Machine Learning

Naive Bayes, Decision Tree

Dom Huh

Discriminative and Generative classifiers

Quick use of **generative** learning

Finding the feature similarities within labels (through probability density of data points)

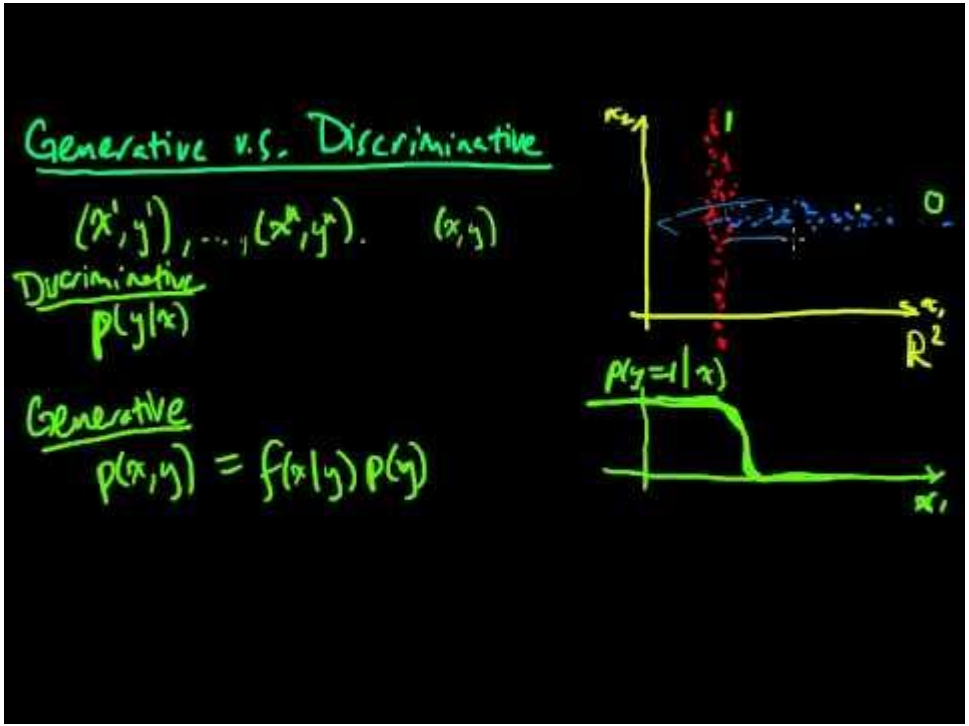
So far, we looked into **discriminative** learning.

Finding the feature distinctions between labels

More detail on:

https://ai.stanford.edu/~ang/papers/nips01-discriminativegenerative.pdf

Discriminative and Generative classifiers



Naive Bayes

Quick generative classifier useful in high-dimensional data sets

Probabilistic classifier with the following characteristics:

- Makes assumption that features are independent of each other
- Uses Bayes' Theorem
  - The probability of A given B is equal to the product of the probability of B given A and the probability of A over the probability of B
- Calculates the conditional probability

$$P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)}$$

Naive Bayes Theorem Machine Learning Classification

An Introduction

hackerearth

Visualization of Probability in Bayes: Likelihood Table

$$P(x \mid c) = P(Sunny \mid Yes) = 3/9 = 0.33$$

Frequency Table		Play Golf	
		Yes	No
Outlook	Sunny	3	2
	Overcast	4	0
	Rainy	2	3

Likelihood Table		Play Golf		
		Yes	No	
Outlook	Sunny	3/9	2/5	5/14
	Overcast	4/9	0/5	4/14
	Rainy	2/9	3/5	5/14
		9/14	5/14	

$$P(x) = P(Sunny) = 5/14 = 0.36$$

$$P(c) = P(Yes) = 9/14 = 0.64$$

Posterior Probability: $P(c \mid x) = P(Yes \mid Sunny) = 0.33 \times 0.64 \div 0.36 = 0.60$

Why use Naive Bayes Classifier

Highly scalable

Easy to implement

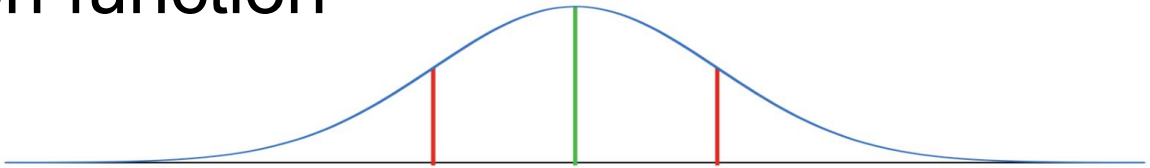
Need less training data

Real time predictions, LINEAR TIME DELAY (for predictions)

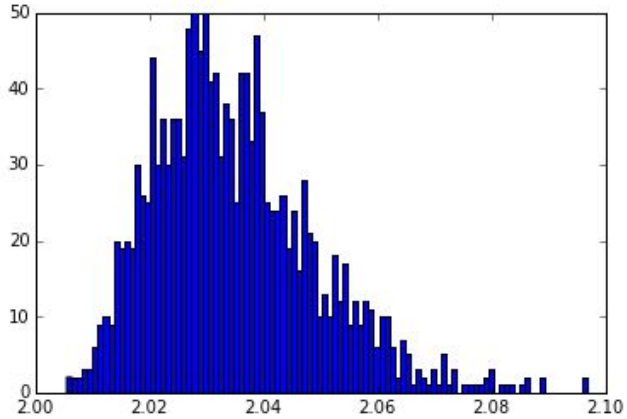
Handles both continuous and discrete data

Modelling distribution function

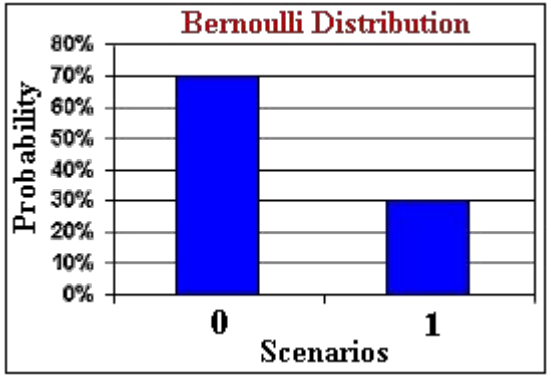
Gaussian



Multinomial

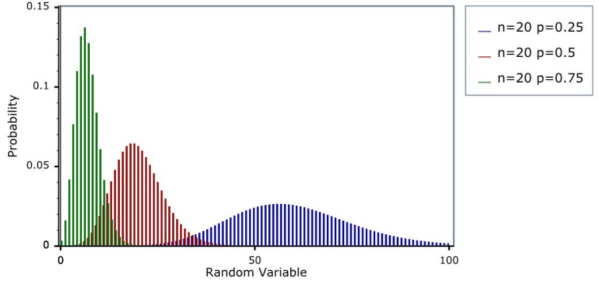


Bernoulli



Base

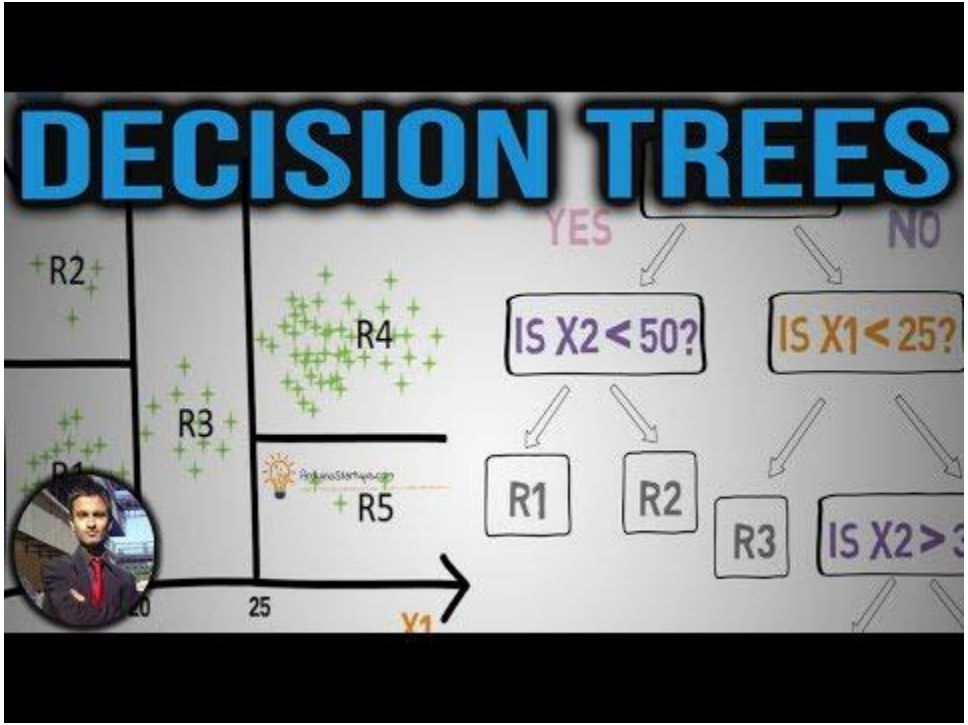
Base Discrete



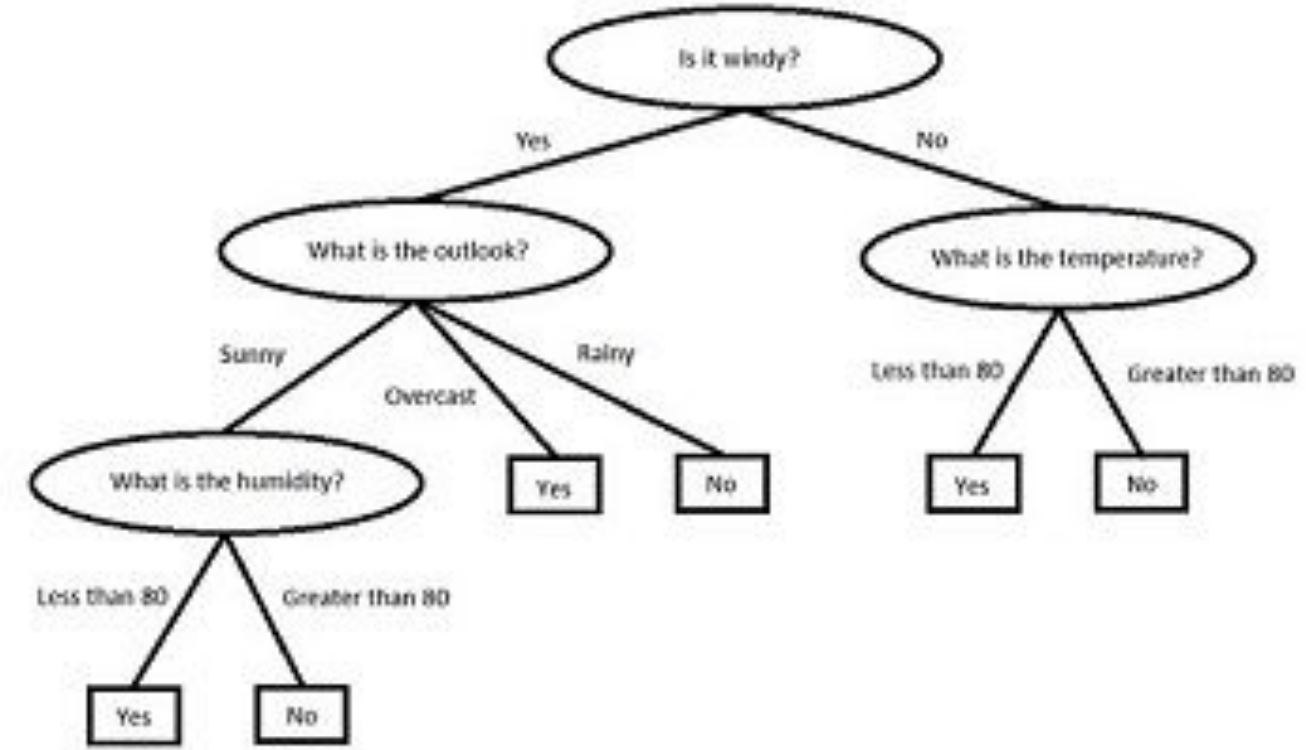
Complement

Naive Bayes Implementation

Decision Trees



Visualization: Decision Trees



Decision Tree (CART)

Incorporates explicit if-else rules to evaluate influential points used to predict the target value

Use as few questions as possible

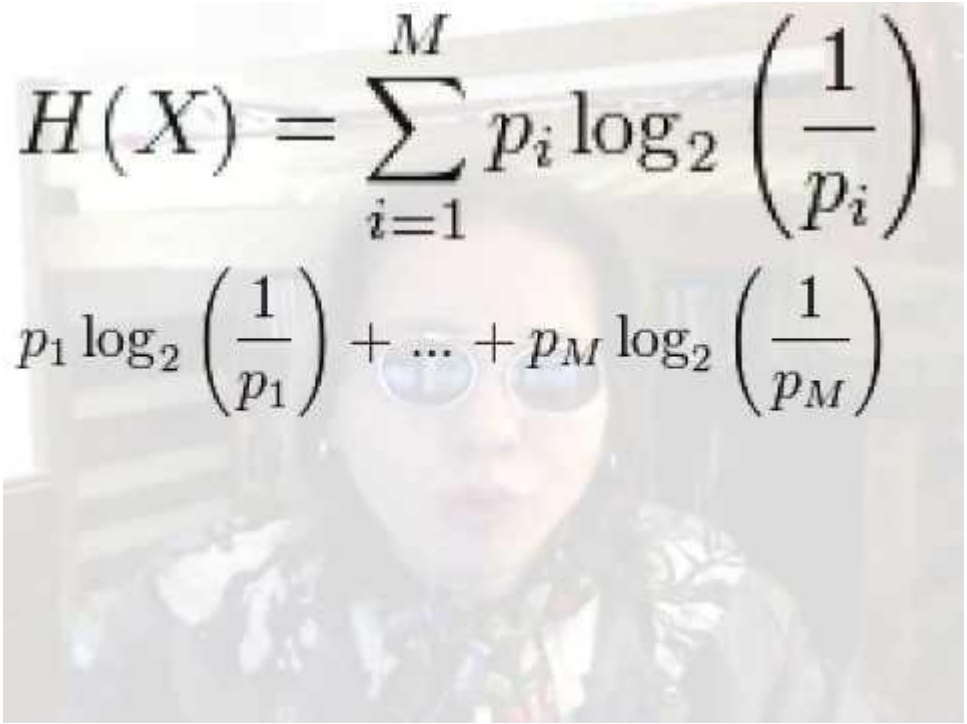
- Broad questions at earlier stages, and specify in lower levels (Root -> Leaf)
- Use binary ‘informative’ splits if possible w/ information gain/entropy
  - More pure nodes, but may still leave mixed nodes
    - Need to recursively layer more nodes

Mathematical understanding of informative splits

Shannon entropy

$$H(X) = \sum_{i=1}^{M} p_i \log_2 \left(\frac{1}{p_i} \right)$$

$$p_1 \log_2 \left(\frac{1}{p_1} \right) + \ldots + p_M \log_2 \left(\frac{1}{p_M} \right)$$



More in-depth:

https://www.youtube.com/watch?v=SXBG3RGr_Rc

Information Gain

Take into account the weight of each split by size and purity.

Problem: Degenerate splits and gain ratio

$$E(T,X) = \sum_{c \in X} P(c)E(c)$$

		Play Golf		
		Yes	No	
Outlook	Sunny	3	2	5
	Overcast	4	0	4
	Rainy	2	3	5
				14

E(PlayGolf, Outlook) = **P**(Sunny)*__E__(3,2) + **P**(Overcast)*__E__(4,0) + **P**(Rainy)*__E__(2,3)

= (5/14)*0.971 + (4/14)*0.0 + (5/14)*0.971

= 0.693

CART comparisons

Regression Trees- for continuous dependent variables

- Mean/Average used for prediction

Classification Trees - for categorical dependent variables

- Mode/Class used for prediction

Potential Problems with Decision Trees

If-else:

What does this lead to?

How do we fix this?

Potential Problems with Decision Trees

If-else structure:

What does this lead to? Overfitting

How do we fix this? Control model complexity (kind of)

Control model complexity-

Pruning(pre, post): Limit decision tree's growth w/

max_depth, max_leaf_nodes, min_sample_leaf… More on sklearn doc

Try to optimize **one** or all of these parameters.

What is wrong with doing this?

Visualize DT and plot feature importance chart*

Others

Gini Impurities/Chi Square (CHAID)

Entropy/Information Gain (CART/ID3/C4.5)

MARS algorithm

Reduction in variance

Decision Tree Implementation

A look into Ensemble Learning

Using multiple learning algorithms into one model