LEARN. DO. EARN

ACADGILD

MACHINE LEARNING WITH R

Website : http://www.acadgild.com
LinkedIn : https://www.linkedin.com/company/acadgild
Facebook : https://www.facebook.com/acadgild

Session 3: Naïve Bayes Classification

Agenda

Sl. No.	Agenda Topics
1.	Things We'd Like To Do
2.	Classification Problem
3.	Another Application
4.	Naïve Bayes Learning
5.	A Refresher on Probability
6.	Back to the Naïve Bayes Classifier
7.	Bayesian Theorem: Basics
8.	Deriving the Naïve Bayes
9.	Estimating Parameters For the Target Function
10.	Naïve Assumptions of Independence
11.	Again About Estimation

Sl. No.	Agenda Topics
12.	The Bayes Classifier
13.	Model Parameters
14.	The Naïve Bayes Model
15.	Why Is This Useful?
16.	Naïve Bayes Training
17.	Naïve Bayes Classification
18.	Another Example of the Naïve Bayes Classifier
19.	The Naive Bayes Classifier for Data Sets with Numerical Attribute Values
20.	Numeric Weather Data with Summary Statistics
21.	Output Probabilities
22.	Performance on a Test Set
23.	Naïve Bayes Assumption
24.	Exclusive-OR Example

Agenda

Sl. No.	Agenda Topics
25.	Evaluating Classification Algorithms
26.	Types of Errors
27.	Sensitivity and Specificity
28.	The ROC Space
29.	The ROC Curve
30.	ROC Analysis
31.	Holdout Estimation
32.	Repeated Holdout Method
33.	Cross-Validation
34.	Leave-One-Out Cross-Validation
35.	Leave-One-Out-CV and Stratification
36.	Points to Remember

Things We’d Like To Do

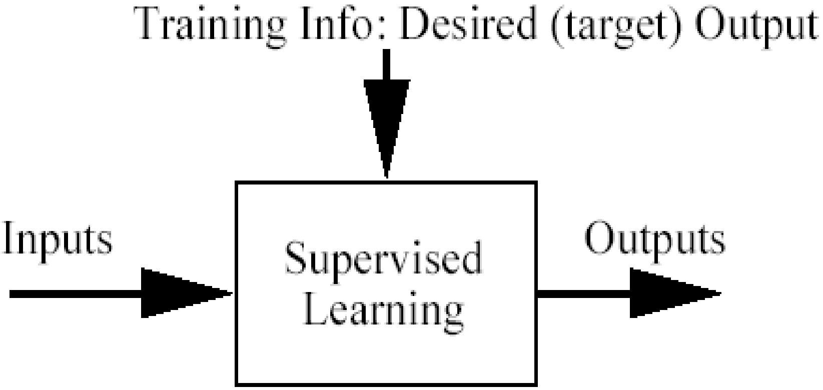
- Spam Classification
  - Given an email, predict whether it is spam or not

- Medical Diagnosis
  - Given a list of symptoms, predict whether a patient has disease X or not

- Weather
  - Based on temperature, humidity, etc… predict if it will rain tomorrow

Classification Problem

- Training data: examples of the form (d,h(d))
  - where d are the data objects to classify (inputs)
  - and h(d) are the correct class info for d, h(d)∈{1,…K}
- Goal: given dnew, provide h(dnew)



Training Info: Desired (target) Output

Inputs

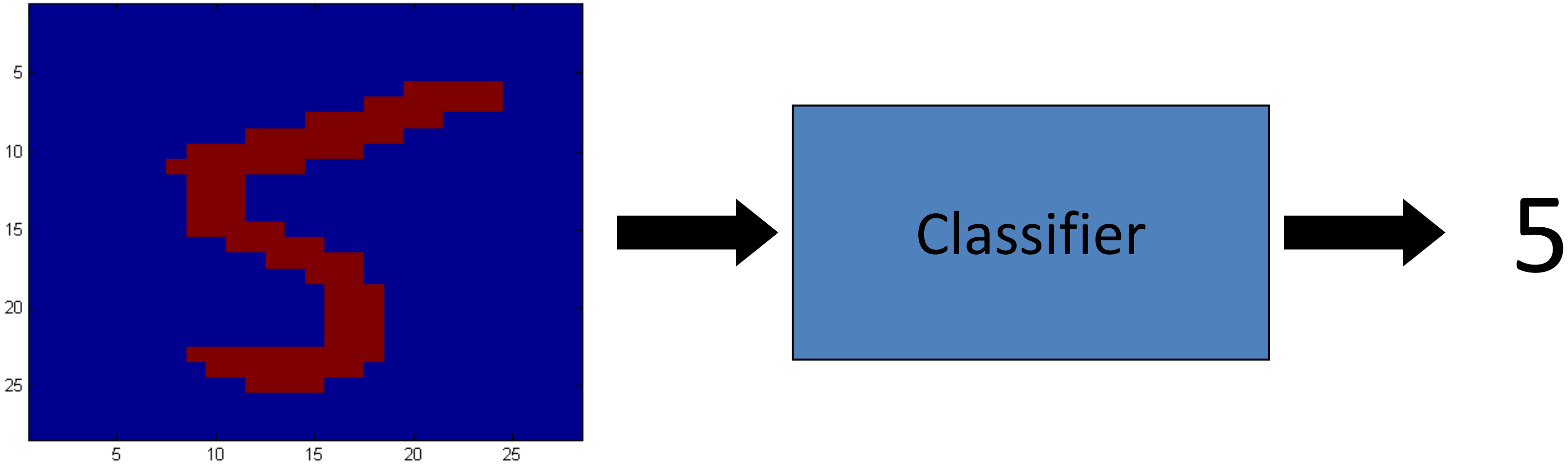
Supervised Learning

Outputs

Error = (target output - actual output)

Another Application

- Digit Recognition



Classifier

5

- X1,…,Xn ∈ {0,1} (Black vs. White pixels)
- Y ∈ {5,6} (predict whether a digit is a 5 or a 6)

Naïve Bayes Learning

Learning Algorithm: Naïve Bayes

Target Function: $\gamma(d) = c$

$$c_{MAP} = \underset{c \in C}{\arg\max}\ \hat{P}(c \mid d) = \underset{c \in C}{\arg\max}\ \hat{P}(c) \prod_{1 \le k \le n_d} \hat{P}(t_k \mid c)$$

$$c_{MAP} = \underset{c \in C}{\arg\max}\ P(c \mid d) = \underset{c \in C}{\arg\max}\ P(c) P(d \mid c)$$

The generative process:

$P(c)$ a priori probability, of choosing a category

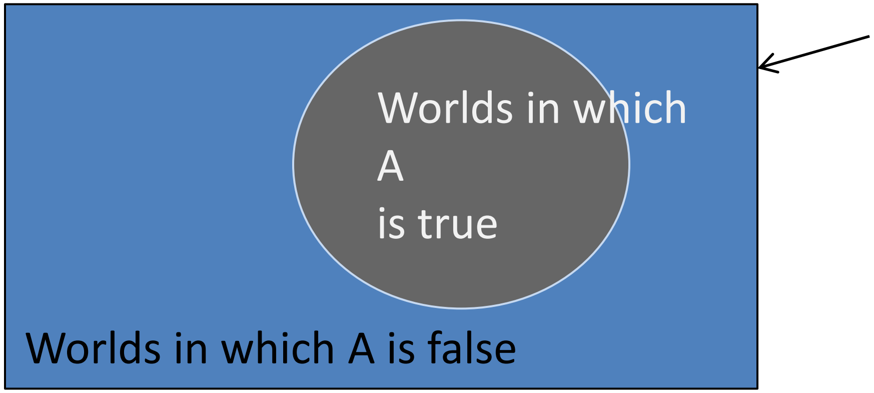
$P(d \mid c)$ the cond. prob. of generating *d*, given the fixed *c*

$P(c \mid d)$ a posteriori probability that *c* generated *d*

A Refresher on Probability

Visualizing Probability

- A is a random variable that denotes an uncertain event
  - Example: A = “I’ ll get an A+ in the final exam”
- P(A) is “the fraction of possible worlds where A is true”



Event space of all possible worlds. Its area is 1.

P(A) = Area of the blue circle.

Slide: Andrew W. Moore

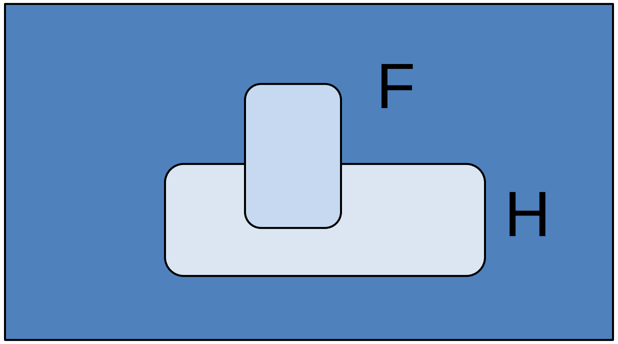
Axioms and Theorems of Probability

- Axioms:
  - 0 <= P(A) <= 1
  - P(True) = 1
  - P(False) = 0
  - P(A or B) = P(A) + P(B) – P(A and B)

- Theorems:
  - P(not A) = P(~A) = 1 – P(A)
  - P(A) = P(A ^ B) + P(A ^ ~B)

Conditional Probability

- P(A|B) = the probability of A being true, given that we know that B is true



H = “I have a headache”
F = “Coming down with flu”

P(H) = 1/10
P(F) = 1/40
P(H/F) = 1/2

Headaches are rare and flu even rarer, but if you got flu, there is a 50-50 chance you’ll have a headache.

Slide: Andrew W. Moore

Deriving the Bayes Rule

Conditional Probability: $$P(A|B) = \frac{P(A \wedge B)}{P(B)}$$

Chain rule: $$P(A \wedge B) = P(A|B)P(B)$$

$$P(A \wedge B) = P(B \wedge A) = P(B|A)P(A)$$

Bayes Rule: $$P(B|A) = \frac{P(A|B)P(B)}{P(A)}$$

Back to the Naïve Bayes Classifier

Bayesian Theorem: Basics

- Let X be a data sample
- Let H be a hypothesis that X belongs to class C
- Classification is to determine P(H|X), the probability that the hypothesis holds given the observed data sample X

Example: customer X will buy a computer given that know the customer’s age and income

Bayesian Theorem: Basics (Contd.)

- P(H) (prior probability), the initial probability

  E.g., X will buy computer, regardless of age, income, …

- P(X): probability that sample data is observed

- P(X|H) (posteriori probability), the probability of observing the sample X, given that the hypothesis holds

  E.g., Given that X will buy computer, the prob. that X is 31..40, medium income

Deriving the Naïve Bayes

$$P(B|A) = \frac{P(A|B)P(B)}{P(A)}$$ (Bayes Rule)

Given two classes c_1, c_2 and the document d'

$$P(c_1 | d') = \frac{P(c_1)P(d'|c_1)}{P(d')} \qquad P(c_2 | d') = \frac{P(c_2)P(d'|c_2)}{P(d')}$$

We are looking for a c_i that maximizes the a-posteriori $P(c_i | d')$

$P(d')$ (the denominator) is the same in both cases

Thus: $$c_{MAP} = \underset{c \in C}{\arg\max}\, P(c)P(d|c)$$

Estimating Parameters For the Target Function

We are looking for the estimates $\hat{P}(c)$ and $\hat{P}(d \mid c)$

P(c) is the fraction of possible worlds where c is true.

$$\hat{P}(c) = \frac{N_c}{N}$$

N – number of all documents

N$_c$ – number $P($ of documents in class c

---

d is a vector in the space X where each dimension is a term:

$$P(d \mid c) = P(\langle t_i, t_2, \ldots, t_{n_d} \rangle \mid c)$$

By using the chain rule: $P(A \wedge B) = P(A \mid B)P(B)$ we have:

$$P(\langle t_i, t_2, \ldots, t_{n_d} \rangle \mid c) = P(t_1 \mid t_2, \ldots, t_{n_d}, c)P(t_2, \ldots, t_{n_d}, c)$$

$$= \ldots$$

Naïve Assumptions of Independence

- All attribute values are independent of each other given the class. (conditional independence assumption)
- The conditional probabilities for a term are the same independent of position in the document.
- We assume the document is a “bag-of-words”.

$$P(d\mid c)=P(\langle t_i,t_2,\ldots,t_{n_d}\rangle\mid c)=\prod_{1\le k\le n_d}P(t_k\mid c)$$

Finally, we get the target function of Slide 8:

$$c_{MAP}=\underset{c\in C}{\arg\max}\,\hat{P}(c\mid d)=\underset{c\in C}{\arg\max}\,\hat{P}(c)\prod_{1\le k\le n_d}\hat{P}(t_k\mid c)$$

Again About Estimation

For each term, t, we need to estimate P(t|c)

$$\hat{P}(t|c) = \frac{T_{ct}}{\sum_{t' \in V} T_{ct'}}$$

T_{ct} is the count of term t in all documents of class c

Because an estimate will be 0 if a term does not appear with a class in the training data, we need smoothing:

Laplace Smoothing

$$\hat{P}(t|c) = \frac{T_{ct} + 1}{\sum_{t' \in V} (T_{ct'} + 1)} = \frac{T_{ct} + 1}{(\sum_{t' \in V} T_{ct'}) + |V|}$$

|V| is the number of terms in the vocabulary

The Bayes Classifier

- a good strategy is to predict:

$$\arg\max_{Y} P(Y|X_1, \ldots, X_n)$$

(for example: what is the probability that the image represents a 5, given its pixels?)

- So … How do we compute that?

The Bayes Classifier (Contd.)

- Use Bayes Rule!

Likelihood Prior

$$P(Y|X_1,\ldots,X_n) = \frac{P(X_1,\ldots,X_n|Y)P(Y)}{P(X_1,\ldots,X_n)}$$

Normalization Constant

- Why did this help? Well, we think that we might be able to specify how features are “generated” by the class label

The Bayes Classifier (Contd.)

- Let’s expand this for digit recognition task:

$$
\begin{aligned}
P(Y=5|X_1,\ldots,X_n) &= \frac{P(X_1,\ldots,X_n|Y=5)P(Y=5)}{P(X_1,\ldots,X_n|Y=5)P(Y=5)+P(X_1,\ldots,X_n|Y=6)P(Y=6)} \\
P(Y=6|X_1,\ldots,X_n) &= \frac{P(X_1,\ldots,X_n|Y=6)P(Y=6)}{P(X_1,\ldots,X_n|Y=5)P(Y=5)+P(X_1,\ldots,X_n|Y=6)P(Y=6)}
\end{aligned}
$$

- To classify, we’ll simply compute these two probabilities and predict based on which one is greater

Model Parameters

- For the Bayes classifier, we need to “learn” two functions, the likelihood and the prior
- How many parameters are required to specify the prior for our digit recognition example?

Model Parameters (Contd.)

- How many parameters are required to specify the likelihood?

(Supposing that each image is 30x30 pixels)

Model Parameters (Contd.)

- The problem with explicit modeling P(X1,…,Xn|Y) is that there are usually too many parameters:
  - We’ll run out of space
  - We’ll run out of time
  - And we’ll need lot of training data (which is usually not available)

The Naïve Bayes Model

- The Naïve Bayes Assumption: Assume that all features are independent given the class label Y
- Equation

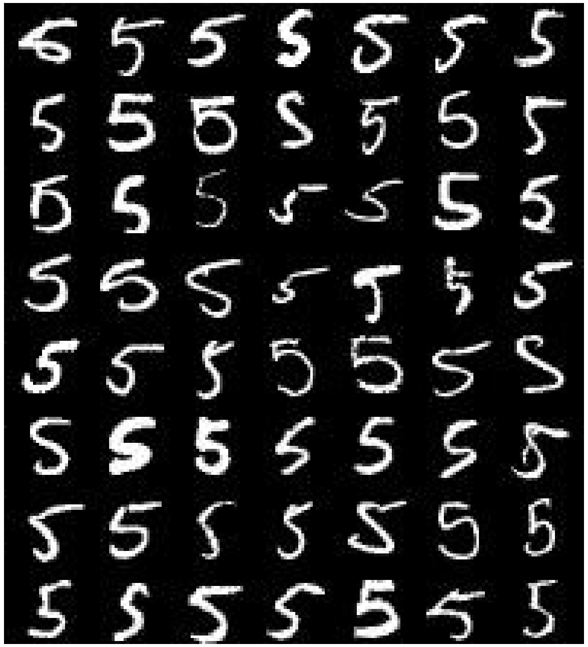
$$P(X_1, \ldots, X_n|Y) = \prod_{i=1}^{n} P(X_i|Y)$$

Why Is This Useful?

- # of parameters for modeling P(X1,…,Xn|Y):
  - 2(2n-1)
- # of parameters for modeling P(X1|Y),…,P(Xn|Y)
  - 2n

Naïve Bayes Training

- Now that we’ve decided to use a Naïve Bayes classifier, we need to train it with some data:



MNIST Training Data

Naïve Bayes Training (Contd.)

- Training in Naïve Bayes is easy:
  - Estimate P(Y=v) as the fraction of records with Y=v

$$P(Y=v) = \frac{Count(Y=v)}{\#\ records}$$

  - Estimate P(Xi=u|Y=v) as the fraction of records with Y=v for which Xi=u

$$P(X_i=u|Y=v) = \frac{Count(X_i=u \wedge Y=v)}{Count(Y=v)}$$

(This corresponds to Maximum Likelihood estimation of model parameters)

Naïve Bayes Training (Contd.)

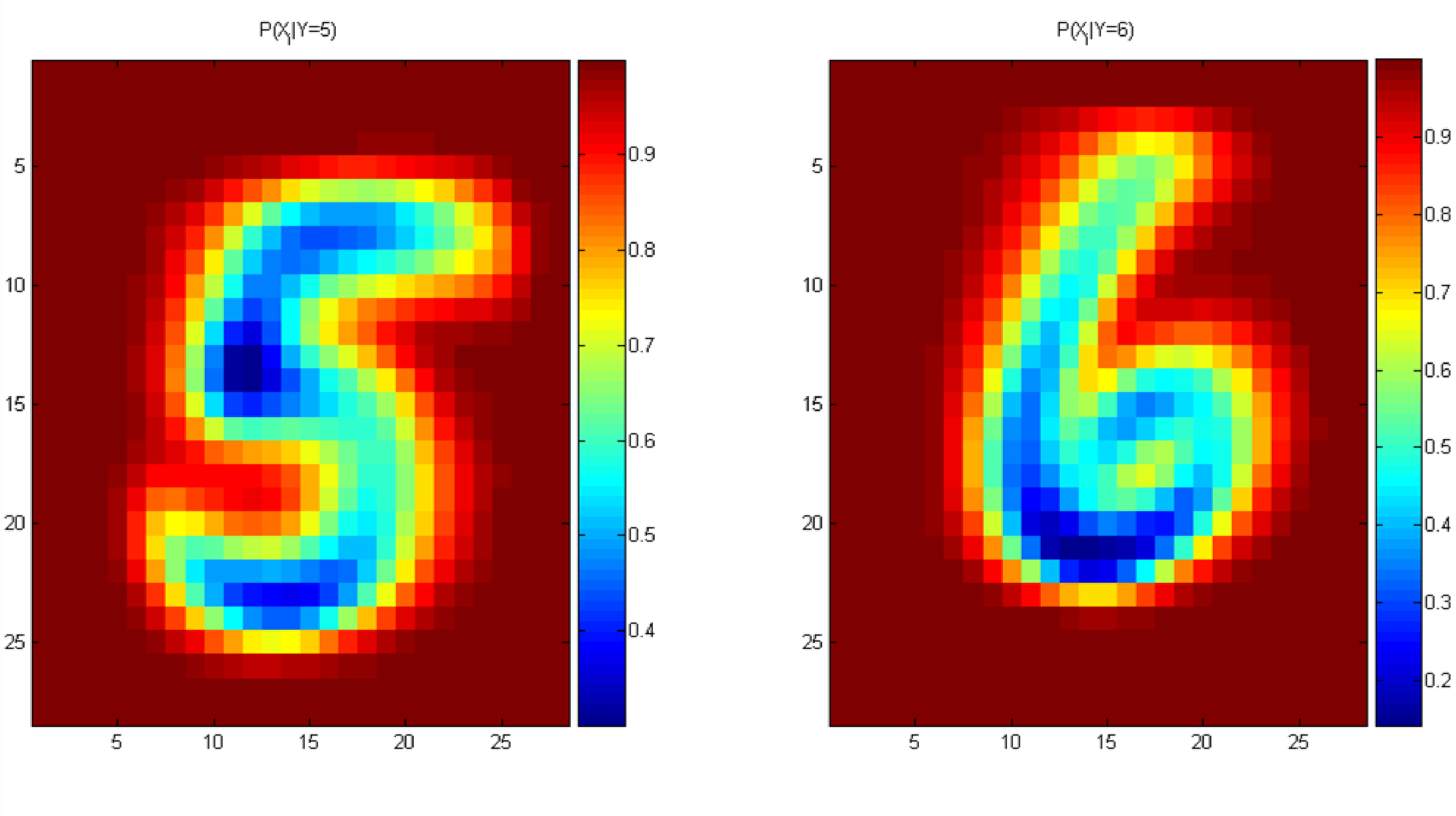
- In practice, some of these counts can be zero
- Fix this by adding "virtual" counts:

$$P(X_i = u|Y = v) = \frac{Count(X_i = u \wedge Y = v) + 1}{Count(Y = v) + 2}$$

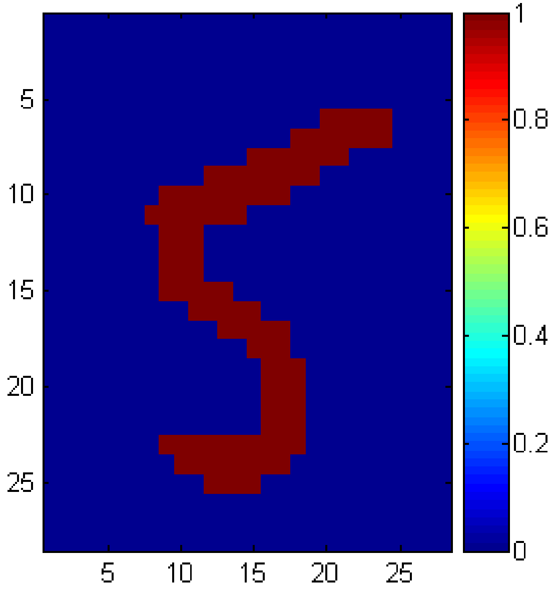
- (This is like putting a prior on parameters and doing MAP estimation instead of MLE)
- This is called Smoothing

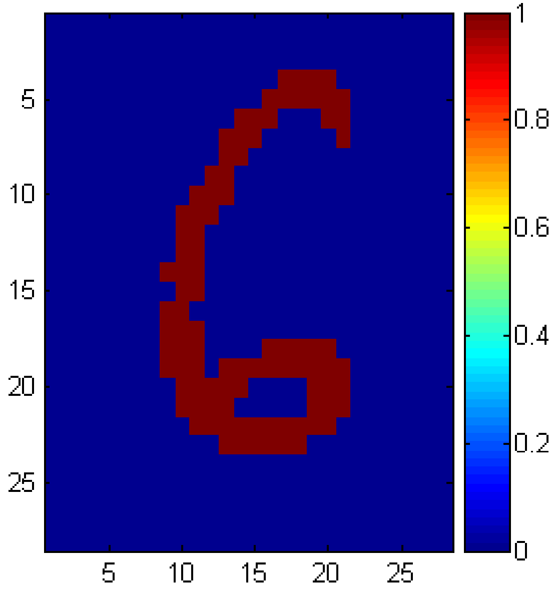
Naïve Bayes Training (Contd.)

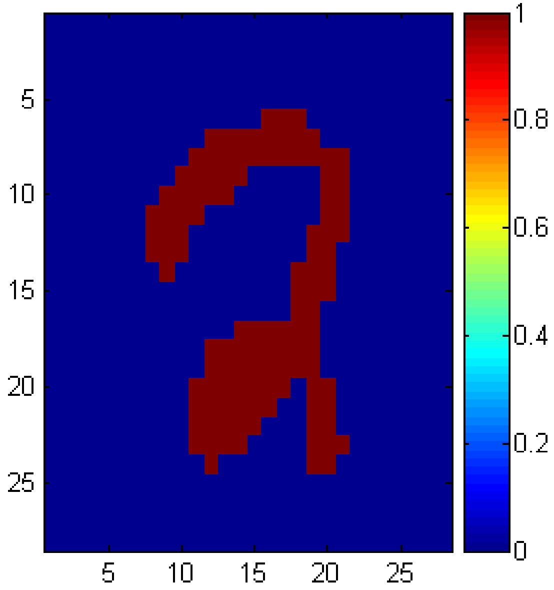
- For binary digits, training amounts to averaging all of the training fives together and all of the training sixes together.



Naïve Bayes Classification


Prediction: 5 with prob 1


Prediction: 6 with prob 9.997968e-001


Prediction: 5 with prob 8.632034e-001

Another Example of the Naïve Bayes Classifier

The weather data, with counts and probabilities

outlook			temperature			humidity			windy			play	
	yes	no		yes	no		yes	no		yes	no	yes	no
sunny	2	3	hot	2	2	high	3	4	false	6	2	9	5
overcast	4	0	mild	4	2	normal	6	1	true	3	3		
rainy	3	2	cool	3	1								
sunny	2/9	3/5	hot	2/9	2/5	high	3/9	4/5	false	6/9	2/5	9/14	5/14
overcast	4/9	0/5	mild	4/9	2/5	normal	6/9	1/5	true	3/9	3/5		
rainy	3/9	2/5	cool	3/9	1/5								

A new day

outlook	temperature	humidity	windy	play
sunny	cool	high	true	?

Another Example of the Naïve Bayes Classifier

- Likelihood of yes

$$= \frac{2}{9} \times \frac{3}{9} \times \frac{3}{9} \times \frac{3}{9} \times \frac{9}{14} = 0.0053$$

- Likelihood of no

$$= \frac{3}{5} \times \frac{1}{5} \times \frac{4}{5} \times \frac{3}{5} \times \frac{5}{14} = 0.0206$$

- Therefore, the prediction is No

The Naive Bayes Classifier for Data Sets with Numerical Attribute Values

- One common practice to handle numerical attribute values is to assume normal distributions for numerical attributes.

Numeric Weather Data with Summary Statistics

The numeric weather data with summary statistics

outlook			temperature			humidity			windy			play	
	yes	no		yes	no		yes	no		yes	no	yes	no
sunny	2	3		83	85		86	85	false	6	2	9	5
overcast	4	0		70	80		96	90	true	3	3		
rainy	3	2		68	65		80	70					
				64	72		65	95					
				69	71		70	91					
				75			80						
				75			70						
				72			90						
				81			75						
sunny	2/9	3/5	mean	73	74.6	mean	79.1	86.2	false	6/9	2/5	9/14	5/14
overcast	4/9	0/5	std dev	6.2	7.9	std dev	10.2	9.7	true	3/9	3/5		
rainy	3/9	2/5											

Numeric Weather Data with Summary Statistics (Contd.)

- Let x_1, x_2, …, x_n be the values of a numerical attribute in the training data set.

$$\mu = \frac{1}{n}\sum_{i=1}^{n} x_i$$

$$\sigma = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \mu)^2$$

$$f(w) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(w-\mu)^2}{\sigma^2}}$$

Numeric Weather Data with Summary Statistics (Contd.)

For example,

$$f\left(\text{temperature} = 66 \mid \text{Yes}\right) = \frac{1}{\sqrt{2\pi}\left(6.2\right)} e^{-\frac{(66-73)^2}{2(6.2)^2}} = 0.0340$$

- Likelihood of Yes = $\frac{2}{9} \times 0.0340 \times 0.0221 \times \frac{3}{9} \times \frac{9}{14} = 0.000036$

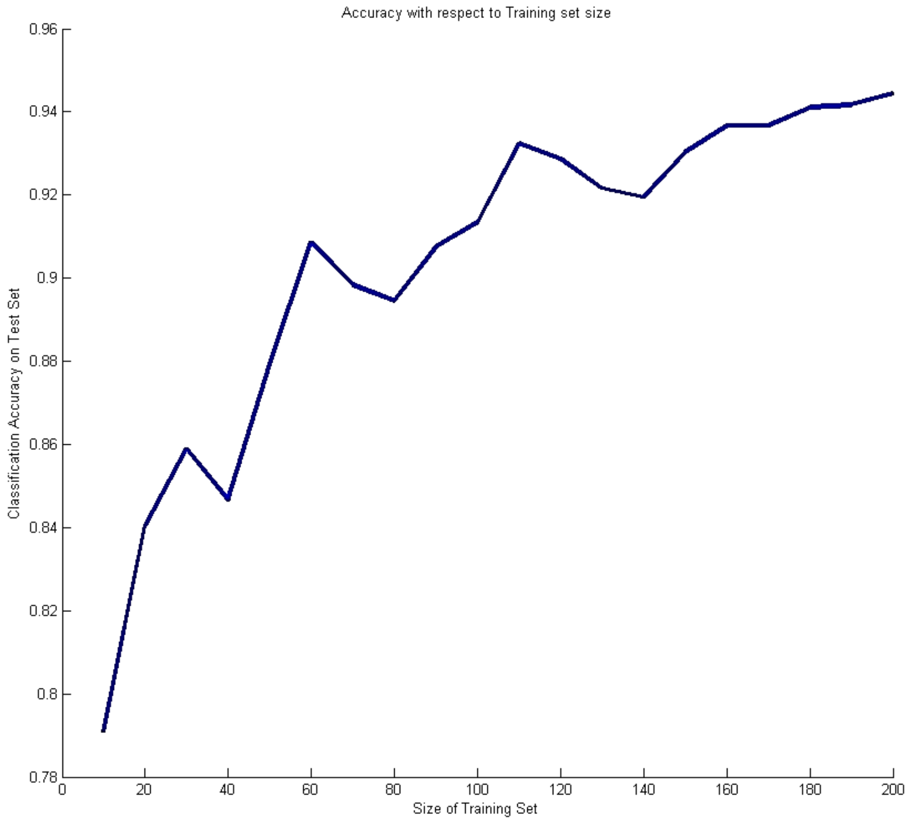
- Likelihood of No = $\frac{3}{5} \times 0.0291 \times 0.038 \times \frac{3}{5} \times \frac{5}{14} = 0.000136$

Output Probabilities

- The advantage of Naïve Bayes (and generative models in general) is that it returns probabilities
  - These probabilities can tell us how confident the algorithm is

Performance on a Test Set

- Naïve Bayes is often a good choice if you don’t have much training data!



Naïve Bayes Assumption

- Recalling the Naïve Bayes assumption:
  - all features are independent given the class label Y
- Does this hold good for the digit recognition problem?

Exclusive-OR Example

- For an example where conditional independence fails:
  - Y=XOR(X_1,X_2)

X_1	X_2	P(Y=0|X_1,X_2)	P(Y=1|X_1,X_2)
0	0	1	0
0	1	0	1
1	0	0	1
1	1	1	0

Exclusive-OR Example (Contd.)

- Actually, the Naïve Bayes assumption is almost never true
- But Naïve Bayes often performs well even when its assumptions do not hold good.

Evaluating Classification Algorithms

- Suppose, you have designed a new classifier.
- You give it to me, and I try it on my image dataset
- I tell you that it achieved 95% accuracy on my data.
- Is your technique a success?

Types of Errors

- But suppose that
  - The 95% is the correctly classified pixels
  - Only 5% of the pixels are actually edges
  - It misses all the edge pixels
- How do we count the effect of different types of error?

Types of Errors (Contd.)

		Prediction	
		Edge	Not edge
Ground Truth	Edge	True Positive	False Negative
	Not Edge	False Positive	True Negative

Types of Errors (Contd.)

Two parts to each: whether you got it correct or not, and what you guessed. For example for a particular pixel, our guess might be labelled…

True Positive

Did we get it correct? True, we did get it correct.

What did we say? We said 'positive', i.e. edge.

or maybe it was labelled as one of the others

False Negative

Did we get it correct? False, we did not get it correct.

What did we say? We said 'negative, i.e. not edge.

Sensitivity and Specificity

Count up the total number of each label (TP, FP, TN, FN) over a large dataset. In ROC analysis, we use two statistics:

$$\textbf{Sensitivity} = \frac{\text{TP}}{\text{TP+FN}}$$

Can be considered as the likelihood of spotting a positive case when presented with one.

Or the proportion of edges we find.

$$\textbf{Specificity} = \frac{\text{TN}}{\text{TN+FP}}$$

Can be considered as the likelihood of spotting a negative case when presented with one.

Or the proportion of non-edges that we find

Sensitivity and Specificity (Contd.)

$$\text{Sensitivity} = \frac{TP}{TP+FN} = ? \qquad \text{Specificity} = \frac{TN}{TN+FP} = ?$$

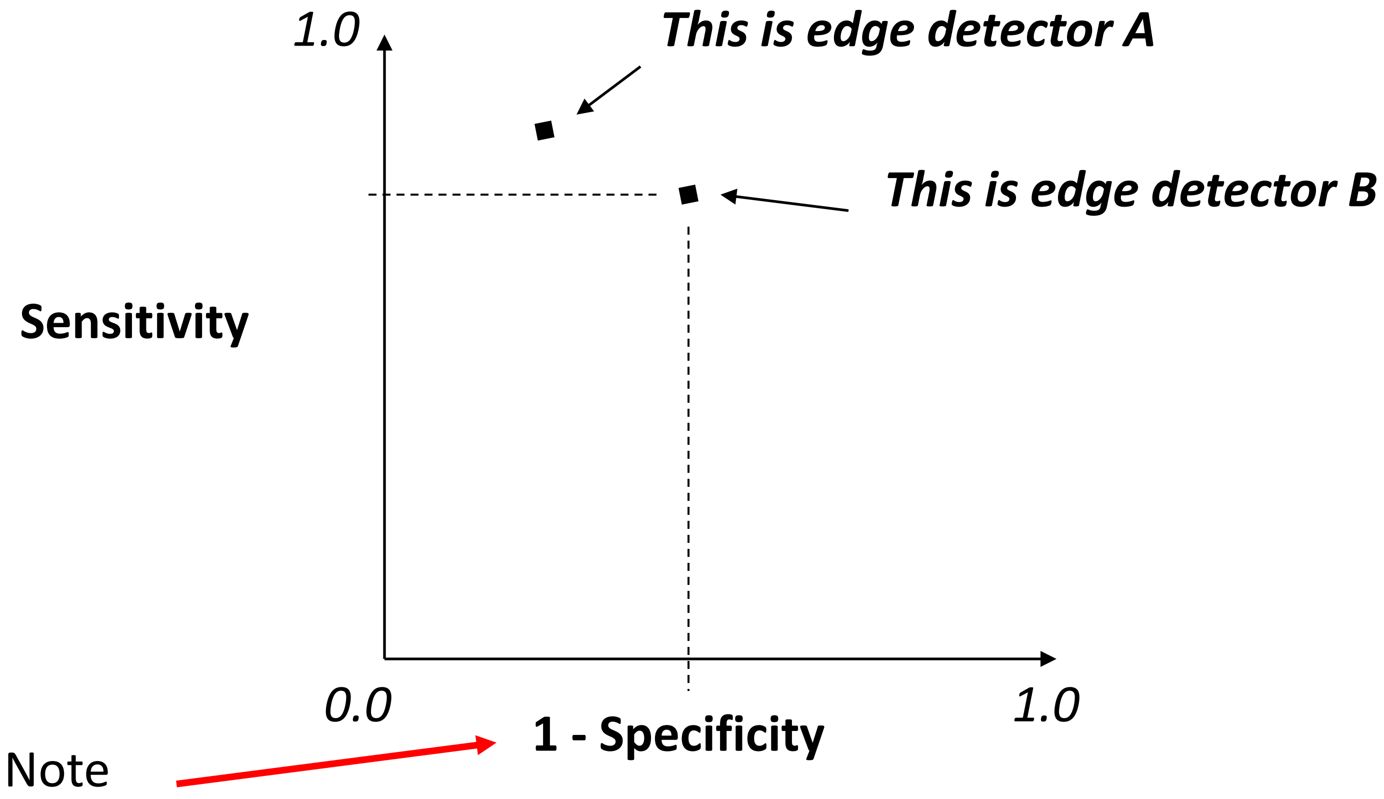
		Prediction	
		1	0
Ground Truth	1	60	30
	0	80	20

60+30 = 90 cases in the dataset were class 1 (edge)

80+20 = 100 cases in the dataset were class 0 (non-edge)

90+100 = 190 examples (pixels) in the data overall

The ROC Space



This is edge detector A

This is edge detector B

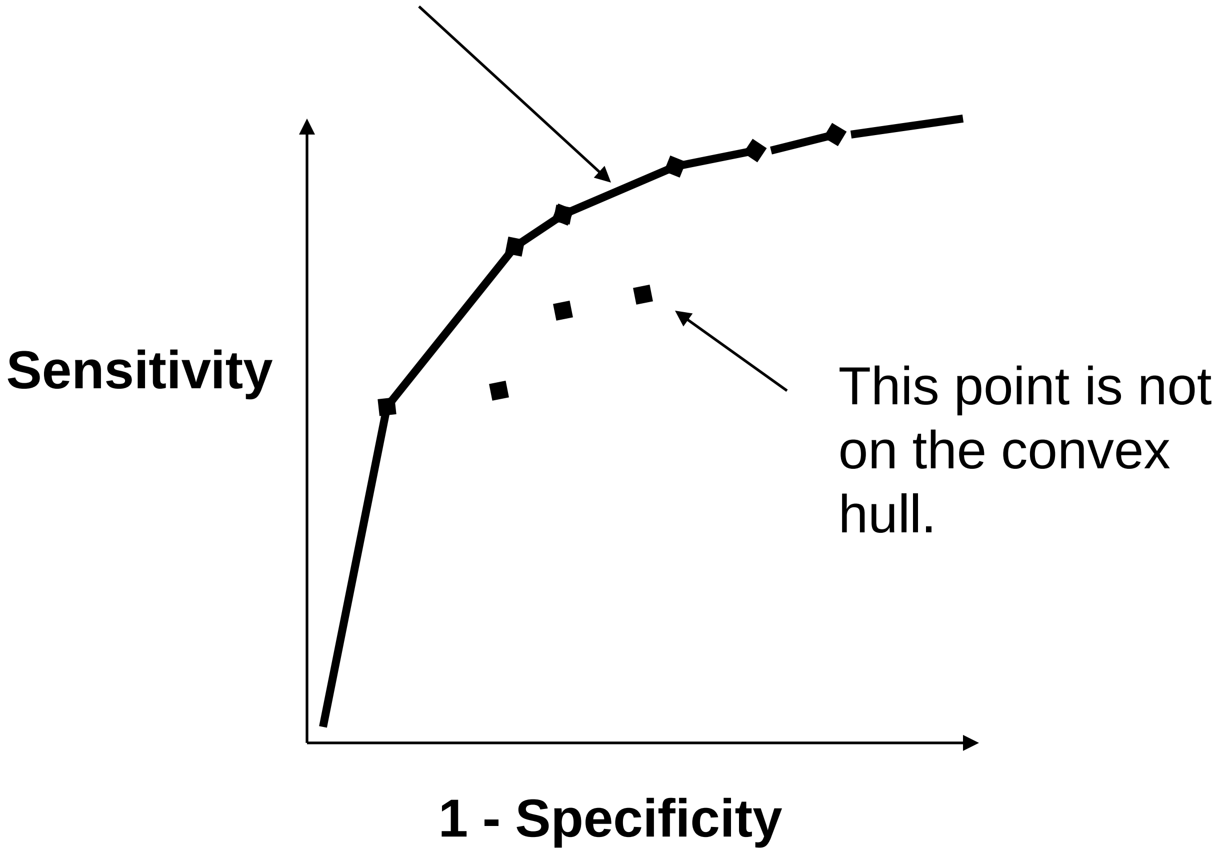
Sensitivity

1 - Specificity

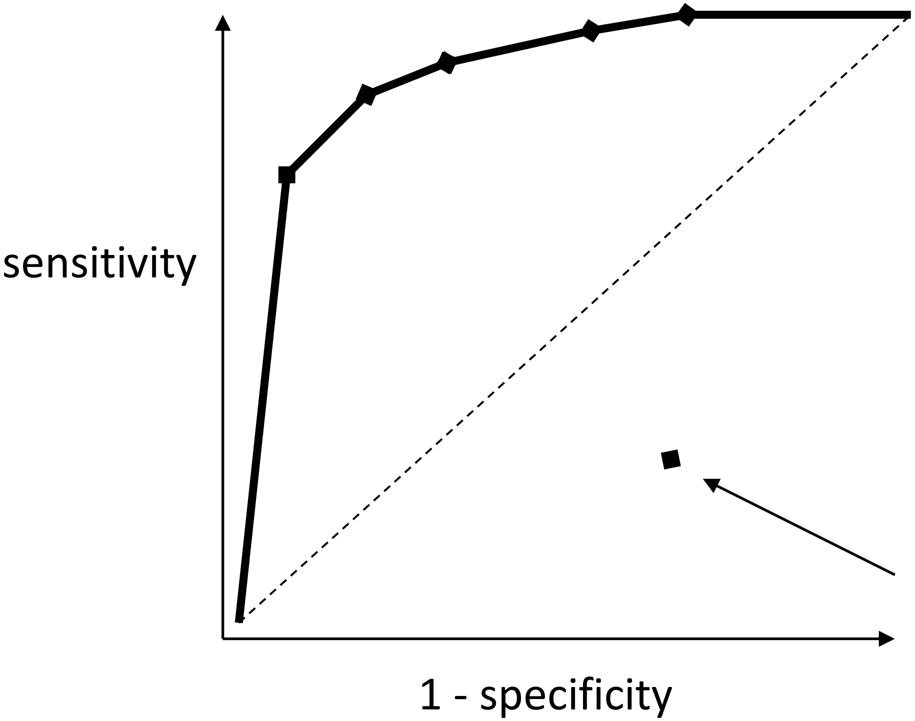
Note

The ROC Curve

Draw a ‘convex hull’ around many points:



ROC Analysis



All the optimal detectors lie on the convex hull.

Which of these is best depends on the ratio of edges to non-edges, and the different cost of misclassification

Any detector on this side can lead to a better detector by flipping its output.

Points to Remember : You should always quote sensitivity and specificity for your algorithm, if possible plotting an ROC graph. Also, remember that any statistic you quote should be an average over a suitable range of tests for your algorithm.

Holdout Estimation

- What do you do if the amount of data is limited?
- The holdout method reserves a certain amount for testing and uses the remainder for training

(Usually, one third for testing, the rest for training)

Holdout Estimation (Contd.)

- Problem: the samples might not be representative

Example: class might be missing in the test data

- Advanced version uses stratification
  - Ensures that each class is represented with approximately equal proportions in both subsets

Repeated Holdout Method

- Repeat process with different subsamples (more reliable)
  - In each iteration, a certain proportion is randomly selected for training (possibly with stratification)
  - The error rates on different iterations are averaged to yield an overall error rate

Repeated Holdout Method (contd.)

- It is still not optimum: the different test sets overlap
  - Can we prevent overlapping?
  - Yes, this is possible

Cross-Validation

- Cross-validation avoids overlapping test sets
  - First step: split data into k subsets of equal size
  - Second step: use each subset in turn for testing the remainder for training
- Called k-fold cross-validation

Cross-Validation (Contd.)

- Often the subsets are stratified before the cross-validation is performed
- The error estimates are averaged to yield an overall error estimate

Cross-Validation (Contd.)

- Standard method for evaluation: stratified ten-fold cross-validation
- Why ten?
  - Empirical evidence supports this as a good choice to get an accurate estimate
  - There is also some theoretical evidence for this
- Stratification reduces the estimate’s variance
- Even better: repeated stratified cross-validation

E.g. ten-fold cross-validation is repeated ten times and results are averaged (reduces the variance)

Leave-One-Out Cross-Validation

- Leave-One-Out is a particular form of cross-validation
  - Set number of folds to number of training instances
  - i.e., for n training instances, build classifier n times
- Makes best use of the data
- Involves no random subsampling
- Computationally very expensive

(exception: NN)

Leave-One-Out-CV and Stratification

- Disadvantage of Leave-One-Out-CV is that stratification is not possible
  - It guarantees a non-stratified sample because there is only one instance in the test set!

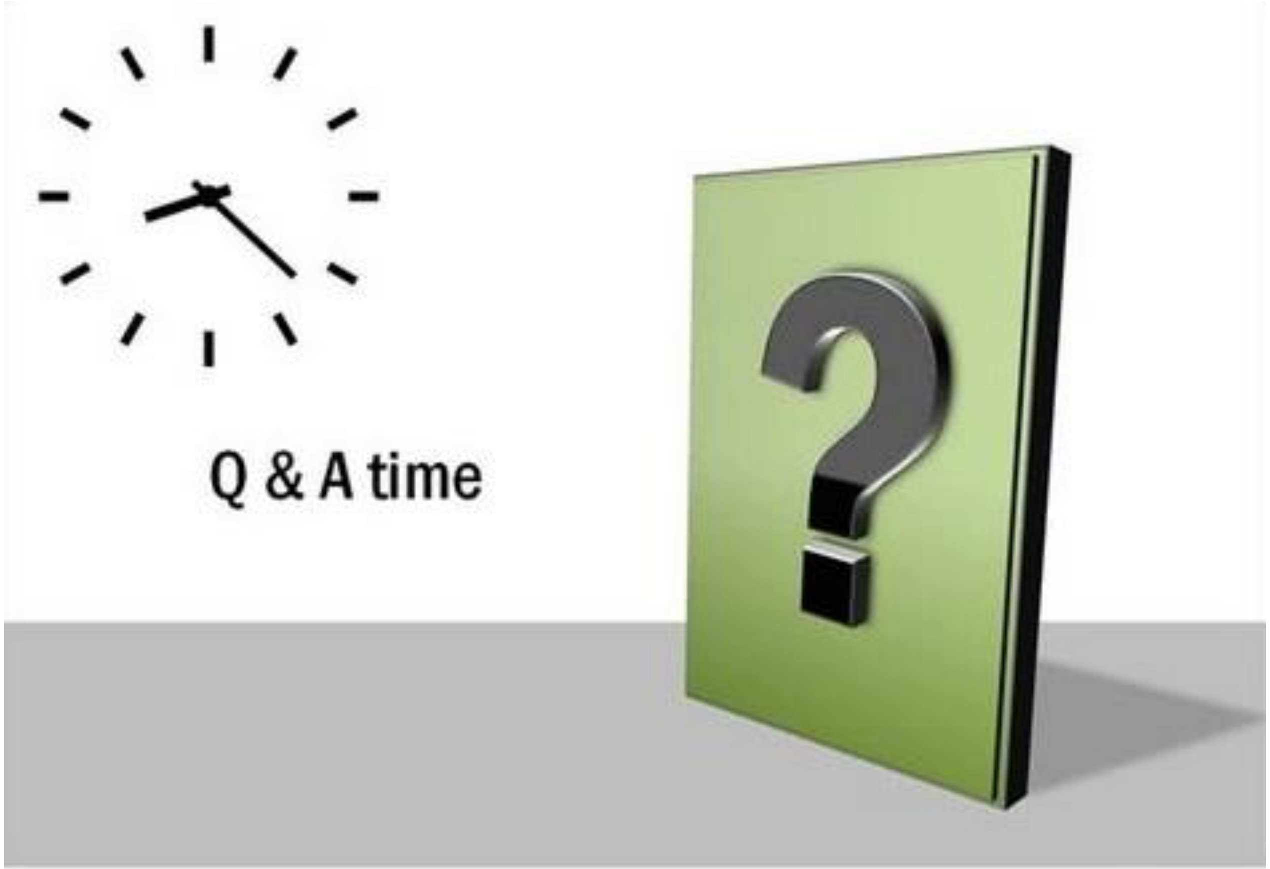
Points to Remember

- Bayes' rule can be turned into a classifier
- Maximum A Posteriori (MAP) hypothesis estimation incorporates prior knowledge; Max Likelihood doesn't
- Naive Bayes Classifier is a simple but effective Bayesian classifier for vector data (i.e. data with several attributes) which assumes that attributes are independent, given the class.
- Bayesian classification is a generative approach to classification

Next Class: Decision Tree

Sl. No.	Agenda Topics
1.	Why Decision Tree?
2.	Key Requirements
3.	Definition
4.	What is a Decision Tree?
5.	Predicting Commute Time
6.	Inductive Learning
7.	Decision Trees as Rules
8.	Decision Tree as a Rule Set
9.	How to Create a Decision Tree
10.	Sample Experience Table
11.	Choosing Attributes
12.	Decision Tree Algorithms

Sl. No.	Agenda Topics
13.	Identifying the Best Attributes
14.	ID3 Heuristic
15.	Entropy
16.	ID3
17.	Pruning Trees
18.	Pre-pruning
19.	Post-pruning
20.	Subtree Replacement
21.	Subtree Raising
22.	Error Propagation
23.	Example of a Decision Tree
24.	Another Example of Decision Tree
25.	Decision Tree Classification Task


Q & A time

Get in Touch with ACADGILD

Contact Info:

- **Website : http://www.acadgild.com**
- **LinkedIn : https://www.linkedin.com/company/acadgild**
- **Facebook : https://www.facebook.com/acadgild**
- **Support: support@acadgild.com**