Advanced Business Data Mining

MSIS 522 – Lesson 2

Foster
School of Business

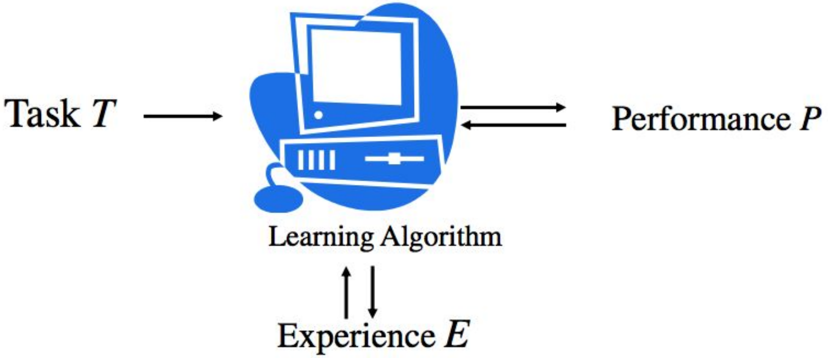
Course Overview

- Lecture 1 - Fundamentals of Machine Learning
- **Lecture 2 - Decision Tree**
- Lecture 3 - Ensemble Learning
- Lecture 4 - Clustering
- Lecture 5 - Recommendation Systems

Recap of Lesson 1

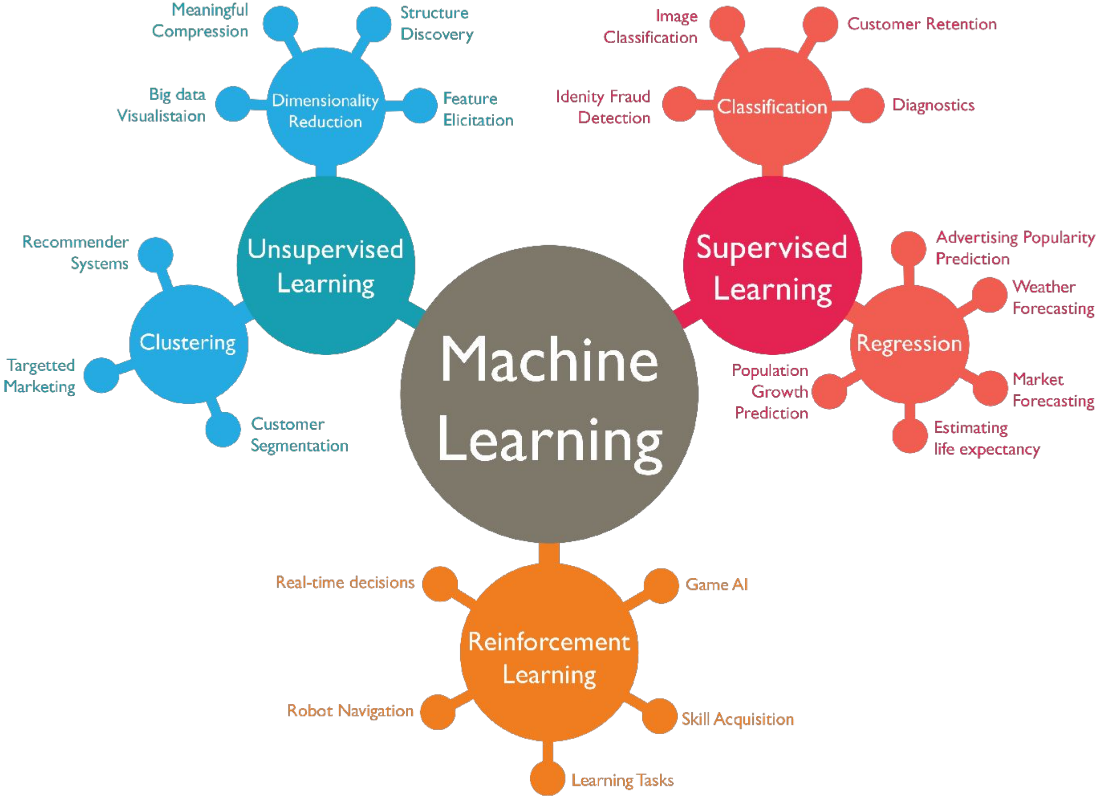
What is Machine Learning?

A computer program is said to learn from **experience *E*** with respect to some class of **tasks *T*** and **performance measure *P*** if its performance at tasks in *T*, as measured by *P*, improves with experience *E*. -- Tom Mitchell



Improving ***performance P*** with ***experience E*** at some ***task T.***

3 Types of Machine Learning Algorithms



W Foster School of Business

Supervised Learning

- **Regression**: A regression model predicts **continuous values**. For example, regression models make predictions that answer questions like the following:
  - What is the value of a house in California?
  - What is the demand of a product on Amazon next month?

- **Classification**: A classification model predicts **discrete values**. For example, classification models make predictions that answer questions like the following:
  - Is a given email message spam or not spam?
  - Is this an image of a dog, a cat, or a hamster?

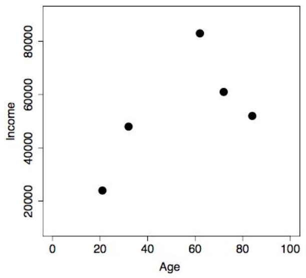
Generalization

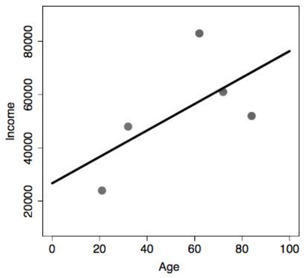
Machine Learning is all about **generalization** to future unseen data points.

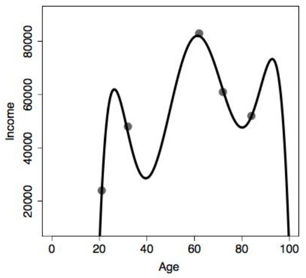
- **Underfitting** – a model is too simple and can not capture the underlying patterns within the data, thus does not perform well on new data.
- **Overfitting** – a model tries to fit the training data so closely that it does not generalize well to new data.

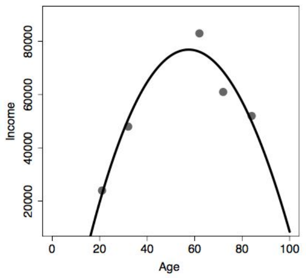
Table: The age-income dataset.

ID	AGE	INCOME
1	21	24,000
2	32	48,000
3	62	83,000
4	72	61,000
5	84	52,000





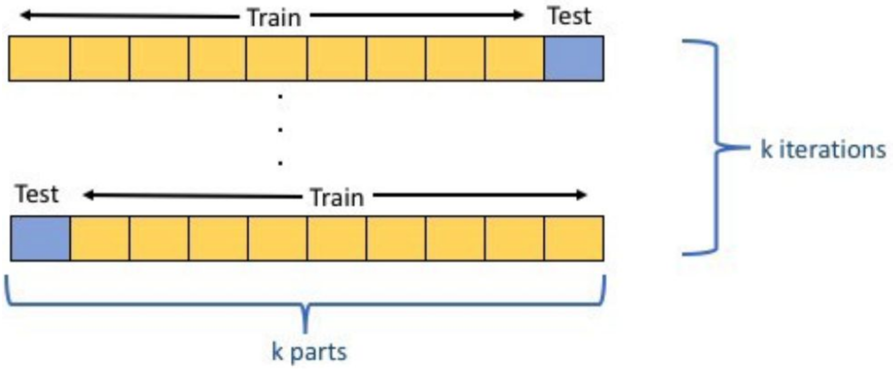




W Foster School of Business

K-fold Cross-validation

1. Divide the training data into K parts.
2. Use K-1 of the parts for training and 1 for testing.
3. Repeat the procedure K times, rotating the test set.
4. Determine the performance based on the results across all K iterations.



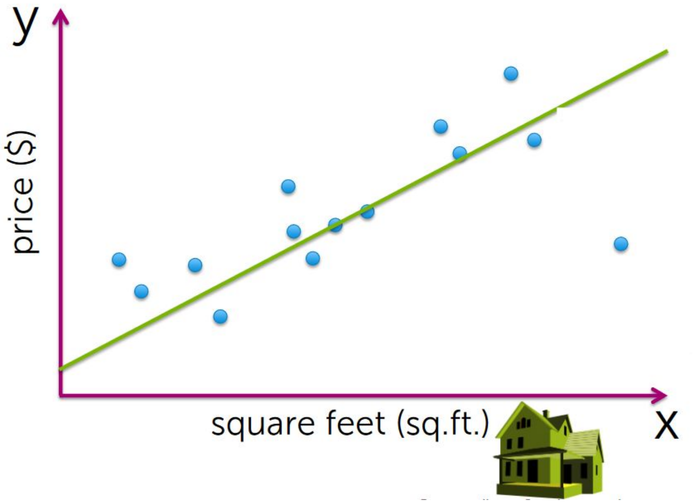
Leave-one-out Cross-validation is the extreme case of K-fold Cross-Validation where we keep only one data point in the test set.

Outline

- Linear Model and Its Limitation
- Decision Tree
- Hyper-parameter Tuning
- One-hot Encoding
- Lab

Linear Model and Its Limitation

Linear Model



Linear Regression

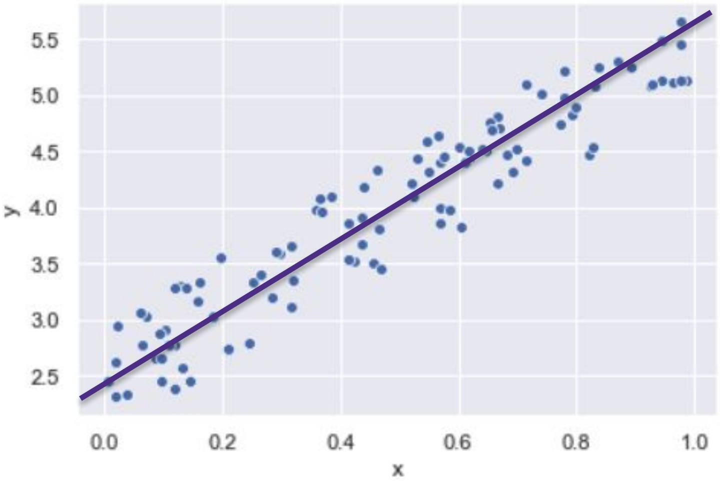
Predict continuous values, e.g. house price

Logistic Regression

Predict discrete values, e.g. email spam

Linear Regression

$$h_w(x) = w_0 + w_1 x$$

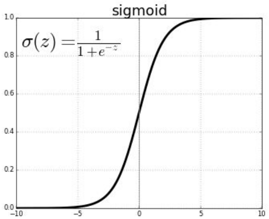


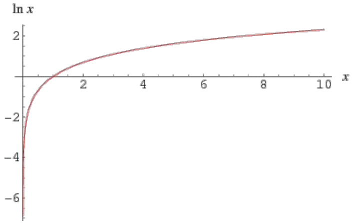
> **Mean Squared Error (MSE)**: the average squared difference between the actual and predicted values.

$$\mathcal{L}(w) = \frac{1}{N}\sum_{i=1}^{N}(y_i - h_w(x_i))^2$$

> Find parameters w = {w_0 , w_1} that minimize MSE over the training dataset.

Logistic Regression





$$h_w(x) = \sigma(w_0 + w_1 x)$$

Decision Boundary

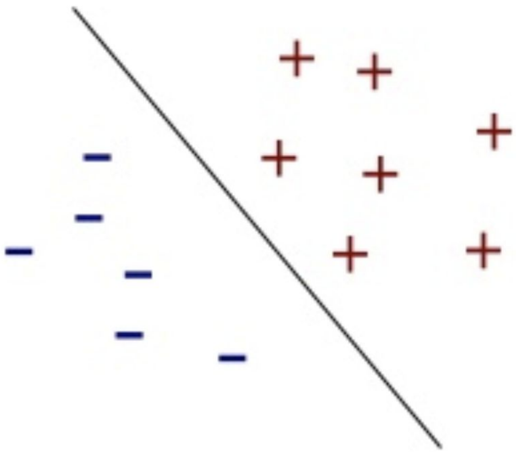
> **Cross Entropy (aka log-loss)** measures the performance of a classification model based on its probabilistic output.

$$\mathcal{L}(w) = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i \log\left(h_w(x_i)\right) + (1 - y_i)\log\left(1 - h_w(x_i)\right)\right]$$

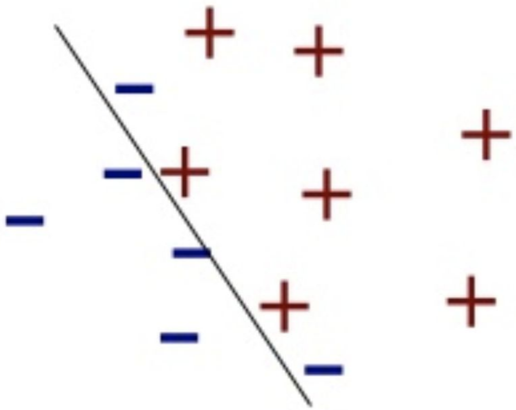
$$\begin{array}{ll} \log\left(h_w(x_i)\right) & if\ y_i = 1 \\ \log\left(1 - h_w(x_i)\right) & if\ y_i = 0 \end{array}$$

> Find parameters w = {w_0 , w_1} that minimize the cross entropy over the training dataset.

Linear Separable



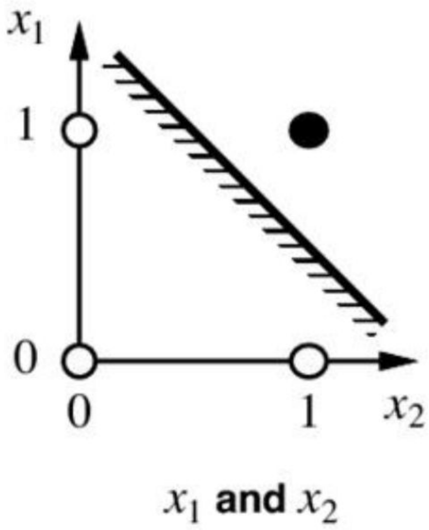
Linear Separable



Not Linear Separable

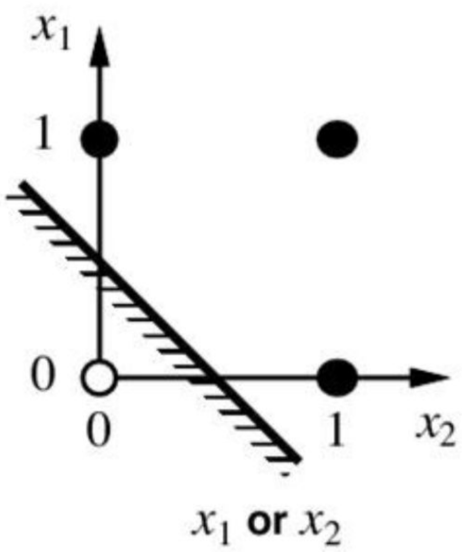
Logical AND

X1	X2	Y
1	1	1
1	0	0
0	1	0
0	0	0



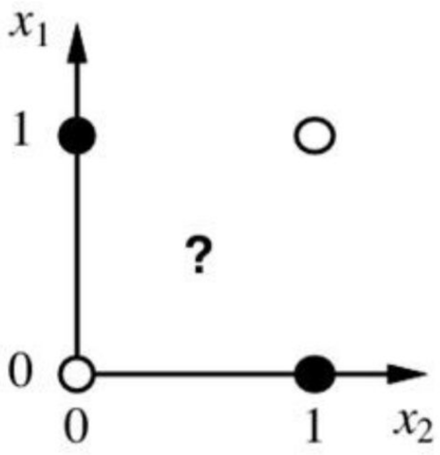
Logical OR

X1	X2	Y
1	1	1
1	0	0
0	1	0
0	0	0



Logical XOR

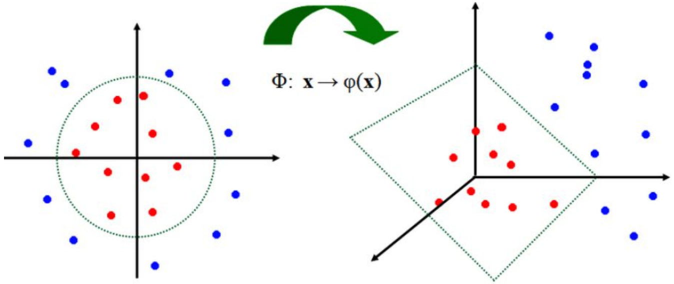
X1	X2	Y
1	1	0
1	0	1
0	1	1
0	0	0



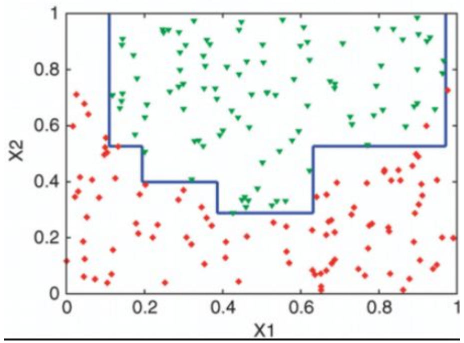
x_1 **xor** x_2

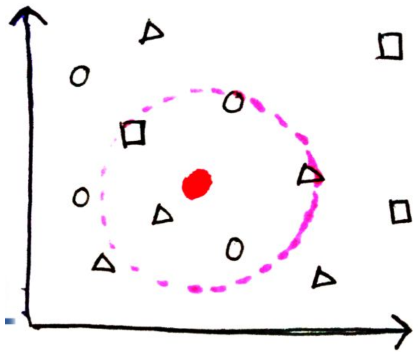
Handle Nonlinear Separable Data

- Project existing data into other dimensional space so that data becomes linear separable in that space and then apply a linear model.



- Use a more powerful model which can model non-linearity in the data by itself.

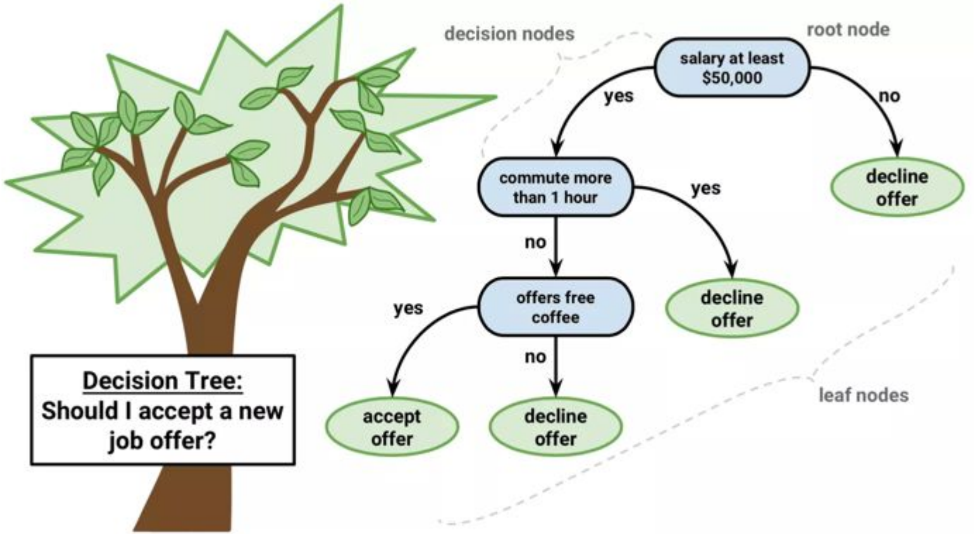




Decision Tree

Decision Tree Basics

- **Decision nodes (blue)**: each node represents a test on a particular attribute.
- **Leaf nodes (green)**: each node represents a prediction.

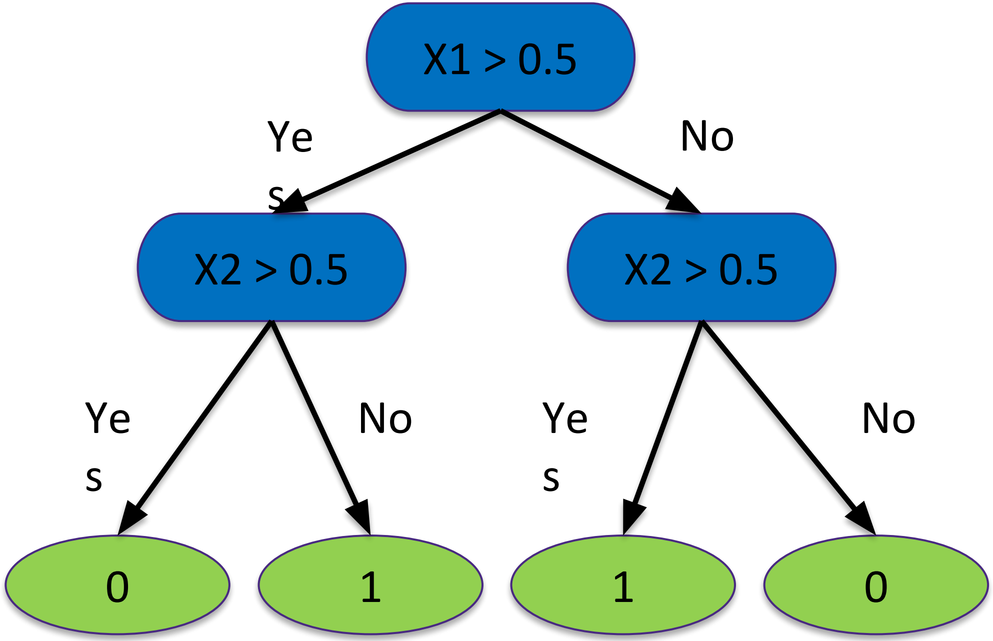


- Read down the tree to derive rules.
- # of leaf nodes equals # of rules encoded in a decision tree.

Decision Tree to the Rescue

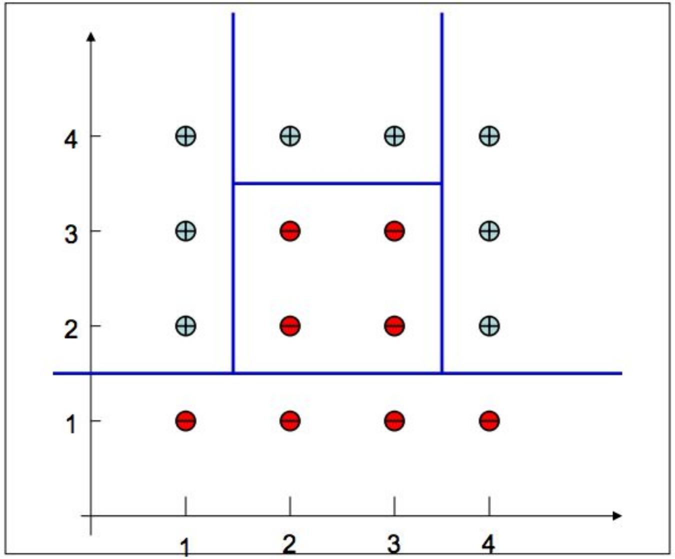
<u>Logical XOR</u>

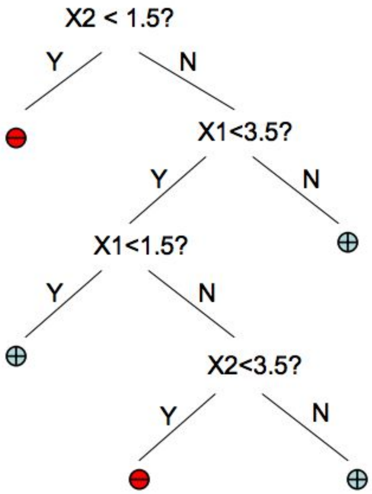
X1	X2	Y
1	1	0
1	0	1
0	1	1
0	0	0



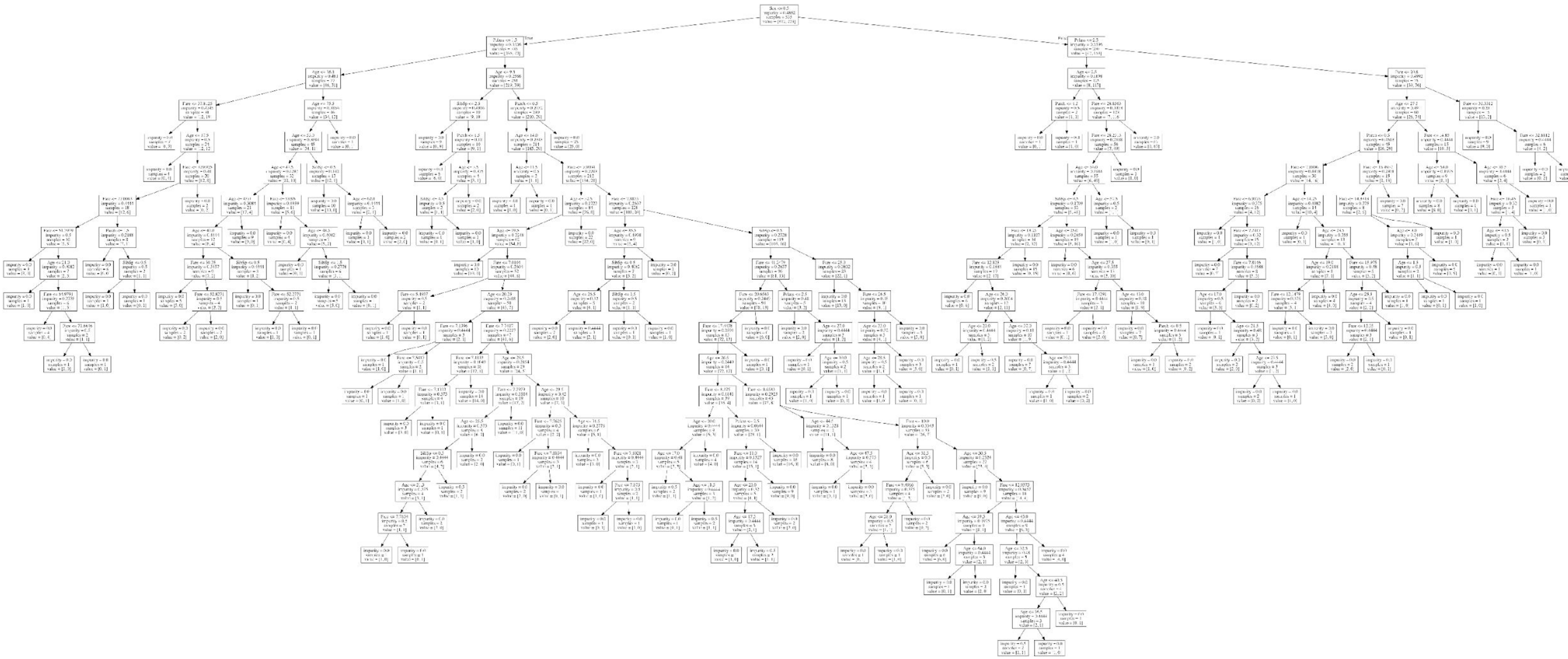
Decision Boundaries

Decision Tree divides the input space into **axis-parallel** rectangles and label each rectangle with the class with most data in it.





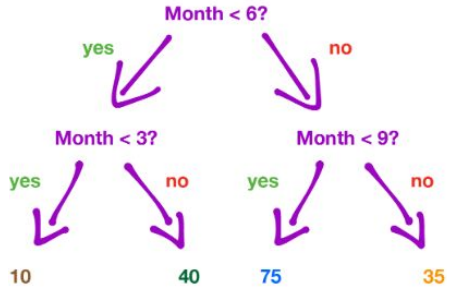
A More Realistic Decision Tree



Classification And Regression Trees (CART)

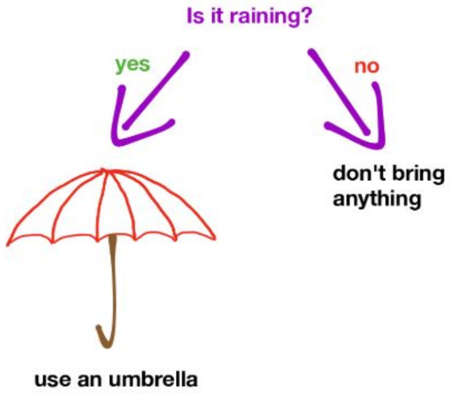
- **Classification Trees** - predict categorical variable.

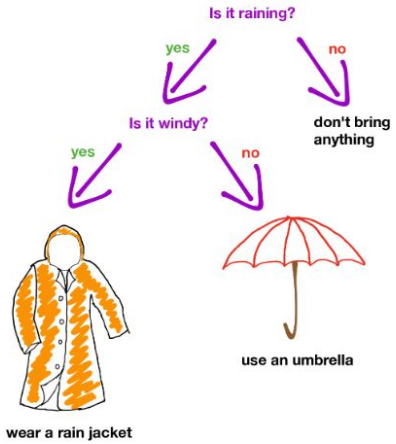
- **Regression Trees** - predict continuous variable.



How to construct a Decision Tree?

- Choose an attribute (i.e. a feature) for root.
- Split data using chosen attribute into disjoint subsets.
- Recursive partitioning for each subset.





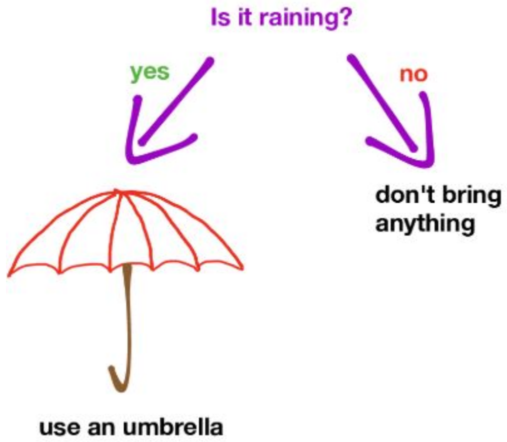
Split of a Categorical Variable

- Examine all possible ways in which the categories can be split into two groups.

- E.g. categories A, B, C can be split 3 ways.
  - {A} and {B, C}
  - {B} and {A, C}
  - {C} and {A, B}

- In theory, we have an exponential number of different splits.

- In practice, we often use one vs the rest.



How to Construct a Decision Tree

Training Set: 3 features and 2 classes

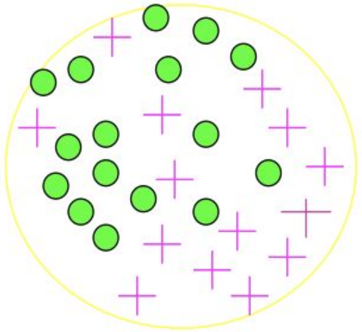
X	Y	Z	C
1	1	1	I
1	1	0	I
0	0	1	II
1	0	0	II

How do we build a Decision Tree to distinguish class I from II?

Classification Impurity Measure: Entropy

- Entropy measures the level of impurity in a group of examples.

$$H(x) = -\sum_i p_i \log(p_i)$$



16/30 are green circles; 14/30 are pink crosses

$\log_2(16/30) = -.9$; $\log_2(14/30) = -1.1$

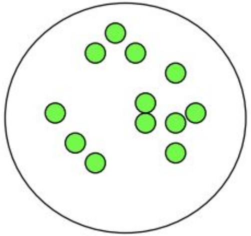
Entropy = $-(16/30)(-.9) - (14/30)(-1.1) = .99$

Entropy for 2-class Cases

- What is the entropy of a group in which all examples belong to the same class?
  - entropy = $-1 \log_2 1 = 0$

  not a good training set for learning

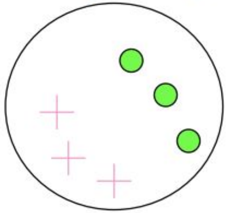
Minimum impurity



- What is the entropy of a group with 50% in either class?
  - entropy = $-0.5 \log_2 0.5 - 0.5 \log_2 0.5 = 1$

  good training set for learning

Maximum impurity



Quiz: Andrew Moore’s Entropy in a Nutshell

Low Entropy

High Entropy

Information Gain

- Determine **which attribute** in a given set of training features is most useful for discriminating between the classes.

- **Information Gain** tells us how important a given attribute is in discriminating between the classes.

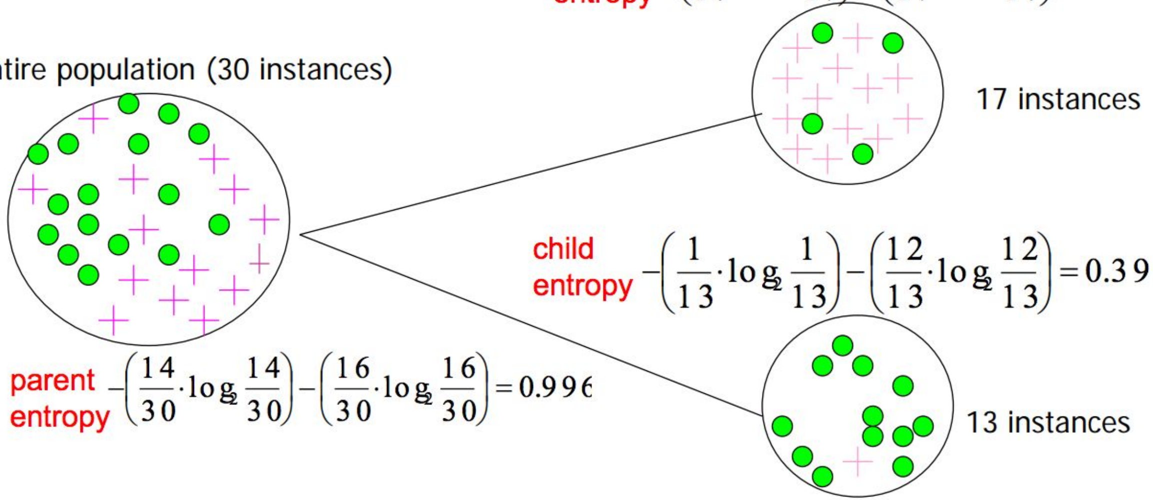
- Choose the attribute and split that maximize the information gain.

Information Gain = entropy(parent) – [average entropy(children)]

Information Gain Example

Information Gain = entropy(parent) – [average entropy(children)]

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



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

parent entropy $-\left(\frac{14}{30}\cdot\log_2\frac{14}{30}\right)-\left(\frac{16}{30}\cdot\log_2\frac{16}{30}\right)=0.996$

(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= 0.996 - 0.615 = 0.38 for this split

Quiz

Training Set: 3 features and 2 classes

X	Y	Z	C
1	1	1	I
1	1	0	I
0	0	1	II
1	0	0	II

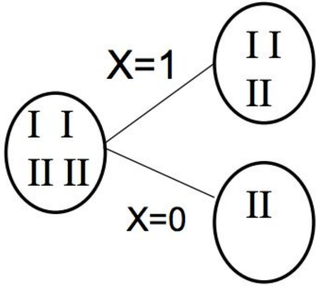
How do we build a Decision Tree to distinguish class I from II?

Quiz: Split on attribute X

X	Y	Z	C
1	1	1	I
1	1	0	I
0	0	1	II
1	0	0	II

Split on attribute X

If X is the best attribute,
this node would be further split.



E_{child1} = $-(1/3)\log_2(1/3)-(2/3)\log_2(2/3)$
$= .5284 + .39$
$= .9184$

$E_{child2} = 0$

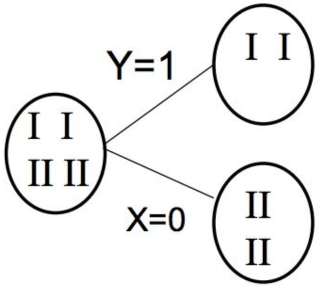
$E_{parent} = 1$

GAIN = 1 – (3/4)(.9184) – (1/4)(0) = .3112

Quiz: Split on attribute Y

X	Y	Z	C
1	1	1	I
1	1	0	I
0	0	1	II
1	0	0	II

Split on attribute Y



$E_{child1} = 0$

$E_{child2} = 0$

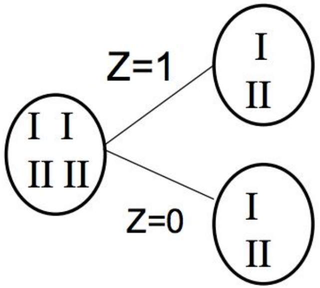
$E_{parent} = 1$

GAIN = 1 –(1/2) 0 – (1/2)0 = 1; BEST ONE

Quiz: Split on attribute Z

X	Y	Z	C
1	1	1	I
1	1	0	I
0	0	1	II
1	0	0	II

Split on attribute Z



$E_{child1} = 1$

$E_{child2} = 1$

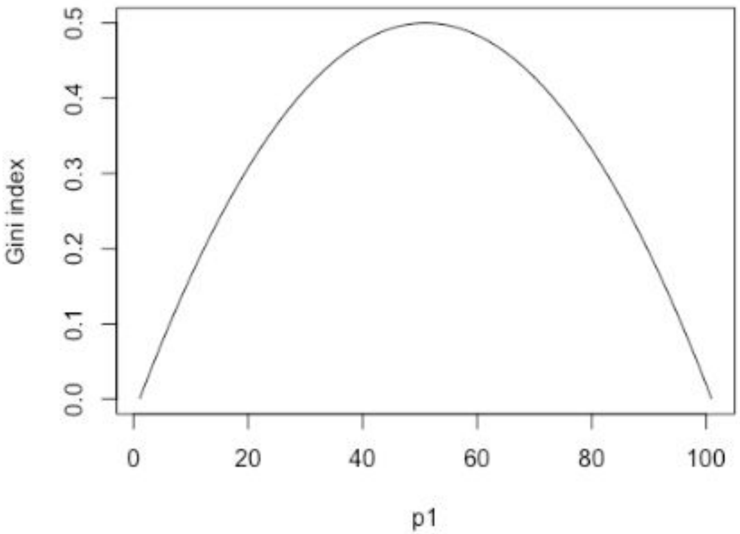
$E_{parent} = 1$

GAIN = 1 – (1/2)(1) – (1/2)(1) = 0 ie. NO GAIN; WORST

Classification Impurity Measure: Gini Impurity

$$I_G = 1 - \sum_{j=1}^{c} p_j^2$$

- **Gini impurity/index** is a measure to quantify the level of impurity in a group of examples.
  - I(A) = 0 when all cases belong to the same class.
  - Max value when all classes are equally represented.



Split of a Numerical Variable

- For each numerical attribute:
  - Sort the attribute from the smallest to the largest.
  - Linearly scan these values and choose the split position leading to the maximum impurity reduction (i.e. information gain).

Sorted Values →

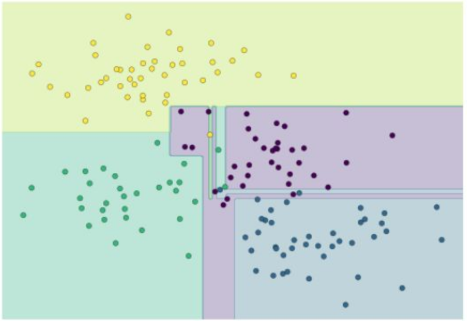
Cheat	No	No	No	Yes	Yes	Yes	No	No	No	No
Taxable Income										
	60	70	75	85	90	95	100	120	125	220

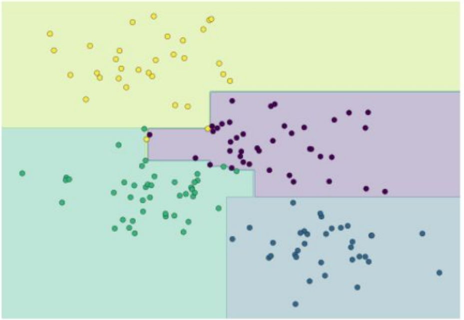
Split Positions →

	55		65		72		80		87		92		97		110		122		172		230	
	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>
Yes	0	3	0	3	0	3	0	3	1	2	2	1	3	0	3	0	3	0	3	0	3	0
No	0	7	1	6	2	5	3	4	3	4	3	4	3	4	4	3	5	2	6	1	7	0
Gini	0.420		0.400		0.375		0.343		0.417		0.400		<u>0.300</u>		0.343		0.375		0.400		0.420	

How to avoid overfitting in Decision Tree?

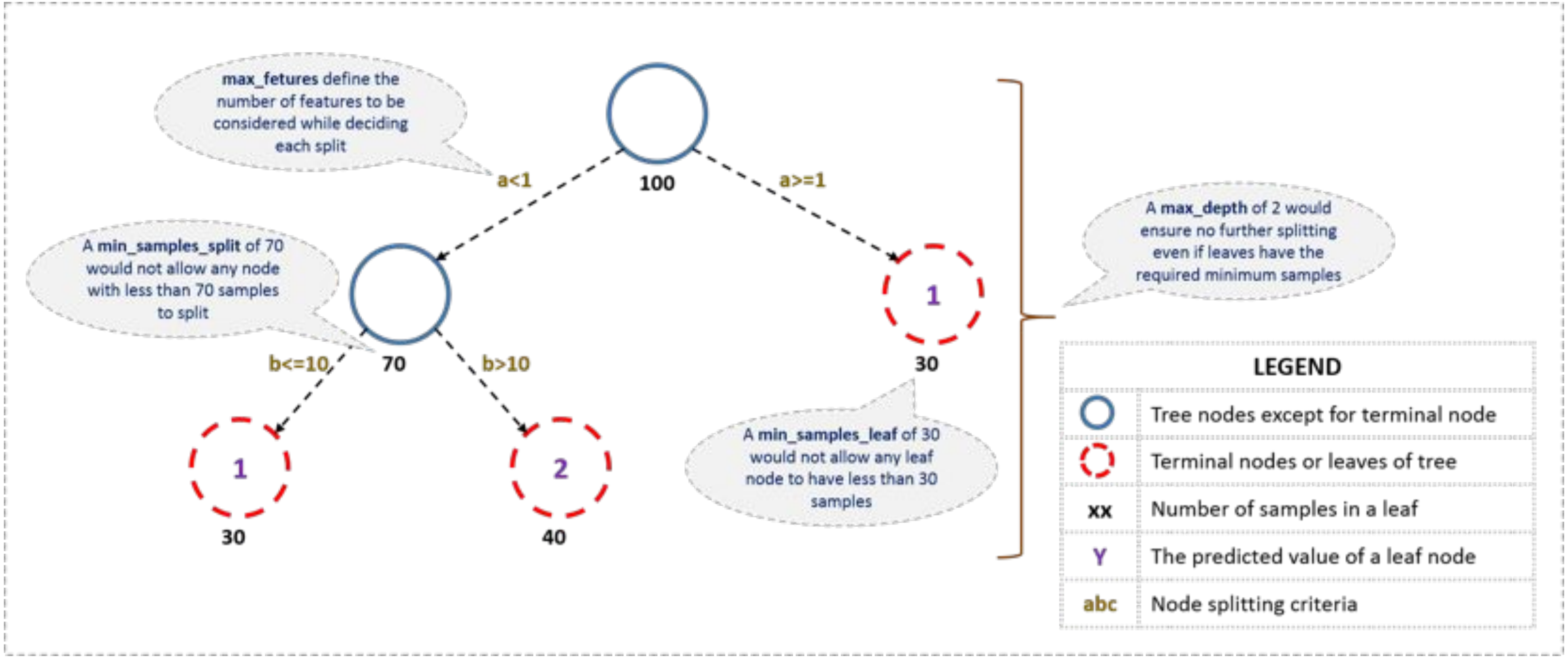
- Decision tree is very powerful in modeling complex patterns within the data.
- As the nodes increase, we can represent arbitrarily complex decision boundaries.





- Two major ways to prevent overfitting:
  - Set constraints on the tree size.
  - True pruning.

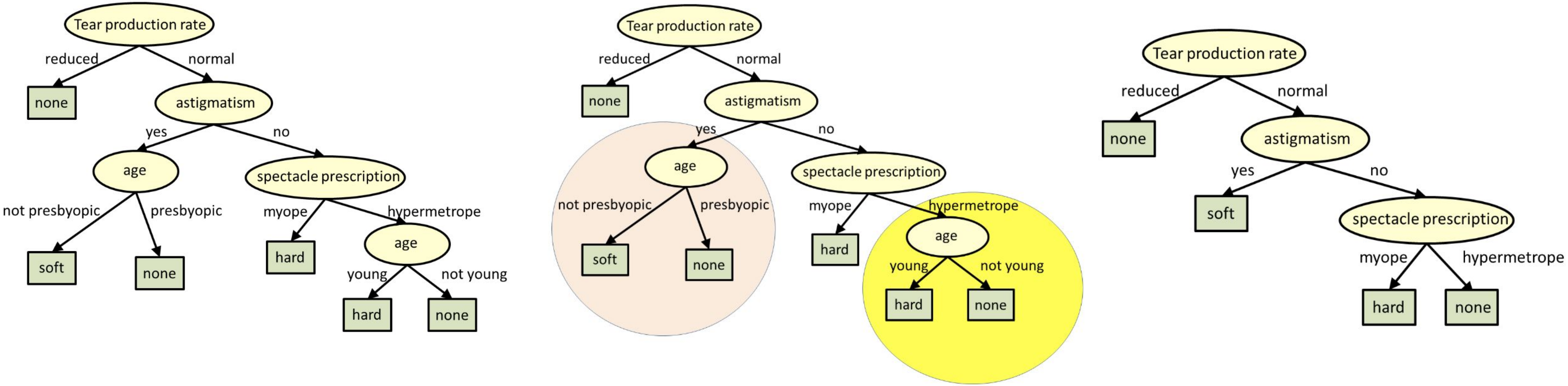
Setting Constraints on Tree Size



Foster School of Business

Pruning Decision Tree

- Grow the decision tree to a large depth.
- Start at the bottom and start removing leaves which are giving us negative returns based on a validation dataset.



Foster School of Business

Pros and Cons of Decision Tree

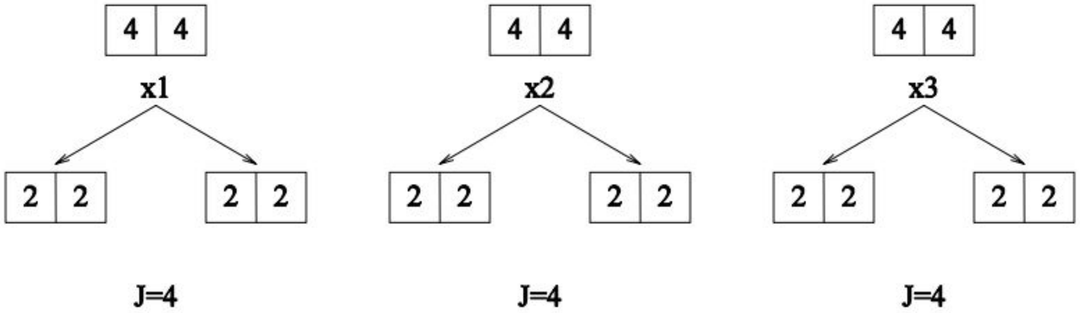
- Pros:
  - Easy to interpret.
  - Model nonlinear decision boundary.
  - Works well out of the box.

- Cons:
  - Tend to overfit if not properly tuned on validation data.
  - Sensitive to feature space rotation due to axis parallel decision boundaries.

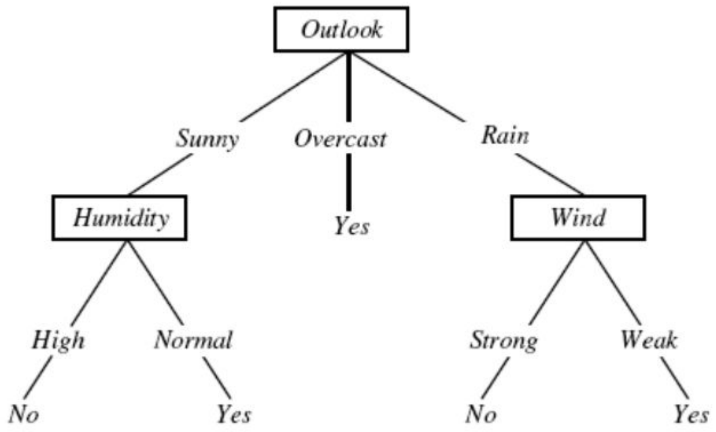
Learning Optimal Decision Tree is NP-complete

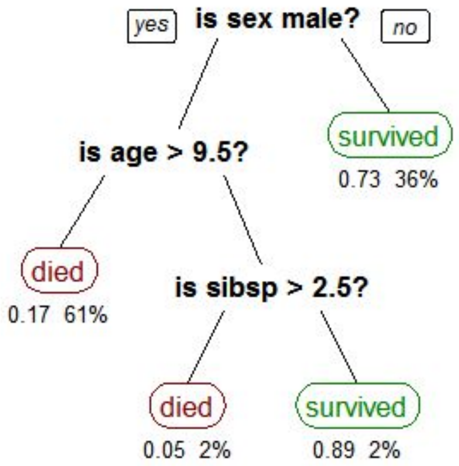
- **Optimal Decision Tree** finds the best partition of the data to achieve the global minimum error.
- Greed learning in constructing a Decision Tree does not guarantee optimality.

x_1	x_2	x_3	y
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0



Decision Tree Variants





C4.5

- Multiple way split
- Error based Pruning

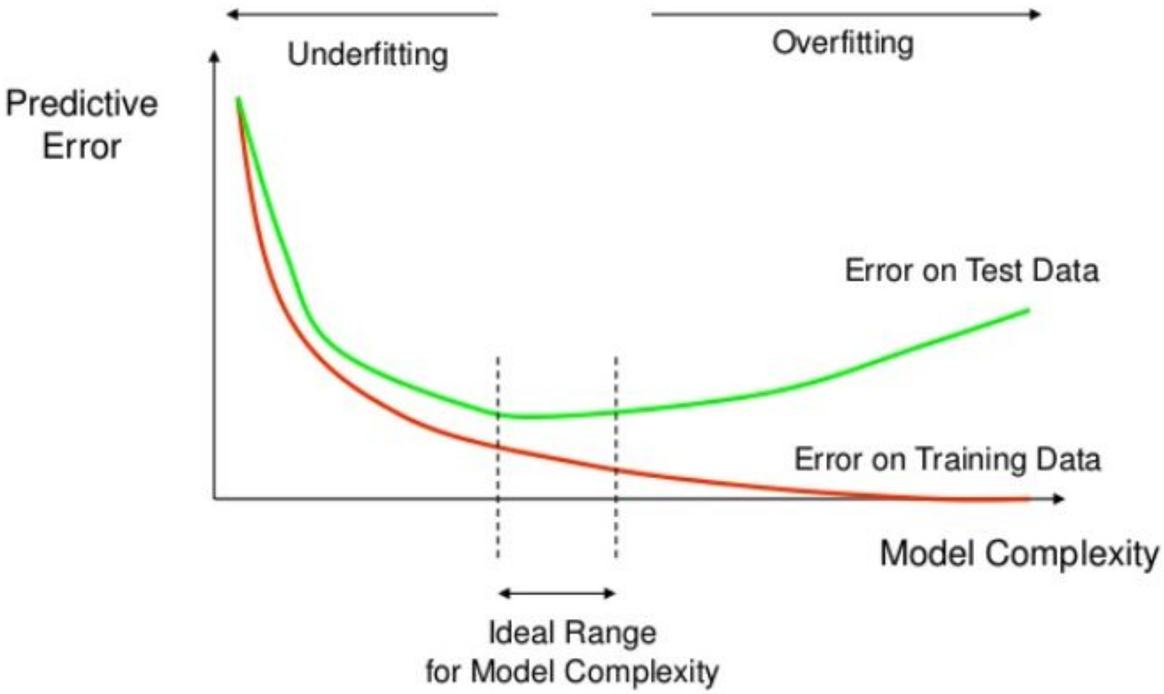
CART

- Binary split
- Cost-Complexity Pruning

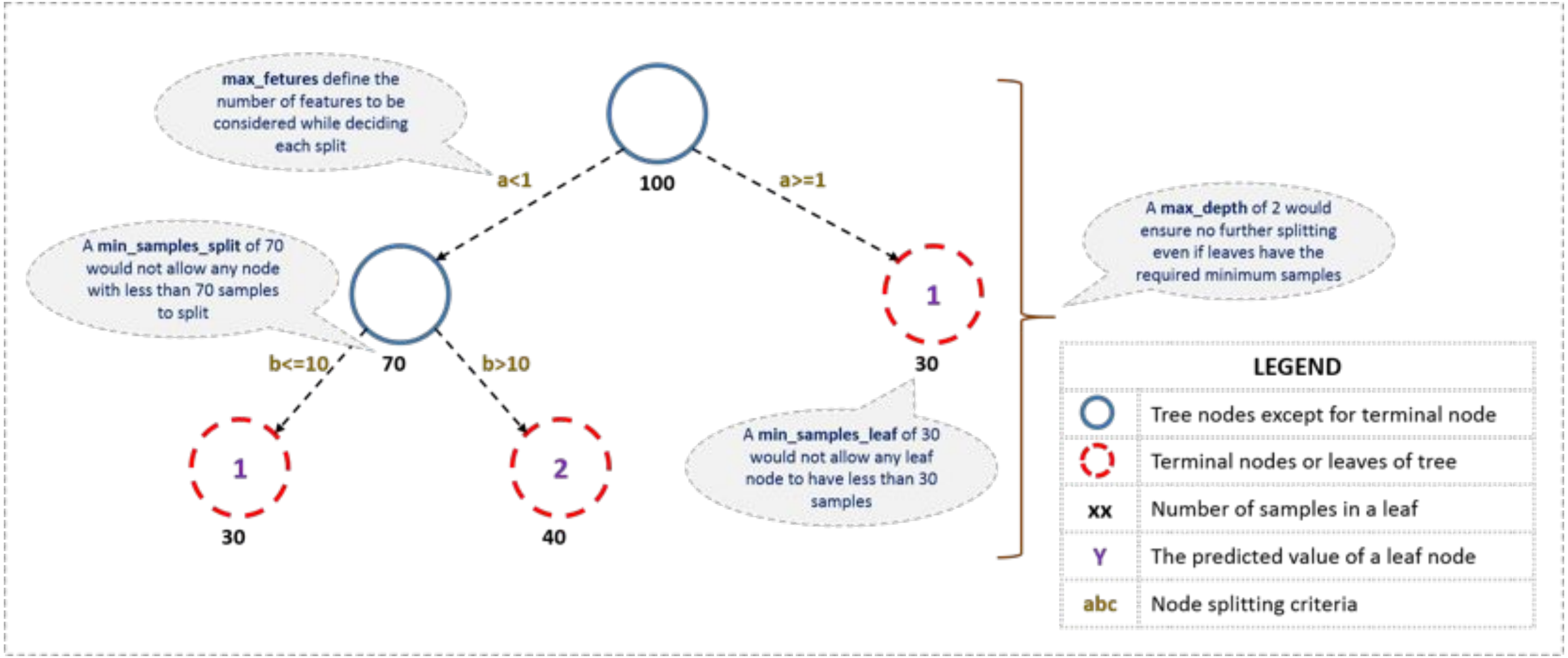
Hyper-parameter Tuning

Generalization

Machine Learning is all about **generalization** to future unseen data points.

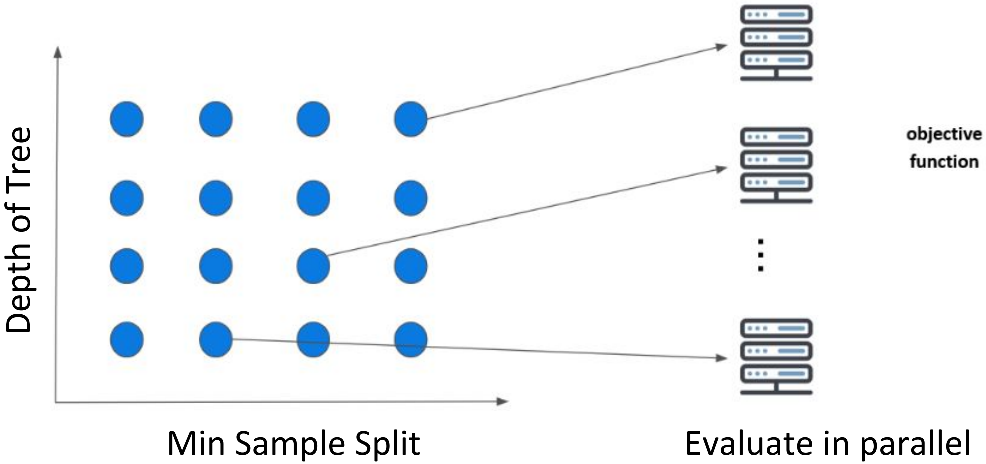


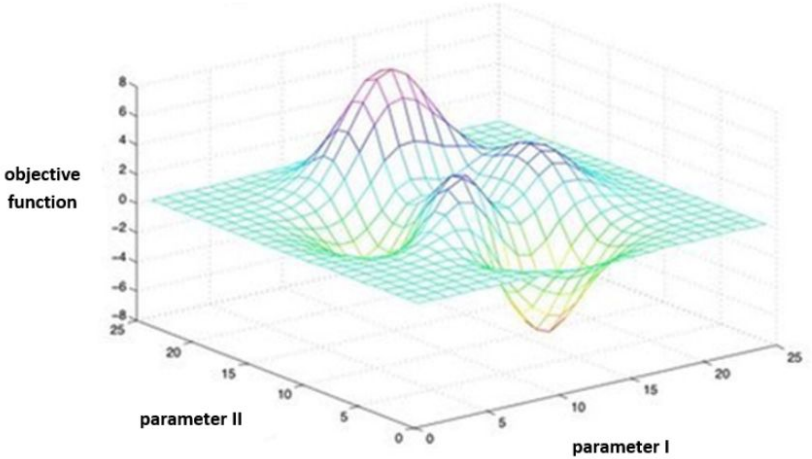
Hyper-parameters of Decision Tree



Grid Search

Find the best configuration for the hyper-parameters used in a ML model.





One-hot Encoding

What’s One-hot Encoding?

- Categorical variables cannot be easily handled for most of the machine learning algorithms, e.g. linear regression, support vector machine and neural networks.

Food Name	Categorical #	Calories
Apple	1	95
Chicken	2	231
Broccoli	3	50

$$2 \text{ Chickens} \overset{?}{=} 1 \text{ Apple} + 1 \text{ Broccoli}$$

- One-hot encoding is a process by which **categorical variables** are converted into a form that could be provided to ML algorithms to do a better job in prediction.

One-hot Encoding Example

Label Encoding

Food Name	Categorical #	Calories
Apple	1	95
Chicken	2	231
Broccoli	3	50

→

One Hot Encoding

Apple	Chicken	Broccoli	Calories
1	0	0	95
0	1	0	231
0	0	1	50

Deal with Categorical Features of High Cardinality

For categorical features of high cardinality, one-hot encoding could potentially creates a huge sparse feature vector, making it hard for ML to learn.

- Solution 1: one-hot encode a subset of the most common values of that variable and encode the rest as one value.
- Solution 2: substitute the value with the average of the target variable for each value in the training set.

Lab

Foster
School of Business