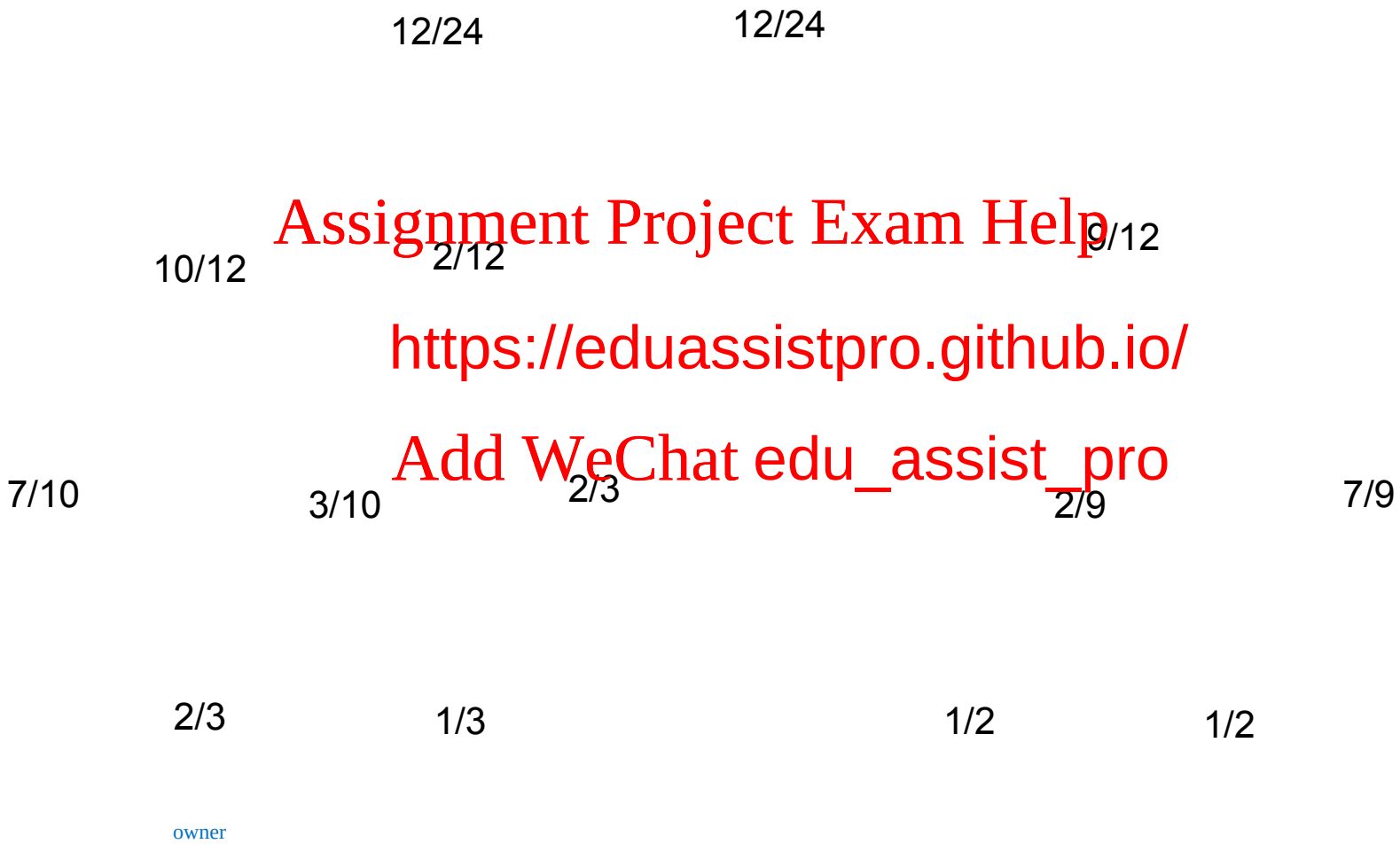
Full Tree

owner

Calculate the probability of each branch

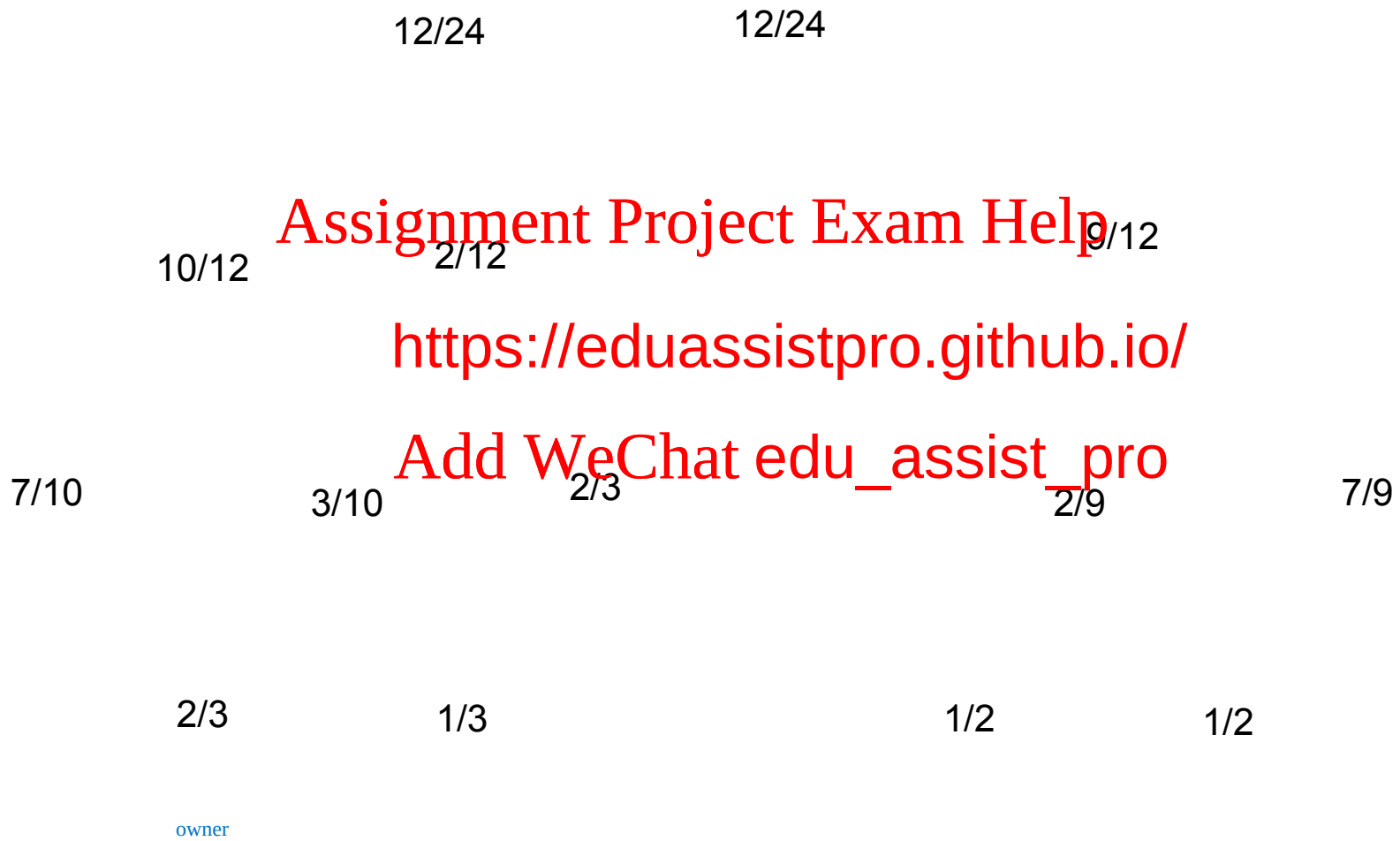


Given lot size = 20, what is the probability of owner?



P(Owner | Lot size = 20) = P(Owner & Lot Size=20)/ (P(Owner & Lot Size=20)+P(Non-Owner & Lot Size=20))

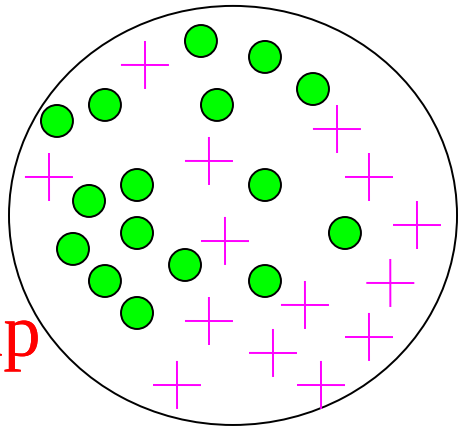
Given Income = 60, what is the probability of owner?



Calculating Impurity

- Impurity = Entropy = $\sum_i -p_i \log_2 p_i$

  p_i is proportion of class i



- For example n is composed of 16 cases of class "Def 4 cases of class "Not default"

Entropy(entire population of examples)=

$$-\left(\frac{14}{30}\cdot\log_2\frac{14}{30}\right)-\left(\frac{16}{30}\cdot\log_2\frac{16}{30}\right)=0.997$$

Calculating the Information Gain of a Split

1. For each sub-group produced by the split, calculate the impurity/entropy of that subset.

2. Calculate the weighted entropy of the split by weighting each sub-group's entr ning examples (out of the training exa t are in that subset.

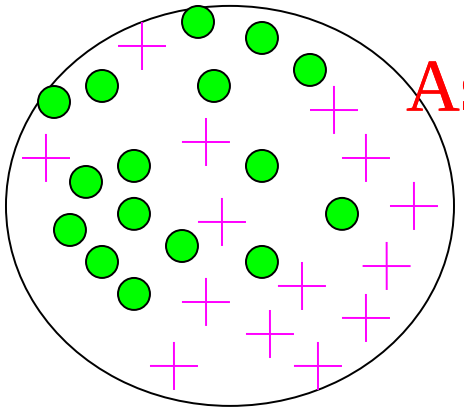
3. Calculate the entropy of the pare ubtract the weighted entropy of the child nodes to obtain the information gain for the split.

Calculating Information Gain

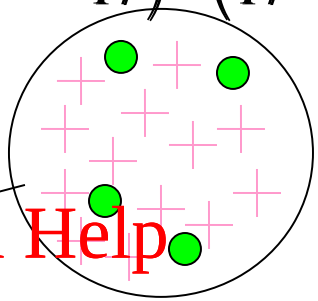
Information Gain = Entropy (parent) – Entropy (children)

$$impurity = -\left(\frac{13}{17}\cdot\log_2\frac{13}{17}\right)-\left(\frac{4}{17}\cdot\log_2\frac{4}{17}\right)=0.787$$

Entire population (30 instances)



Balance>=50K

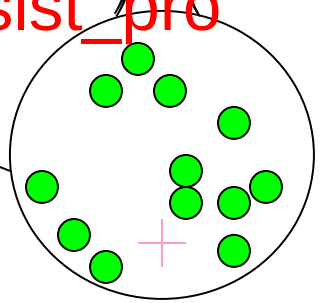


17 instances

$$\left(\frac{1}{13}\right)-\left(\frac{12}{13}\cdot\log_2\frac{12}{13}\right)=0.391$$

Balance<50K

$$impurity = -\left(\frac{14}{30}\cdot\log_2\frac{14}{30}\right)-\left(\frac{16}{30}\cdot\log_2\frac{16}{30}\right)=0.997$$



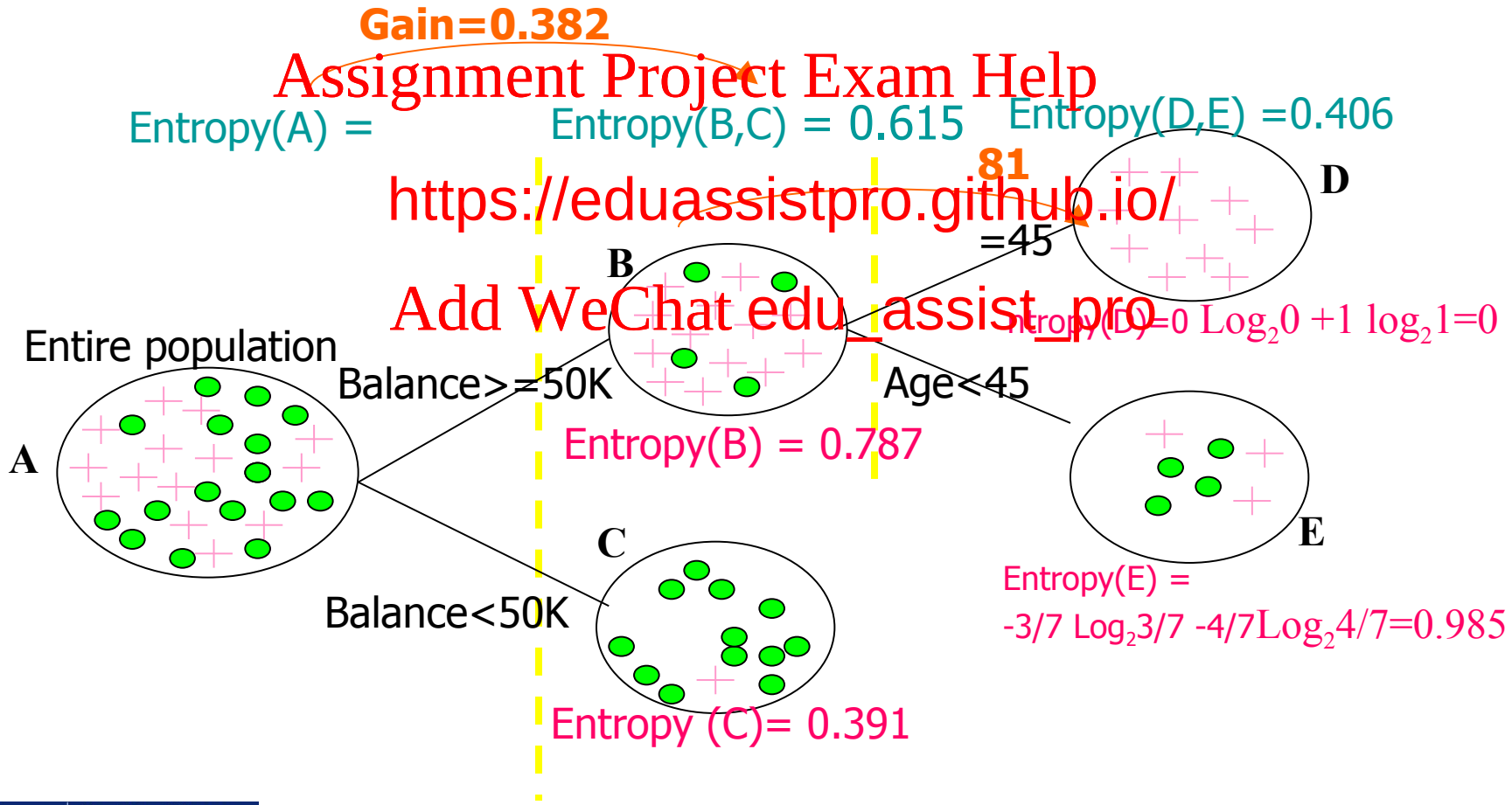
13 instances

(Weighted) Average Entropy of Children = $\left(\frac{17}{30}\cdot 0.787\right)+\left(\frac{13}{30}\cdot 0.391\right)=0.615$

Information Gain= 0.997 - 0.615 = 0.382

Information Gain

Information Gain = Entropy (parent) – Entropy (children)



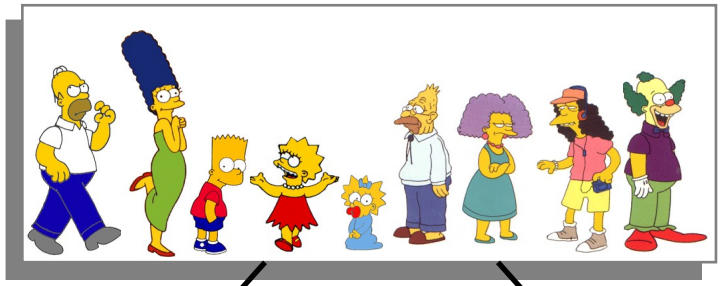
Which attribute to split over?

- At each node examine splits over each of the attributes
- Select the attribute for which the maximum information gain is obtaine
  - For a continuo ider different ways of splitting (>50 or <=50; >60 or
  - For a categorical attribute with l values, sometimes also need to consider how to group these values (branch 1 corresponds to {A,B,E} and branch 2 corresponds to {C,D,F,G})

Example 2

Person	Hair Length	Weight	Age	Class
 Homer	0”	250	36	M
 Marge	10”	150	34	F
 Bart	2”	90	10	M
 Lisa	6”	78	8	F
			1	F
 Abe	1”		70	M
 Selma	8”	160	41	F
 Otto	10”	180	38	M
 Krusty	6”	200	45	M

 Comic	8”	290	38	?

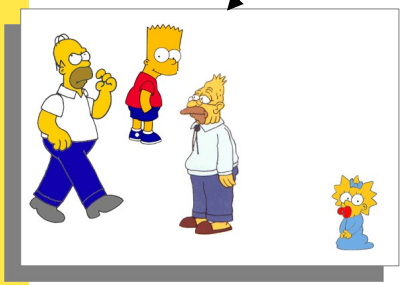


$Entropy(4\mathbf{F},5\mathbf{M}) = -(4/9)\log_2(4/9) - (5/9)\log_2(5/9)$
$= \mathbf{0.9911}$

yes

no

Hair Length <= 5?



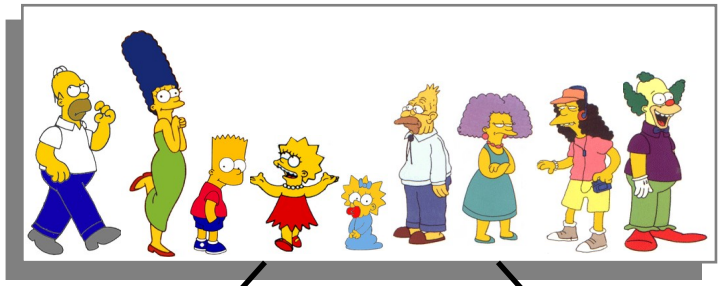
Let us try splitting on *Hair length*

$Entropy(1\mathbf{F},3\mathbf{M}) = -(1/4)\log_2(1/4) - (3/4)\log_2(3/4)$
$= \mathbf{0.8113}$

$Entropy(3$
$= \mathbf{0.9710}$
$g_2(3/5) - (2/5)\log_2(2/5)$

***Gain*= Entropy of parent – Weighted average of entropies of the children**

Gain(Hair Length <= 5) = **0.9911** – (4/9 * **0.8113** + 5/9 * **0.9710**) = **0.0911**

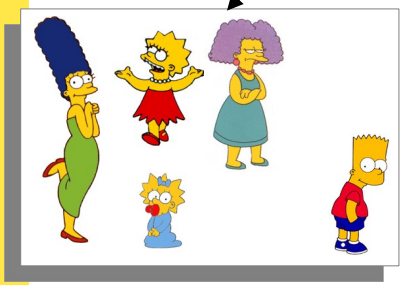


$Entropy(4\mathbf{F},5\mathbf{M}) = -(4/9)\log_2(4/9) - (5/9)\log_2(5/9)$
$= \mathbf{0.9911}$

yes

no

Weight <= 160?

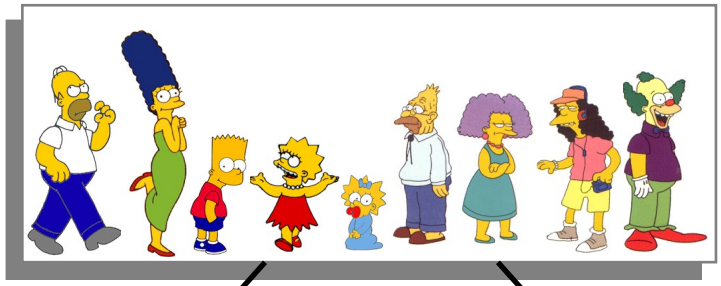


Let us try splitting on *Weight*

$Entropy(4\mathbf{F},1\mathbf{M}) = -(4/5)\log_2(4/5) - (1/5)\log_2(1/5)$
$= \mathbf{0.7219}$

$Entropy(0$ $g_2(0/4) - (4/4)\log_2(4/4)$
$= \mathbf{0}$

Gain(Weight <= 160) = **0.9911** – (5/9 * **0.7219** + 4/9 * **0**) = **0.5900**

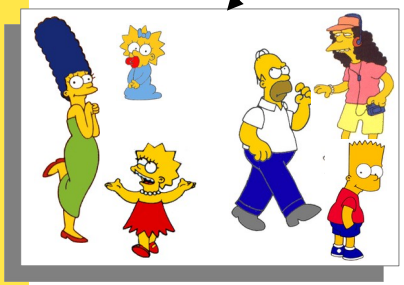


$Entropy(4\mathbf{F},5\mathbf{M}) = -(4/9)\log_2(4/9) - (5/9)\log_2(5/9)$
$= \mathbf{0.9911}$

yes

no

age <= 40?



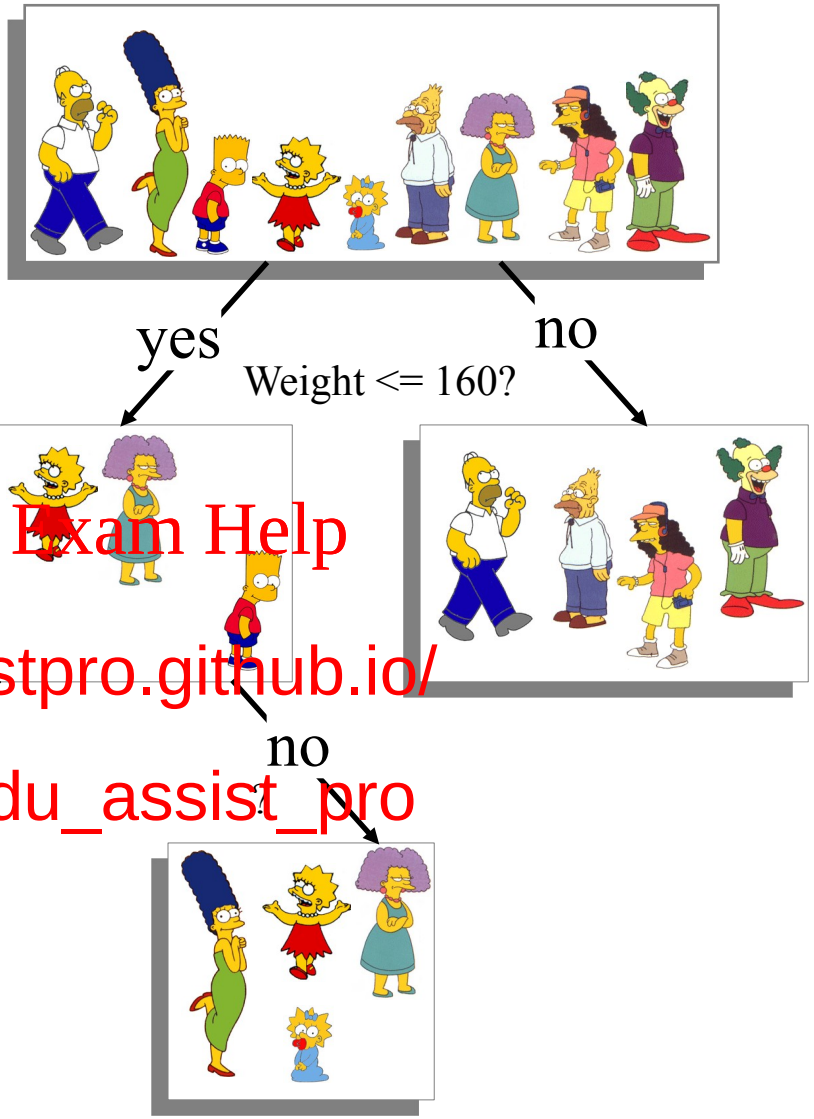
Let us try splitting on *Age*

$Entropy(3\mathbf{F},3\mathbf{M}) = -(3/6)\log_2(3/6) - (3/6)\log_2(3/6)$
$= \mathbf{1}$

$Entropy(1$ $g_2(1/3) - (2/3)\log_2(2/3)$
$= \mathbf{0.9183}$

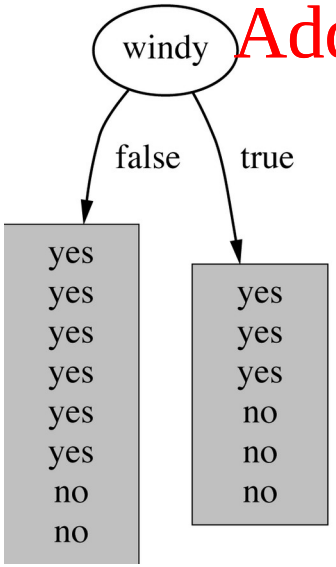
Gain(Age <= 40) = **0.9911** – (6/9 * **1** + 3/9 * **0.9183**) = **0.0183**

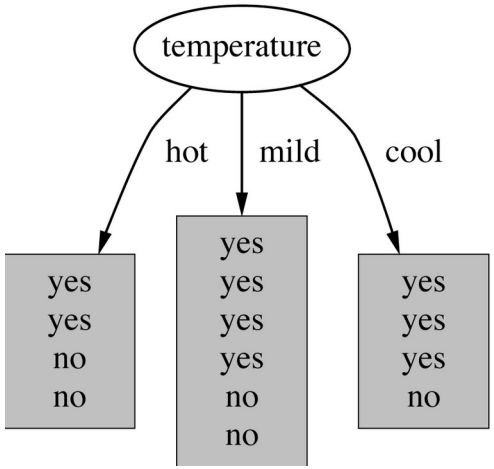
Of the 3 features we had, *Weight* was the best. But while people who weigh over 160 are perfectly classified (as males), the under 160 people are not perfectly classified… So we continue splitting



This time we find that we can split on *Hair length,* and then we are done!

Example 3: Which attribute to split on?





Exercise – Decision Tree

Customer ID	Student	Credit Rating	Class: Buy PDA
1	No	Fair	No
2	No	Excellent	No
3	No	Fair	Yes
4	No	Fai	
5	Yes	Fai	
6	Yes	Excellent	No
7	Yes	Excellent	Yes
8	No	Excellent	No

Which attribute to split on first?

$\log_2(2/3) = -0.585$, $\log_2(1/3) = -1.585$, $\log_2(1/2) = -1$, $\log_2(3/5) = -0.737$,
$\text{Log}_2(2/5) = -1.322$, $\log_2(1/4) = -2$, $\log_2(3/4) = -0.415$

Building a Tree - Stopping Criteria

- You can stop building the tree when:
  - The impurity of all nodes is zero: Problem is that this tends to lead to bushy, highly branching trees, often with one ex
  - No split ach in purity (information gain not high
  - Node size is too small: Th are less than a certain number of examples, or proportion of the training set, at each node.

Over-fitting

Overfitting & Underfitting

- **Overfitting:** the model performs poorly on new examples (e.g. testing examples) as it is too highly trained to the specific training examples (pick up patterns and noises).
- **Underfitting:** the model performs poorly on new examples as it is too simplistic to distinguish between them (i.e. has not picked up the important patterns from the training examples)

Notice how the error rate on the testing data increases for overly large trees.

Pruning

A decision trees is typically more accurate on its *training* data than on its *test* data. Removing branches from a tree can often improve its accuracy on a test set.

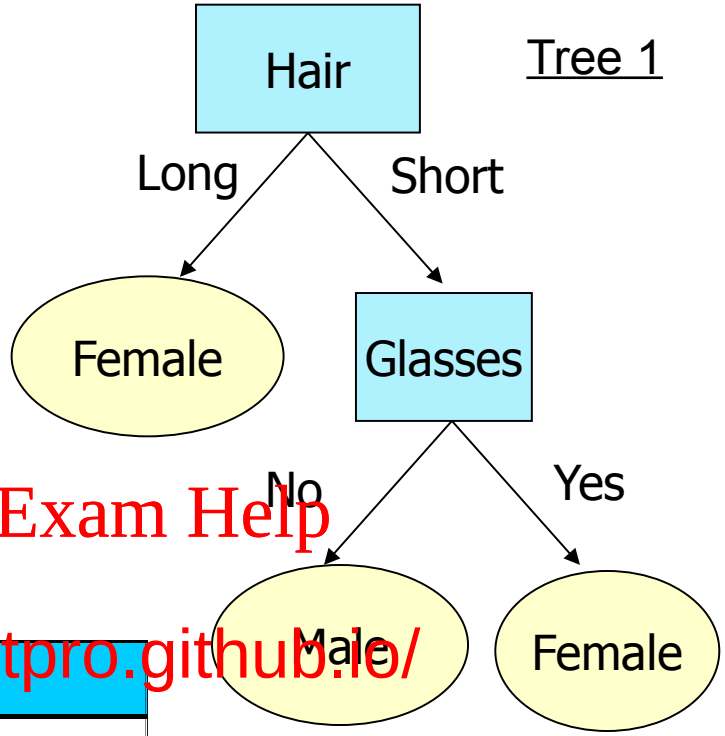
Classification and Regression Tree (CART) : Use validation data to delete "weak" sub-trees

Assess whether rity by a statistically significant amou

Training

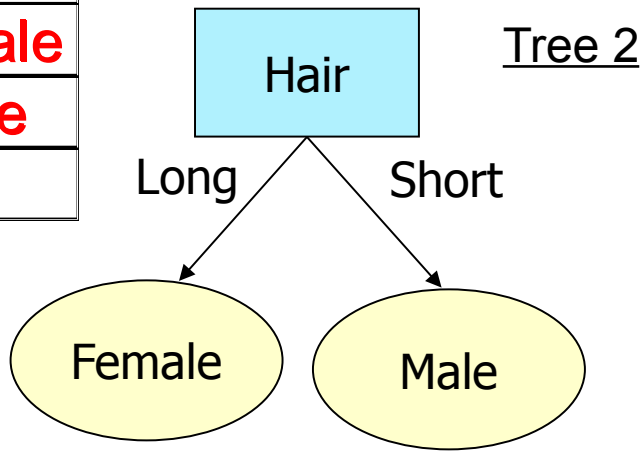
Name	Hair	Glasses	Class
Mary	Long	No	Female
Mike	Short	No	Male
Bill	Short	No	Male
Jane	Long	No	Female
Ann	Short	Yes	Female

100% accurate on training data



Testing

Hair	Glasses	Tr		
Short	Yes	Female	Male	
Short	No	Male	Male	
Long	No	Female	Female	Female
Short	Yes	Female	Male	Male
	Error:	75%	25%	



There are many possible splitting rules that perfectly classify the data, but will not generalize to future datasets.

Decision Tree Classification in a Nutshell

- Decision tree
  - A tree structure
  - Internal node denotes a test on an attribute
  - Branch represents an outcome of the test
  - Leaf nodes represent class labels
- Decision tree g ases
  - Tree constr
    - At start, the root
    - Partition examples recursiv elected attributes
  - Tree pruning
    - Identify and remove branches that reflect noise or outliers
    - To avoid overfitting
- Use of decision tree: Classifying an unknown sample
  - Test the attribute values of the sample against the decision tree

Strengths

- In practice: One of the most popular methods
  - Very comprehensible – the tree structure specifies the entire decision structure
    - Easy for decision makers to understand model's rational
    - Map nicely to a set of business rules
  - Relatively easy t
- Very fast to run (large data sets
- Good at handling missing values: issing" as a value – can become a good predictor
- Weakness
  - Bad at handling continuous data, good at categorical input and output.

Which attribute will you use as the root of the tree, given the following information:

gain(*Outlook*) = 0.247 bits
gain(*Temperature*) = 0.029 bits
gain(*Humidity*) = 0.152 bits
gain(*Windy*) = 0.048 bits

A: Outlook

B: Humidity

C: Windy

D: Temperature

E: None of the above

What is overfitting?

A: When the model fit is better on the top side

B: When the model fit is worse on the top side

C: When the m ect trend and has best acc

D: When the model captures e data, hurting accuracy

E: None of the above

Weka Example – Classification using Naïve Bayes

- Download file from Canvas:
  - 4bank-data-8.arff
- Switch tab t
- Select meth
- Verify class variable set t
- Use 10 fold cross validation
- Run classifier
- Examine confusion matrix

Weka Exercise

- Follow instructions on
- http://faculty.cbpa.louisville.edu/mxanalis/classes/ect584/WEKA/c
- Data files po
- We will use J48 which is entation of the C4.5 algorithm

Next Session

- Association Rules