Data Analytics

Course Taught at IIFT

Day 10: Pattern Classification and Bayesian Classifier

Dr. Tanujit Chakraborty

www.ctanujit.org

Today's discussion…

- Introduction to Machine Learning (also called pattern recognition/classification)
- Machine Learning Techniques
    - Supervised and Unsupervised Learning
- Bayesian Classifier
    - Principle of Bayesian classifier
    - Bayes' theorem of probability
- Naïve Bayes Classifier

INTRODUCTION TO MACHINE LEARNING

Machine Learning

- “Machine learning is the field of study that gives computers the ability to learn without being explicitly programmed” - Arthur L. Samuel, AI pioneer.

- Role of Machine Learning: Building efficient algorithms to

  1. solve an optimization problem
  2. represent and evaluate the model for inference
  3. create programs that can automatically learn rules from data

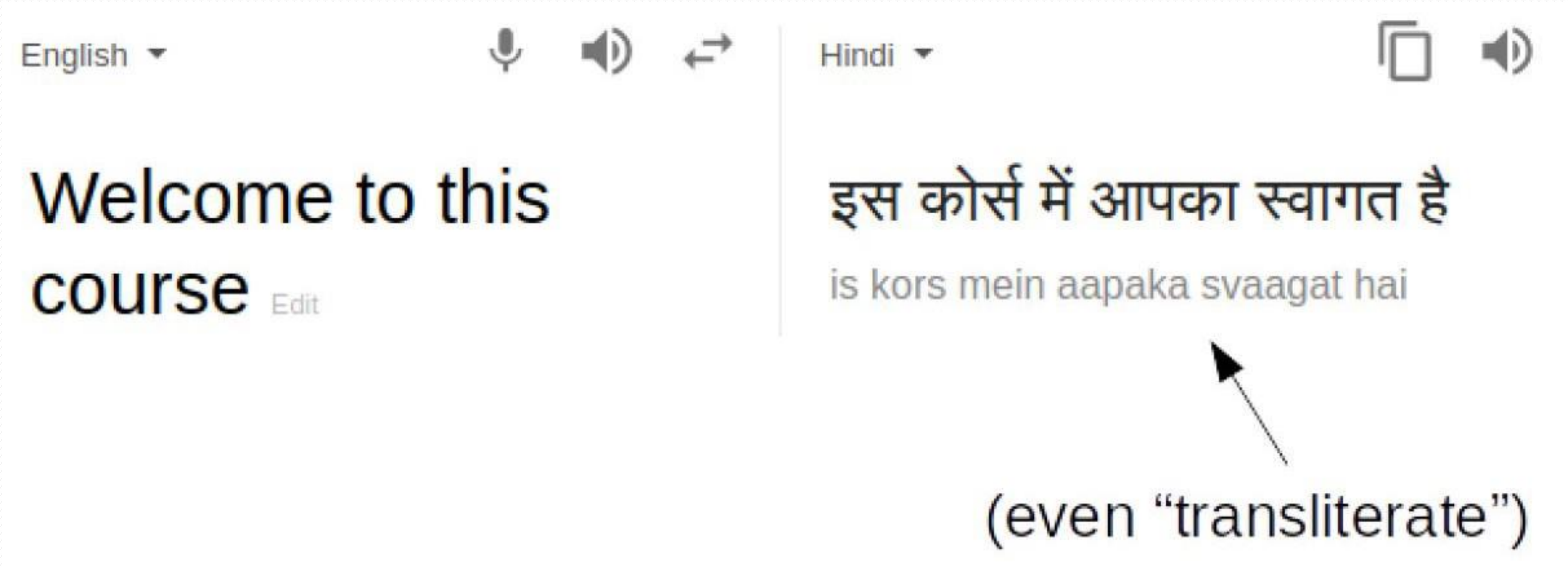
- Remember: “Prediction is very difficult, especially if it's about the future" - Niels Bohr, Father of Quantum.

Examples of ML in Data Analytics

- Life Science: Predicting tumor cells as benign or malignant
- Banking: Classifying credit card transactions as legitimate or fraudulent
- Prediction: Weather, voting, political dynamics, etc.
- Entertainment: Categorizing news stories as finance, weather, entertainment, sports, etc.
- Social media: Identifying the current trend and future growth

Machine Learning helps Natural Language Processing

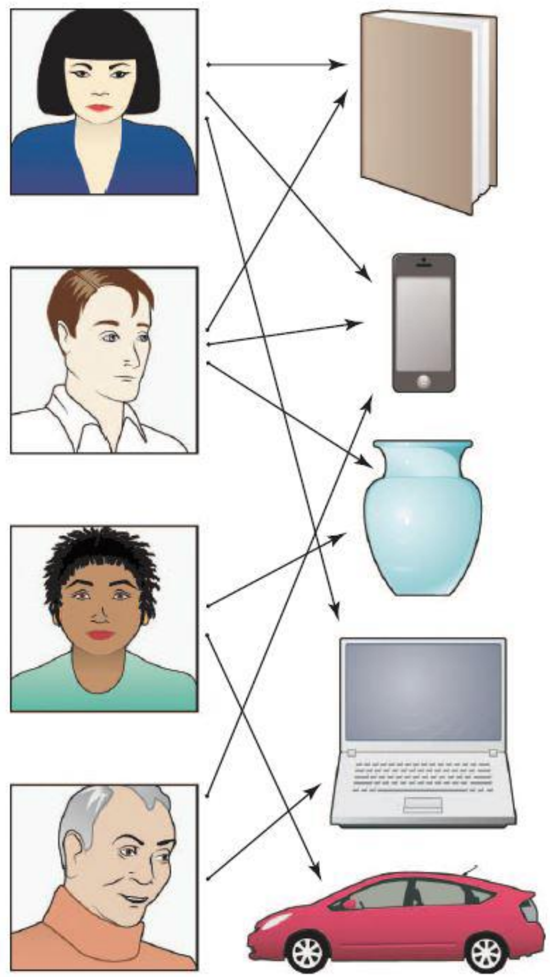
ML algorithm can learn to translate text…



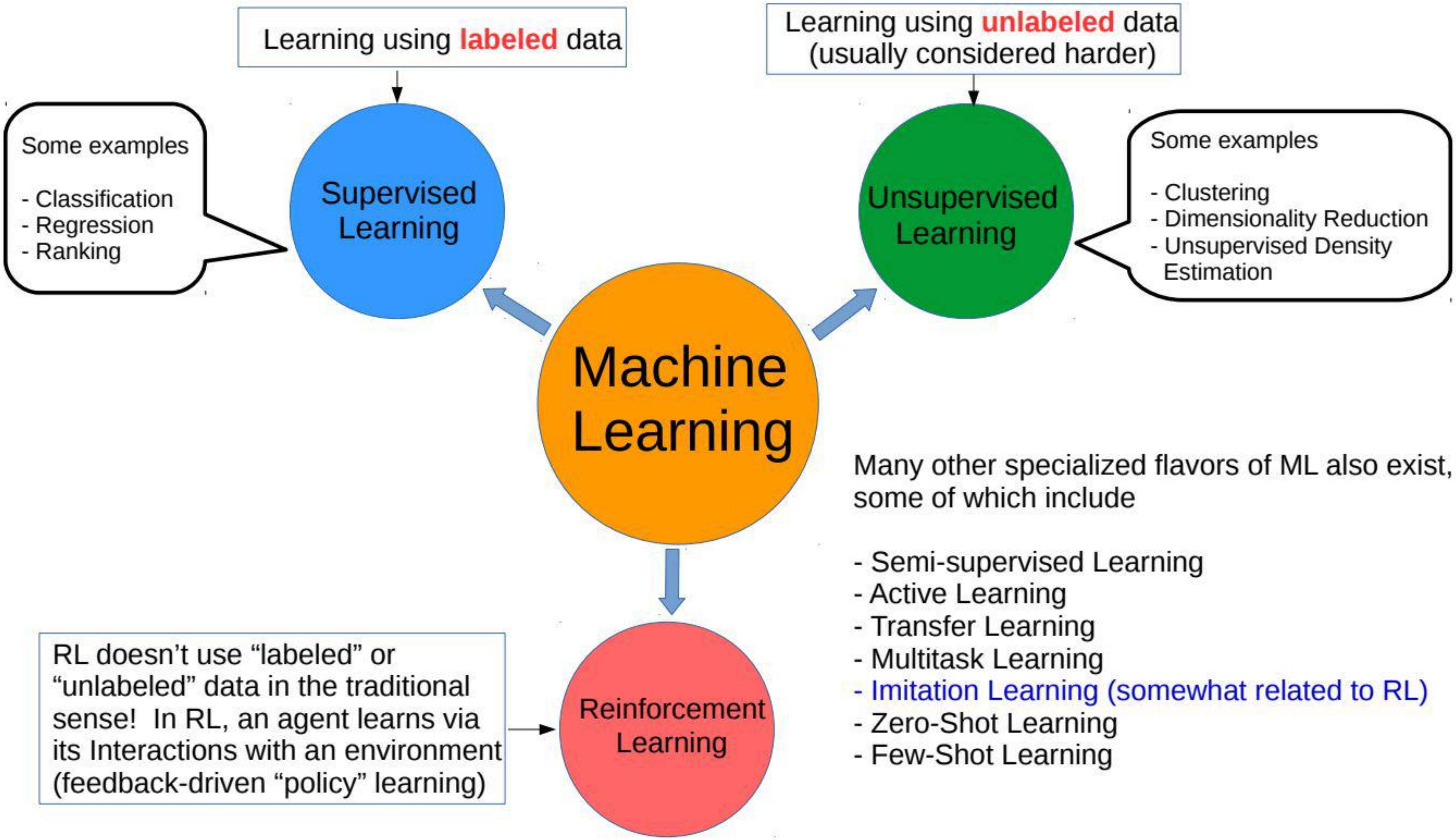
Machine Learning helps Image Recognition & Finance, etc.

Machine Learning helps Recommendation systems

- A recommendation system is a machine-learning system that is based on data that indicate links between a set of a users (e.g., people) and a set of items (e.g., products).
- A link between a user and a product means that the user has indicated an interest in the product in some fashion (perhaps by purchasing that item in the past).
- The machine-learning problem is to suggest other items to a given user that he or she may also be interested in, based on the data across all users.



Taxonomy for Machine Learning



Classification : Definition

- Classification is a form of data analysis to extract models describing important data classes.

- Essentially, it involves dividing up objects so that each is assigned to one of a number of mutually exhaustive and exclusive categories known as classes.

  - The term "mutually exhaustive and exclusive" simply means that each object must be assigned to precisely one class.

- Classification consists of assigning a class label to a set of unclassified cases.

- **Supervised Classification :** The set of possible classes is known in advance.

- **Unsupervised Classification :** Set of possible classes is not known. After classification we can try to assign a name to that class. Unsupervised classification is called **clustering**.

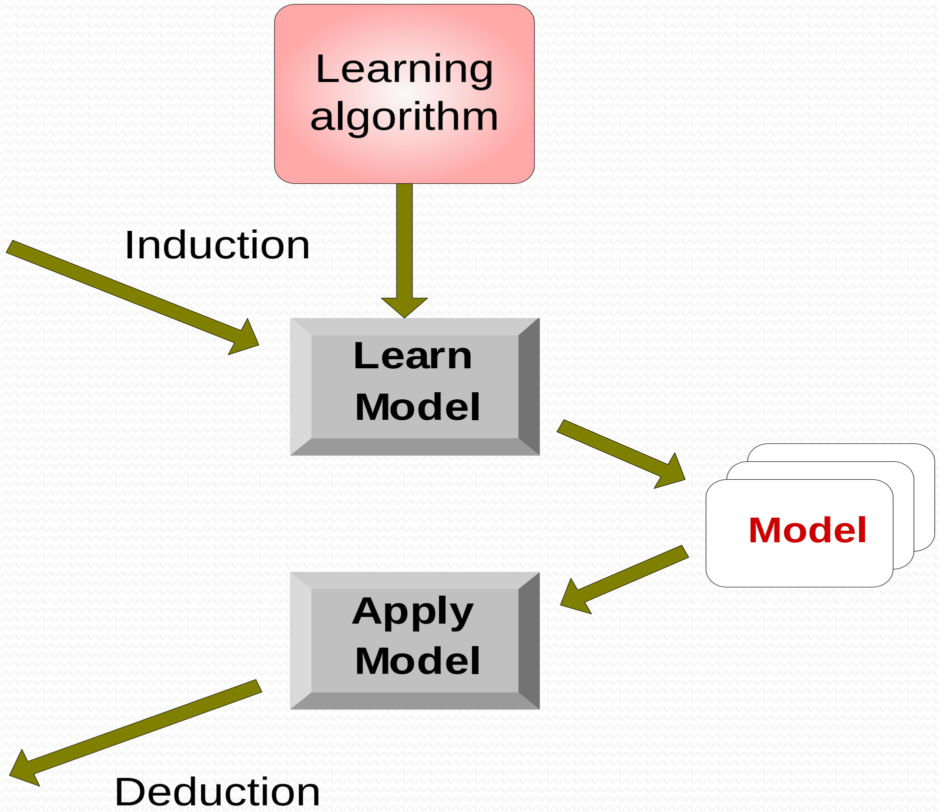
Supervised Classification Technique

- Given a collection of records (*training set*)
  - Each record contains a set of *attributes*, one of the attributes is the *class*.
- Find a *model* for class attribute as a function of the values of other attributes.
- Goal: Previously unseen records should be assigned a class as accurately as possible.
  - Satisfy the property of "mutually exclusive and exhaustive".

Illustrating Classification Tasks

Tid	Attrib1	Attrib2	Attrib3	Class
1	Yes	Large	125K	No
2	No	Medium	100K	No
3	No	Small	70K	No
4	Yes	Medium	120K	No
5	No	Large	95K	Yes
6	No	Medium	60K	No
7	Yes	Large	220K	No
8	No	Small	85K	Yes
9	No	Medium	75K	No
10	No	Small	90K	Yes

Training Set



Tid	Attrib1	Attrib2	Attrib3	Class
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?

Test Set

Classification : Example of Credit Scoring

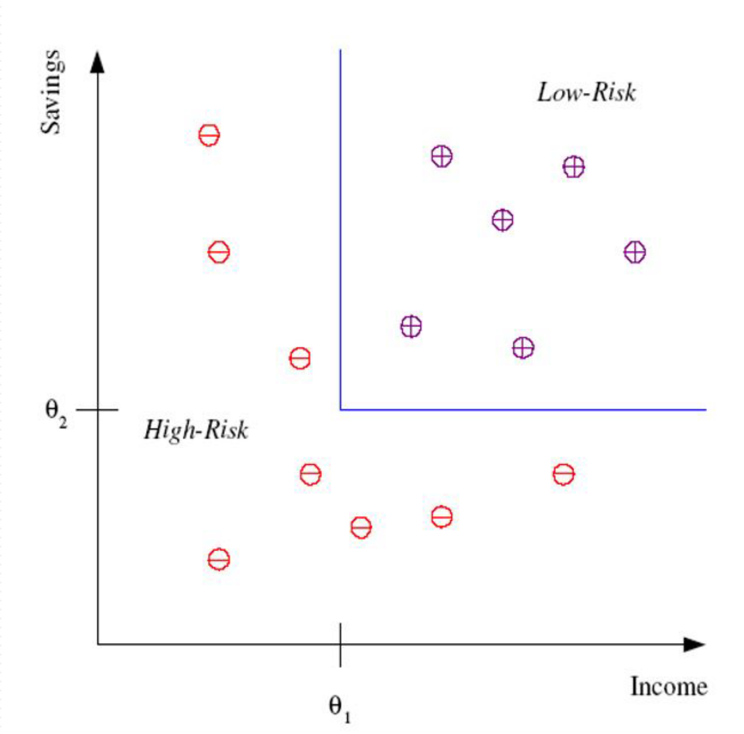
- Differentiating between low-risk and high-risk customers from their income and savings.

- Discriminant: IF Income > θ_1 AND Savings > θ_2 THEN low-risk ELSE high-risk.

- Classification: Learn a linear/nonlinear separator (the “model") using training data consisting of input-output pairs (each output is discrete-valued “label" of the corresponding input).

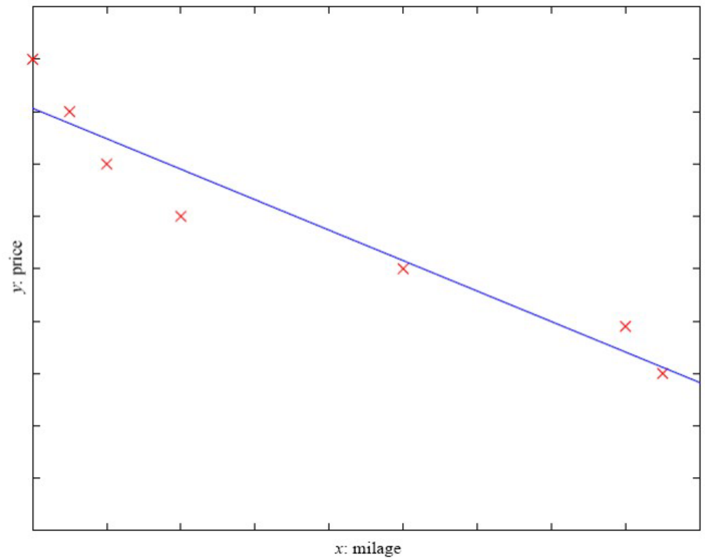
- Use it to predict the labels for new “test” inputs.

- Other Applications: Image Recognition, Spam Detection, Medical Diagnosis.



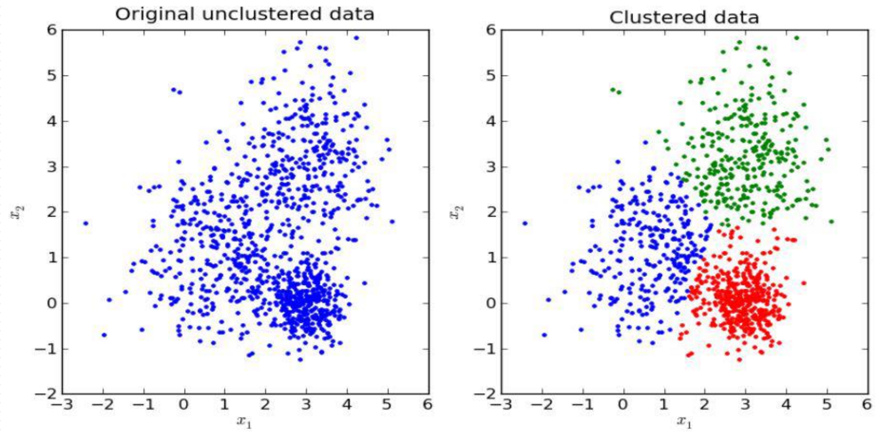
Regression : Price of a used car

- X : car attributes; Y : price and Y = f(X, θ)and savings.
- f( ) is the model and θ be the model parameters.
- Regression: Learn a line/curve (the “model") using training data consisting of input-output pairs (each output is a real-valued number).
- Use it to predict the outputs for new “test” inputs.
- Other Applications: Price Estimation, Process Improvement, Weather Forecasting.



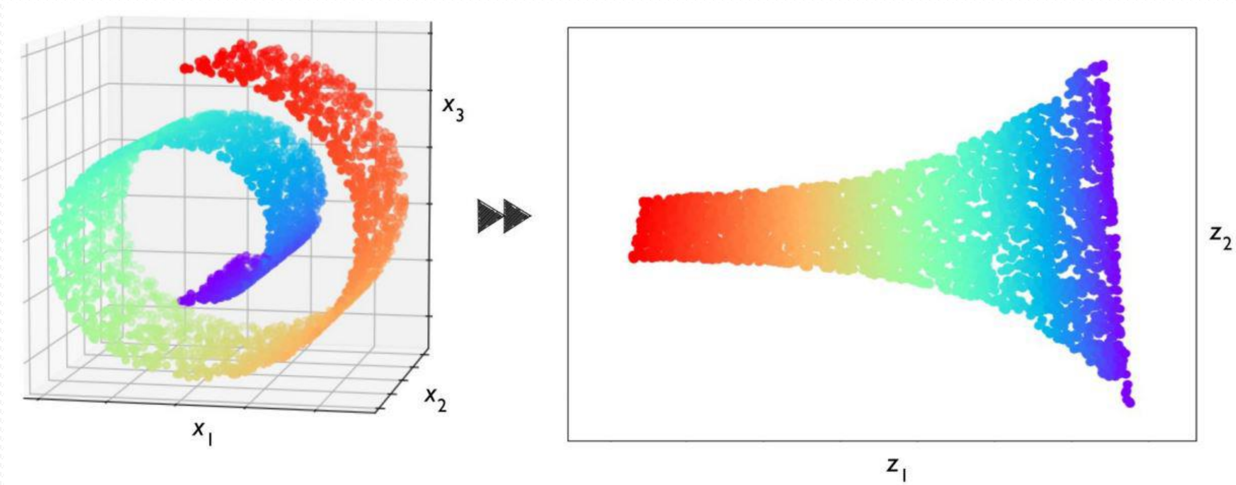
Clustering

- Given: Training data in form of unlabeled instances $\{x_1, x_2, \dots, x_n\}$.
- Goal : Learn the intrinsic latent structure that summarizes/explains data.
- Clustering: Learn the grouping structure for a given set of unlabelled inputs.
- Homogeneous groups as latent structure: Clustering.
- Applications: Topic Modelling, Image Segmentation, Social Networking.

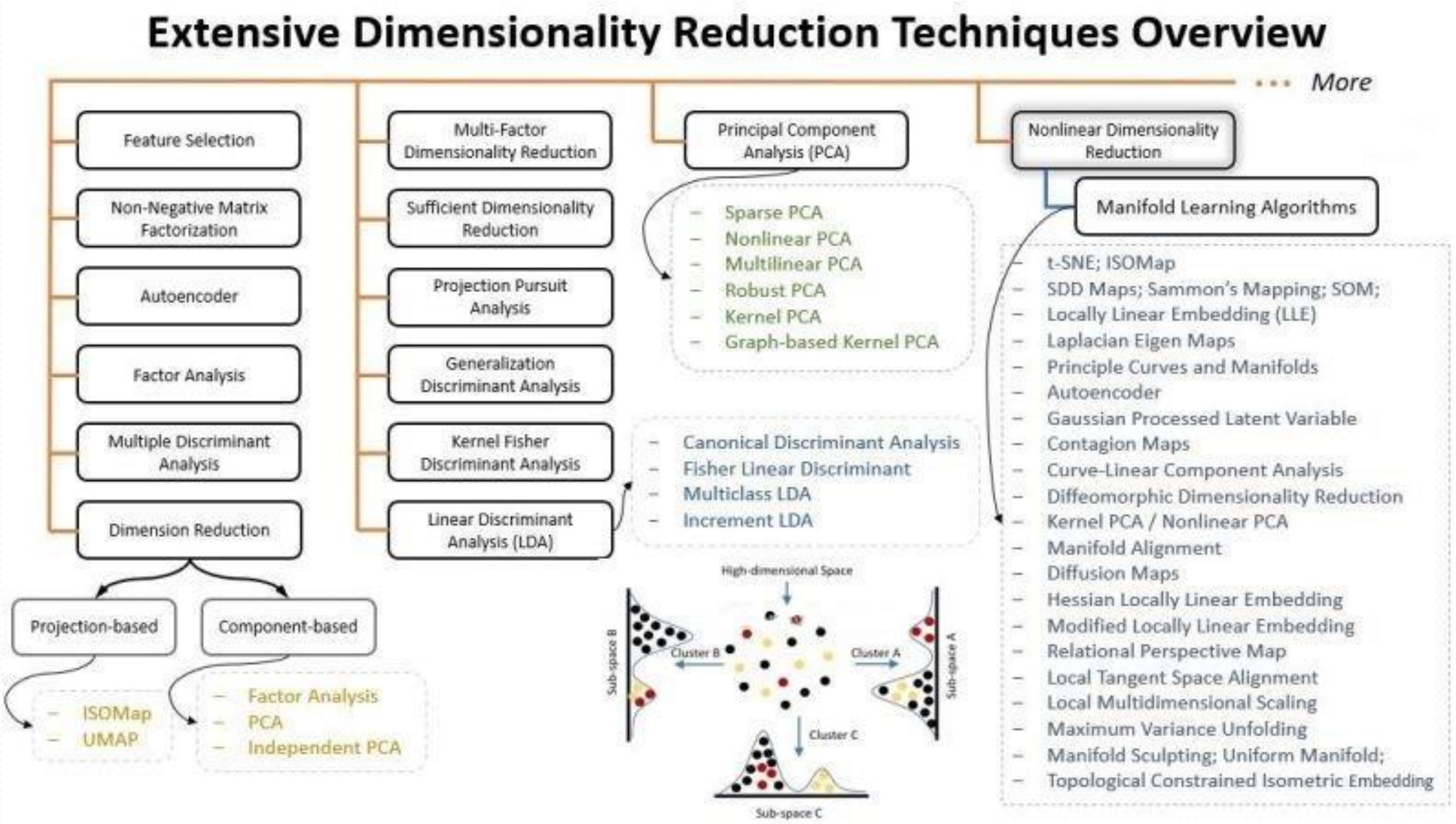


Dimensionality Reduction

- Low-dimensional latent structure: Dimensionality Reduction
- Goal : Learn a Low-dimensional representation for a given set of high-dimensional inputs.
- Figure: Three-dimension to two-dimension nonlinear projection (e.g., manifold learning).
- Applications: Customer relationship management, Text mining, Image retrieval, etc.



Dimensionality Reduction



ML Techniques

1. Traditional Statistical Methods
   - Regression Methods
   - Bayesian Classifier

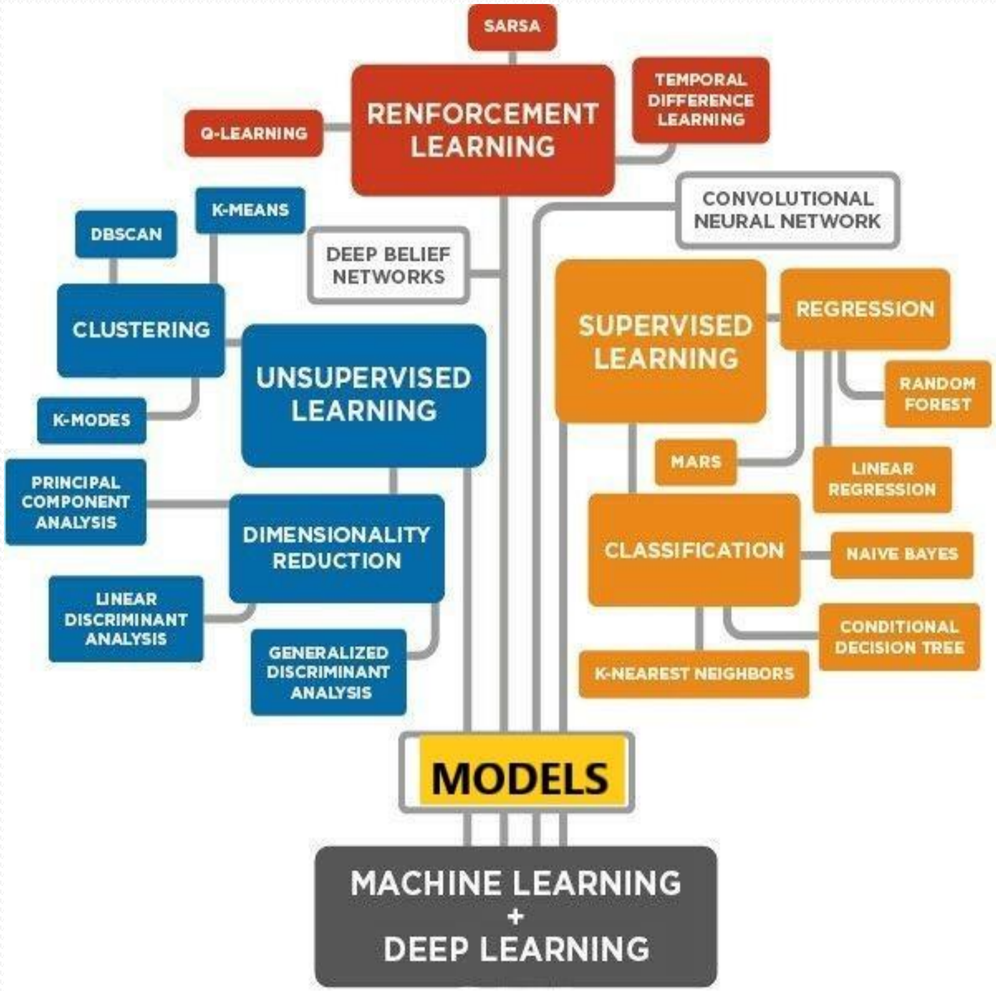
2. Distance-Based Classification
   - K-Nearest Neighbours

3. Decision Tree-Based Classification
   - ID3, C 4.5, CART, RF

5. Classification using Largest Margin (SVM)

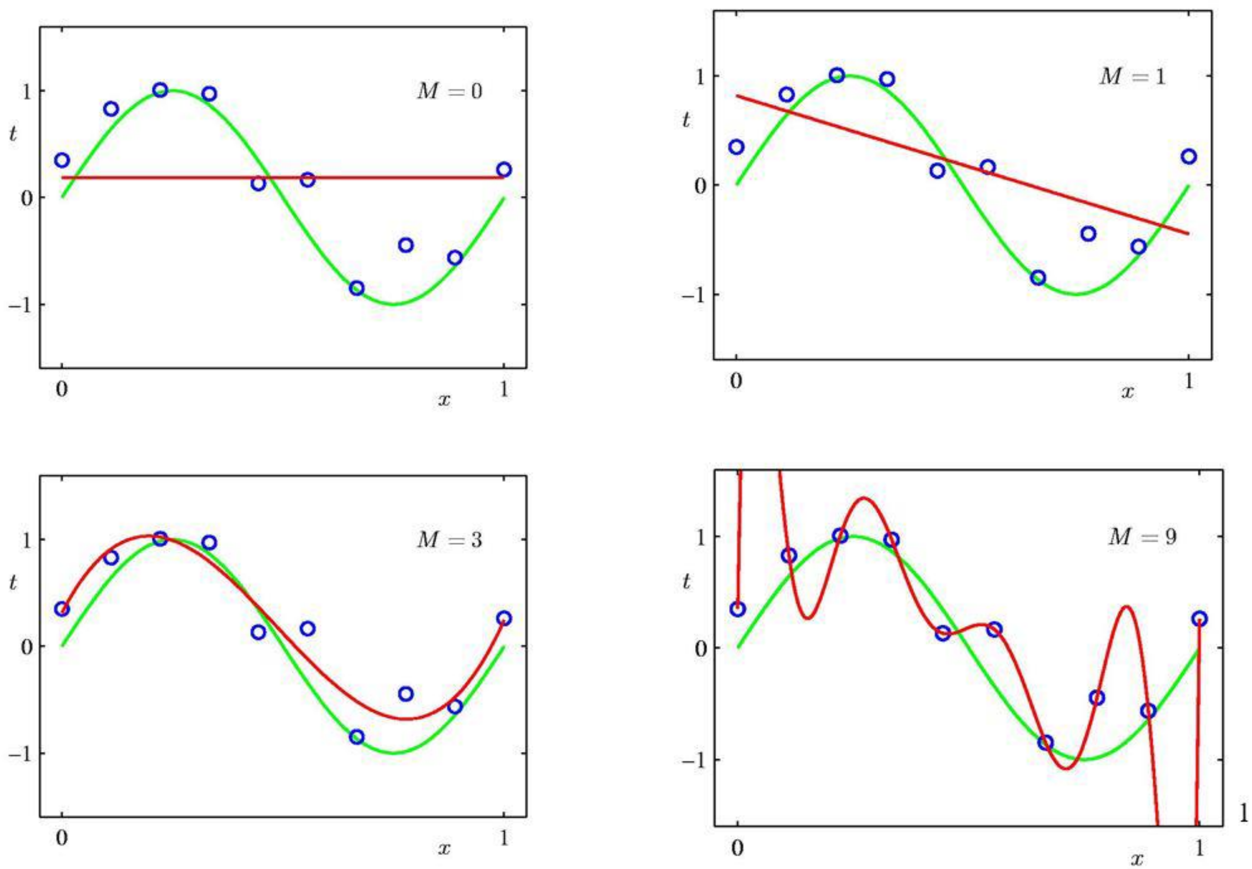
6. Classification using Neural Network (ANN)

7. .....



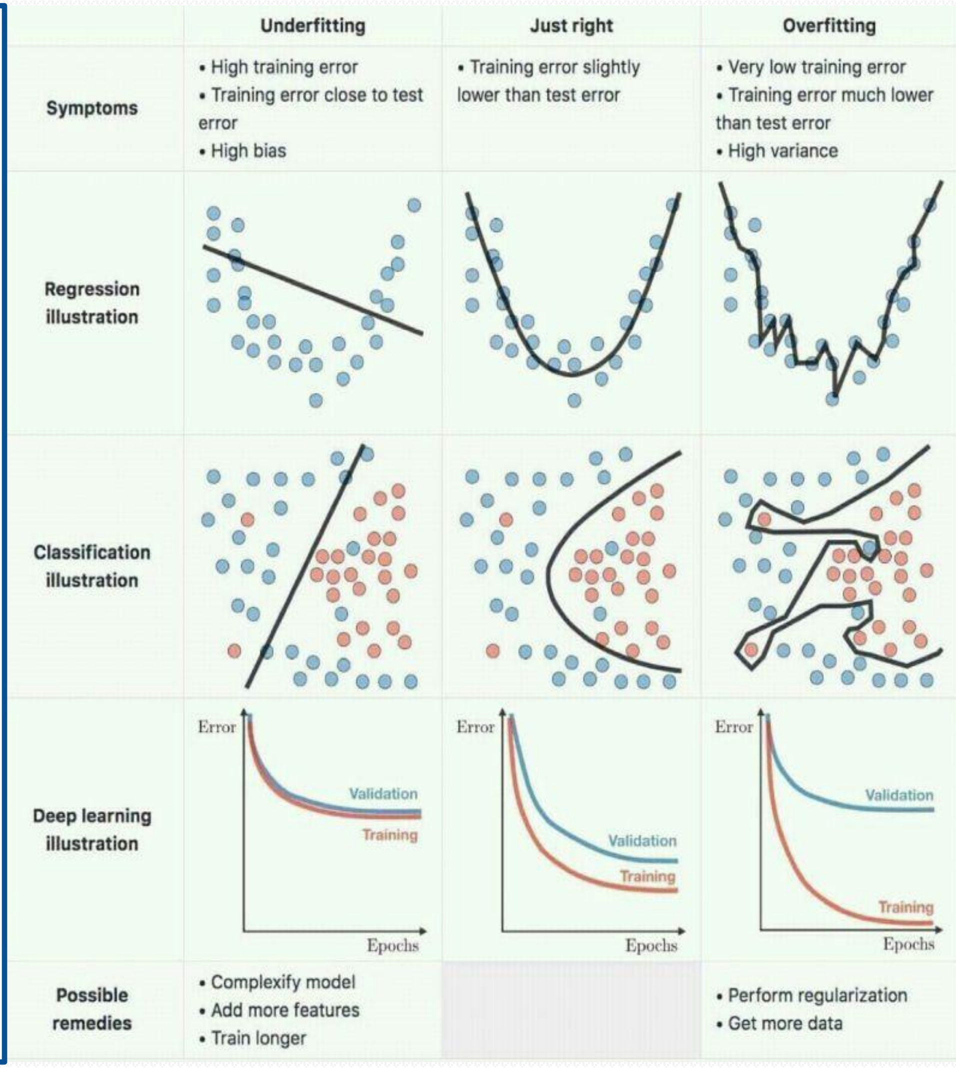
Some fits to the data: which is best?

Desired: Hypotheses that are not too simple, not too complex (so as to not over-fit on the training data).



Overfitting and Generalization

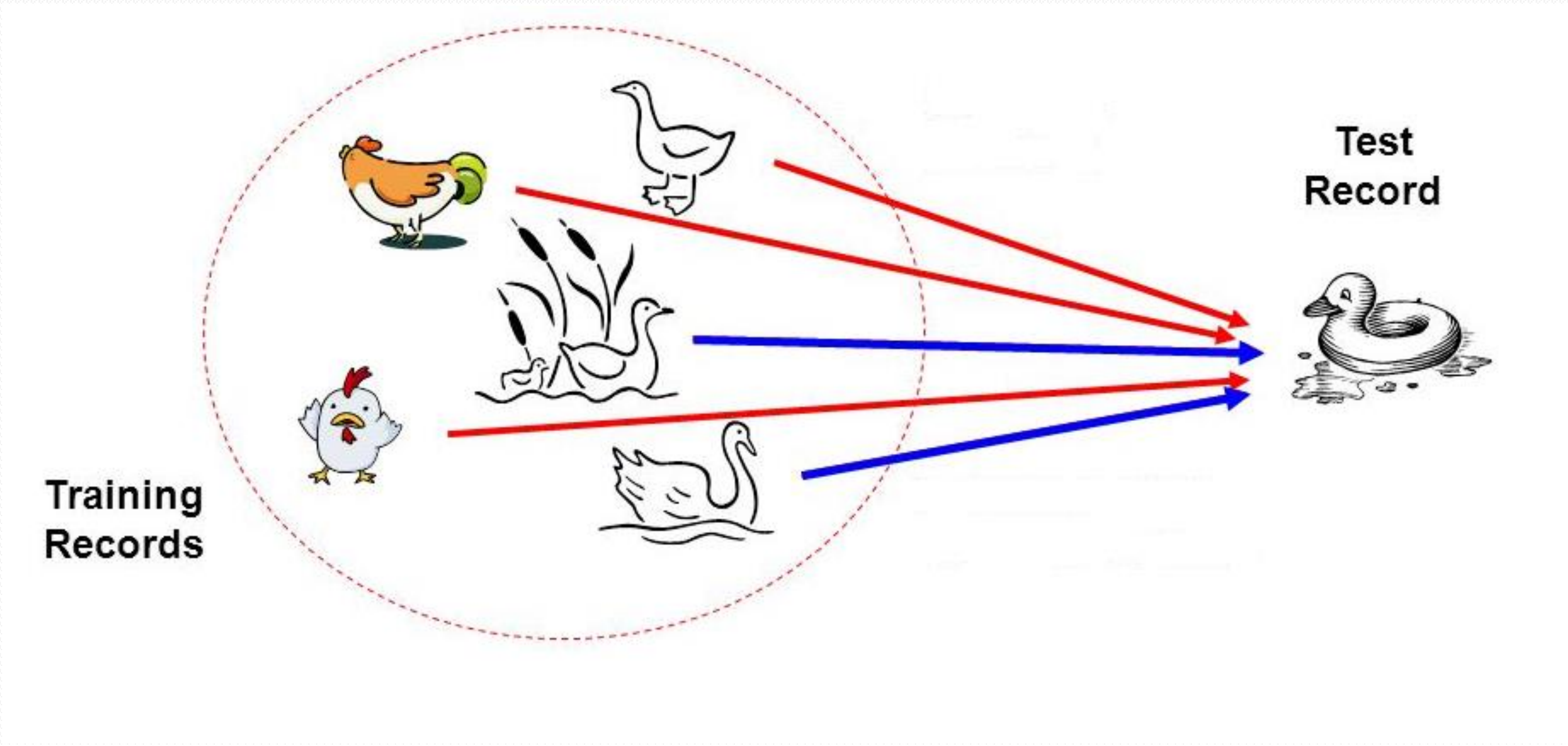
- Doing well on the training data is not enough for an ML algorithm.
- Trying to do too well (or perfectly) on training data may lead to bad "generalization".
- Generalization: Ability of an ML algorithm to do well on future "test" data.
- Simple models tend to prevent overfitting and generalize well : A key principle in designing ML algorithms (called "regularization").
- **No Free Lunch Theorem (Explore!)**



Bayesian Classifier

Bayesian Classifier

- **Basic idea :**
  - If it walks like a duck, quacks like a duck, then it is probably a duck

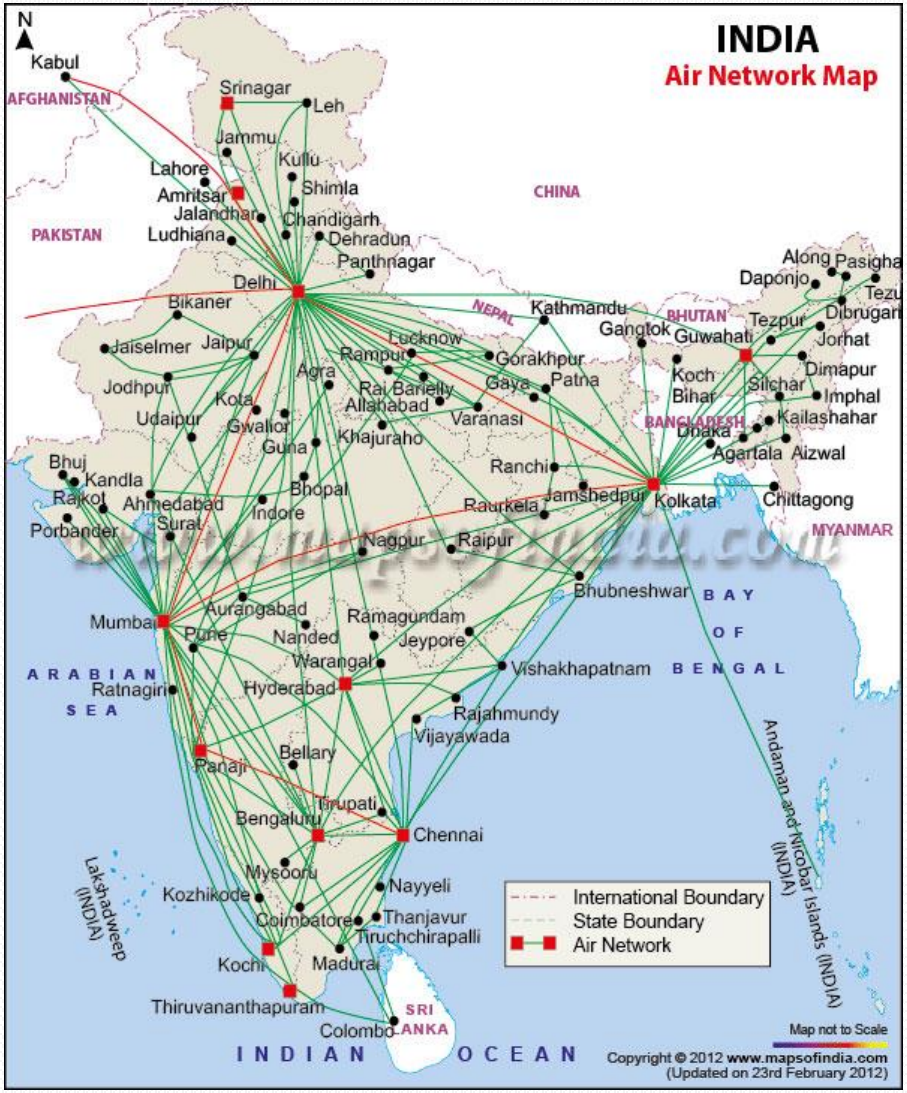


Bayesian Classifier

- A statistical classifier
  - Performs *probabilistic prediction*, *i.e.*, predicts class membership probabilities
- Foundation
  - Based on Bayes' Theorem.
- Assumptions
  1. The classes are mutually exclusive and exhaustive.
  2. The attributes are independent given the class.
- Called "Naïve" classifier because of these assumptions.
  - Empirically proven to be useful.
  - Scales very well.

Example: Bayesian Classification

- **Example :** Air Traffic Data
  - Let us consider a set observation recorded in a database
    - Regarding the arrival of airplanes in the routes from any airport to New Delhi under certain conditions.



Air-Traffic Data

Days	Season	Fog	Rain	Class
Weekday	Spring	None	None	On Time
Weekday	Winter	None	Slight	On Time
Weekday	Winter	None	None	On Time
Holiday	Winter	High	Slight	Late
Saturday	Summer	Normal	None	On Time
Weekday	Autumn	Normal	None	Very Late
Holiday	Summer	High	Slight	On Time
Sunday	Summer	Normal	None	On Time
Weekday	Winter	High	Heavy	Very Late
Weekday	Summer	None	Slight	On Time

Cond. to next slide...

Air-Traffic Data

Cond. from previous slide...

Days	Season	Fog	Rain	Class
Saturday	Spring	High	Heavy	Cancelled
Weekday	Summer	High	Slight	On Time
Weekday	Winter	Normal	None	Late
Weekday	Summer	High	None	On Time
Weekday	Winter	Normal	Heavy	Very Late
Saturday	Autumn	High	Slight	On Time
Weekday	Autumn	None	Heavy	On Time
Holiday	Spring	Normal	Slight	On Time
Weekday	Spring	Normal	None	On Time
Weekday	Spring	Normal	Heavy	On Time

Air-Traffic Data

- In this database, there are four attributes

  A = [Day, Season, Fog, Rain]

  with 20 tuples.
- The categories of classes are:

  C= [On Time, Late, Very Late, Cancelled]

- Given this is the knowledge of data and classes, we are to find most likely classification for any other unseen instance, for example:

Week Day	**Winter**	**High**	**Heavy**	**???**

- Classification technique eventually to map this tuple into an accurate class.

Bayesian Classifier

- In many applications, the relationship between the attributes set and the class variable is non-deterministic.
  - In other words, a test cannot be classified to a class label with certainty.
  - In such a situation, the classification can be achieved probabilistically.
- The Bayesian classifier is an approach for modelling probabilistic relationships between the attribute set and the class variable.
- More precisely, Bayesian classifier use Bayes' Theorem of Probability for classification.
- Before going to discuss the Bayesian classifier, we should have a quick look at the Theory of Probability and then Bayes' Theorem.

Bayes' Theorem of Probability

Simple Probability

Definition : Classical Probability

If there are n elementary events associated with a random experiment and m of n of them are favorable to an event A, then the probability of happening or occurrence of A is

$$P(A) = \frac{m}{n}$$

Suppose, A and B are any two events and $P(A)$, $P(B)$ denote the probabilities that the events A and B will occur, respectively.

Classical Probability

- **Mutually Exclusive Events:**
  - Two events are mutually exclusive, if the occurrence of one precludes the occurrence of the other.

  Example: Tossing a coin (two events)

  Tossing a Ludo cube (Six events)

Can you give an example, so that two events are not mutually exclusive?

Hint: Tossing two identical coins.

- **Independent events:** Two events are independent if occurrences of one does not alter the occurrence of other.

Example: Tossing both coin and Ludo cube together.

(How many events are here?)

Can you give an example, where an event is dependent on one or more other events(s)?

Hint: Rain and dating.

Joint Probability

Definition : **Joint Probability**

If *P(A)* and *P(B)* are the probability of two events, then

$$P(A \cup B) = P(A) + P(B) - P(A \cap B)$$

If A and B are mutually exclusive, then $P(A \cap B) = 0$
If A and B are independent events, then $P(A \cap B) = P(A).P(B)$

Thus, for mutually exclusive events

$$P(A \cup B) = P(A) + P(B)$$

Conditional Probability

Definition : **Conditional Probability**

If events are dependent, then their probability is expressed by conditional probability. The probability that A occurs given that B is denoted by $P(A|B)$.

Suppose, A and B are two events associated with a random experiment. The probability of A under the condition that B has already occurred and $P(B) \neq 0$ is given by

$$P(A|B) = \frac{\text{Number of events in } B \text{ which are favourable to } A}{\text{Number of events in } B}$$

$$= \frac{\text{Number of events favourable to } A \cap B}{\text{Number of events favourable to } B}$$

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

Conditional Probability

Corollary : **Conditional Probability**

$$P(A \cap B) = P(A).P(B|A), \quad if\ P(A) \neq 0$$

or $$P(A \cap B) = P(B).P(A|B), \quad if\ P(B) \neq 0$$

For three events A, B and C

$$P(A \cap B \cap C) = P(A).P(B).P(C|A \cap B)$$

For n events $A_1, A_2, \ldots, A_n$ and if all events are mutually independent to each other

$$P(A_1 \cap A_2 \cap \ldots \ldots \ldots \ldots \cap A_n) = P(A_1).P(A_2) \ldots \ldots \ldots \ldots P(A_n)$$

Note:

$P(A|B) = 0$ if events are **mutually exclusive**

$P(A|B) = P(A)$ if A and B are **independent**

$P(A|B) \cdot P(B) = P(B|A) \cdot P(A)$ otherwise, $\mathrm{P(A \cap B) = P(B \cap A)}$

Conditional Probability

- Generalization of Conditional Probability:

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

$$= \frac{P(B|A) \cdot P(A)}{P(B)} \qquad \because P(A \cap B) = P(B|A) \cdot P(A) = P(A|B) \cdot P(B)$$

- By the law of total probability : $P(B) = P[(B \cap A) \cup (B \cap \bar{A})]$, where $\bar{A}$ denotes the compliment of event A. Thus,

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P[(B \cap A) \cup (B \cap \bar{A})]}$$

$$= \frac{P(B|A) \cdot P(A)}{P(B|A) \cdot P(A) + P(B|\bar{A}) \cdot P(\bar{A})}$$

- In general,

$$P(A|D) = \frac{P(A) \cdot P(D|A)}{P(A) \cdot P(D|A) + P(B) \cdot P(D|B) + P(C) \cdot P(D|C)}$$

Total Probability

Definition : **Total Probability Theorem**

Let $E_1, E_2, \dots \dots E_n$ be n mutually exclusive and exhaustive events associated with a random experiment. If A is any event which occurs with E_1 or E_2 or $\dots \dots E_n$, then

$$P(A) = P(E_1).P(A|E_1) + P(E_2).P(A|E_2) + \cdots \dots \dots \dots . + P(E_n).P(A|E_n)$$

Total Probability: An Example

Example 1

A bag contains 4 red and 3 black balls. A second bag contains 2 red and 4 black balls. One bag is selected at random. From the selected bag, one ball is drawn. What is the probability that the ball drawn is red?

This problem can be answered using the concept of Total Probability

E_1 =Selecting bag I

E_2 =Selecting bag II

A = Drawing the red ball

Thus, $P(A) = P(E_1).P(A|E_1) + P(E_2).P(A|E_2)$

where, $P(A|E_1)$ = Probability of drawing red ball when first bag has been chosen

and $P(A|E_2)$ = Probability of drawing red ball when second bag has been chosen

Reverse Probability

Example 2

A bag (Bag I) contains 4 red and 3 black balls. A second bag (Bag II) contains 2 red and 4 black balls. You have chosen one ball at random. It is found as red ball. What is the probability that the ball is chosen from Bag I?

Here,

E_1 = Selecting bag I

E_2 = Selecting bag II

A = Drawing the red ball

We are to determine P(E_1|A). Such a problem can be solved using Bayes' theorem of probability.

Bayes' Theorem

Theorem : **Bayes' Theorem**

Let $E_1, E_2, \ldots \ldots E_n$ be n mutually exclusive and exhaustive events associated with a random experiment. If A is any event which occurs with E_1 or E_2 or $\ldots \ldots E_n$, then

$$P(E_i|A) = \frac{P(E_i).P(A|E_i)}{\sum_{i=1}^{n} P(E_i).P(A|E_i)}$$

Prior and Posterior Probabilities

- P(A) and P(B) are called prior probabilities
- P(A|B) and P(B|A) are called posterior probabilities

Example : Prior versus Posterior Probabilities

- This table shows that the event *Y* has two outcomes namely *A* and *B*, which is dependent on another event *X* with various outcomes like x_1, x_2 and x_3.
- **Case1:** Suppose, we don't have any information of the event *A*. Then, from the given sample space, we can calculate $P(Y = A) = \frac{5}{10} = 0.5$.
- **Case2:** Now, suppose, we want to calculate

  $P(X = x_2|Y = A) = \frac{2}{5} = 0.4$.

The later is the conditional or posterior probability, where as the former is the prior probability.

X	*Y*
x_1	*A*
x_2	*A*
x_3	*B*
x_3	*A*
x_2	*B*
x_1	*A*
x_1	*B*
x_3	*B*
x_2	*B*
x_2	*A*

Naïve Bayesian Classifier

Naïve Bayesian Classifier

- Suppose, Y is a class variable and $X = \{X_1, X_2, \ldots\ldots, X_n\}$ is a set of attributes, with instance of Y.

INPUT (X)	CLASS(Y)
… … …	
… … …	…
$x_1, x_2, \ldots, x_n$	y_i
… … …	…

- The classification problem, then can be expressed as the class-conditional probability

$$P\left(Y = y_i \mid (X_1 = x_1) \text{ AND } (X_2 = x_2) \text{ AND } \ldots\ldots (X_n = x_n)\right)$$

Naïve Bayesian Classifier

- Naïve Bayesian classifier calculate this posterior probability using Bayes' theorem, which is as follows.
- From Bayes' theorem on conditional probability, we have

$$P(Y|X) = \frac{P(X|Y)\cdot P(Y)}{P(X)}$$

$$= \frac{P(X|Y)\cdot P(Y)}{P(X|Y=y_1)\cdot P(Y=y_1) + \cdots + P(X|Y=y_k)\cdot P(Y=y_k)}$$

where,

$$P(X) = \sum_{i=1}^{k} P(X|Y=y_i)\cdot P(Y=y_i)$$

Note:

- $P(X)$ is called the evidence (also the total probability) and it is a constant.
- The probability $P(Y|X)$ (also called class conditional probability) is therefore proportional to $P(X|Y)\cdot P(Y)$.
- Thus, $P(Y|X)$ can be taken as a measure of Y given that X.

$$P(Y/X) \approx P(X|Y)\cdot P(Y)$$

Naïve Bayesian Classifier

- Suppose, for a given instance of X (say x = ($X_1 = x_1$) and ..... ($X_n = x_n$)).
- There are any two class conditional probabilities namely $P(Y = y_i | X = x)$ and $P(Y = y_j | X = x)$.
- If $P(Y = y_i | X = x) > P(Y = y_j | X = x)$, then we say that y_i is more stronger than y_j for the instance $X = x$.
- The strongest y_i is the classification for the instance $X = x$.

Revisiting Air-Traffic Data

Days	Season	Fog	Rain	Class
Weekday	Spring	None	None	On Time
Weekday	Winter	None	Slight	On Time
Weekday	Winter	None	None	On Time
Holiday	Winter	High	Slight	Late
Saturday	Summer	Normal	None	On Time
Weekday	Autumn	Normal	None	Very Late
Holiday	Summer	High	Slight	On Time
Sunday	Summer	Normal	None	On Time
Weekday	Winter	High	Heavy	Very Late
Weekday	Summer	None	Slight	On Time

Cond. to next slide…

Revisiting Air-Traffic Data

Cond. from previous slide...

Days	Season	Fog	Rain	Class
Saturday	Spring	High	Heavy	Cancelled
Weekday	Summer	High	Slight	On Time
Weekday	Winter	Normal	None	Late
Weekday	Summer	High	None	On Time
Weekday	Winter	Normal	Heavy	Very Late
Saturday	Autumn	High	Slight	On Time
Weekday	Autumn	None	Heavy	On Time
Holiday	Spring	Normal	Slight	On Time
Weekday	Spring	Normal	None	On Time
Weekday	Spring	Normal	Heavy	On Time

Classification Problem : Air-Traffic Data

- In this database, there are four attributes

A = [Day, Season, Fog, Rain]

with 20 tuples.

- The categories of classes are:

C= [On Time, Late, Very Late, Cancelled]

- Given this is the knowledge of data and classes, we are to find most likely classification for any other unseen instance, for example:

Week Day	**Winter**	**High**	**Heavy**	**???**

Naïve Bayesian Classifier

- **Example:** With reference to the Air Traffic Dataset mentioned earlier, let us tabulate all the posterior and prior probabilities as shown below.

		Class			
Attribute		On Time	Late	Very Late	Cancelled
Day	Weekday	9/14 = 0.64	½ = 0.5	3/3 = 1	0/1 = 0
	Saturday	2/14 = 0.14	½ = 0.5	0/3 = 0	1/1 = 1
	Sunday	1/14 = 0.07	0/2 = 0	0/3 = 0	0/1 = 0
	Holiday	2/14 = 0.14	0/2 = 0	0/3 = 0	0/1 = 0
Season	Spring	4/14 = 0.29	0/2 = 0	0/3 = 0	0/1 = 0
	Summer	6/14 = 0.43	0/2 = 0	0/3 = 0	0/1 = 0
	Autumn	2/14 = 0.14	0/2 = 0	1/3= 0.33	0/1 = 0
	Winter	2/14 = 0.14	2/2 = 1	2/3 = 0.67	0/1 = 0

Naïve Bayesian Classifier

		Class			
Attribute		On Time	Late	Very Late	Cancelled
Fog	None	5/14 = 0.36	0/2 = 0	0/3 = 0	0/1 = 0
	High	4/14 = 0.29	1/2 = 0.5	1/3 = 0.33	1/1 = 1
	Normal	5/14 = 0.36	1/2 = 0.5	2/3 = 0.67	0/1 = 0
Rain	None	5/14 = 0.36	1/2 = 0.5	1/3 = 0.33	0/1 = 0
	Slight	8/14 = 0.57	0/2 = 0	0/3 = 0	0/1 = 0
	Heavy	1/14 = 0.07	1/2 = 0.5	2/3 = 0.67	1/1 = 1
Prior Probability		14/20 = 0.70	2/20 = 0.10	3/20 = 0.15	1/20 = 0.05

Naïve Bayesian Classifier

Instance:

Week Day	Winter	High	Heavy	???

Case1: Class = On Time : $0.70 \times 0.64 \times 0.14 \times 0.29 \times 0.07 = 0.0013$

Case2: Class = Late : $0.10 \times 0.50 \times 1.0 \times 0.50 \times 0.50 = 0.0125$

Case3: Class = Very Late : $0.15 \times 1.0 \times 0.67 \times 0.33 \times 0.67 = 0.0222$

Case4: Class = Cancelled : $0.05 \times 0.0 \times 0.0 \times 1.0 \times 1.0 = 0.0000$

Case3 is the strongest; Hence correct classification is **Very Late**

Naïve Bayesian Classifier

Algorithm: Naïve Bayesian Classification

Input: Given a set of k mutually exclusive and exhaustive classes $C = \{c_1, c_2, \ldots\ldots, c_k\}$, which have prior probabilities $P(C_1)$, $P(C_2)$,.....,$P(C_k)$.

There are n-attribute set $A = \{A_1, A_2, \ldots\ldots, A_n\}$, which for a given instance have values $A_1 = a_1$, $A_2 = a_2$,....., $A_n = a_n$

Step: For each $c_i \in$ C, calculate the class condition probabilities, i = 1,2,.....,k

$$p_i = P(C_i) \times \prod_{j=1}^{n} P(A_j = a_j | C_i)$$

$$p_x = \max\{p_1, p_2, \ldots\ldots, p_k\}$$

Output: C_x is the classification

Note: $\sum \boldsymbol{p_i} \neq \mathbf{1}$, because they are not probabilities rather proportion values (to posterior probabilities)

Naïve Bayesian Classifier

Pros and Cons

- The Naïve Bayes' approach is a very popular one, which often works well.
- However, it has a number of potential problems
  - It relies on all attributes being categorical.
  - If the data is less, then it estimates poorly.

Naïve Bayesian Classifier

Approach to overcome the limitations in Naïve Bayesian Classification

- Estimating the posterior probabilities for continuous attributes
  - In real life situation, all attributes are not necessarily be categorical, In fact, there is a mix of both categorical and continuous attributes.
  - In the following, we discuss the schemes to deal with continuous attributes in Bayesian classifier.

1. We can discretize each continuous attributes and then replace the continuous values with its corresponding discrete intervals.

2. We can assume a certain form of probability distribution for the continuous variable and estimate the parameters of the distribution using the training data. A Gaussian distribution is usually chosen to represent the posterior probabilities for continuous attributes. A general form of Gaussian distribution will look like

$$P(x: \mu, \sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} \exp\{^{-\frac{(x-\mu)^2}{2\sigma^2}}\}$$

where, μ and σ^2 denote mean and variance, respectively.

Naïve Bayesian Classifier

For each class C_i, the posterior probabilities for attribute A_j (it is the numeric attribute) can be calculated following Gaussian normal distribution as follows.

$$P(A_j = a_j | C_i) = \frac{1}{\sqrt{2\pi}\sigma_{ij}} e^{-\frac{(a_j - \mu ij)^2}{2\sigma_{ij}^2}}$$

Here, the parameter μ_{ij} can be calculated based on the sample mean of attribute value of A_j for the training records that belong to the class C_i.

Similarly, σ_{ij}^2 can be estimated from the calculation of variance of such training records.

Naïve Bayesian Classifier

M-estimate of Conditional Probability

- The M-estimation is to deal with the potential problem of Naïve Bayesian Classifier when training data size is too poor.
  - If the posterior probability for one of the attribute is zero, then the overall class-conditional probability for the class vanishes.
  - In other words, if training data do not cover many of the attribute values, then we may not be able to classify some of the test records.
- This problem can be addressed by using the M-estimate approach.

M-estimate Approach

M-estimate approach can be stated as follows

$$P(A_{j} = a_j | C_i) = \frac{n_{c_i} + mp}{n + m}$$

where, n = total number of instances from class C_i

n_{c_i} = number of training examples from class C_i that take the value $A_j = a_j$

m = it is a parameter known as the equivalent sample size, and

p = is a user specified parameter.

Note:

If n = 0, that is, if there is no training set available, then $P(a_i|C_i) = p$,
so, this is a different value, in absence of sample value.

A Practice Example

Example 3

Class:
C1:buys_computer = 'yes'
C2:buys_computer = 'no'

Data instance
X = (age <=30,
Income = medium,
Student = yes
Credit_rating = fair)

age	income	student	credit_rating	_comp
<=30	high	no	fair	no
<=30	high	no	excellent	no
31…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no

A Practice Example: Solution

- $P(C_i)$: P(buys_computer = "yes") = 9/14 = 0.643
  P(buys_computer = "no") = 5/14= 0.357

- Compute $P(X|C_i)$ for each class
  P(age = "<=30" | buys_computer = "yes") = 2/9 = 0.222
  P(age = "<= 30" | buys_computer = "no") = 3/5 = 0.6
  P(income = "medium" | buys_computer = "yes") = 4/9 = 0.444
  P(income = "medium" | buys_computer = "no") = 2/5 = 0.4
  P(student = "yes" | buys_computer = "yes) = 6/9 = 0.667
  P(student = "yes" | buys_computer = "no") = 1/5 = 0.2
  P(credit_rating = "fair" | buys_computer = "yes") = 6/9 = 0.667
  P(credit_rating = "fair" | buys_computer = "no") = 2/5 = 0.4

- **X = (age <= 30 , income = medium, student = yes, credit_rating = fair)**

$P(X|C_i)$: P(X|buys_computer = "yes") = 0.222 × 0.444 × 0.667 × 0.667 = 0.044
P(X|buys_computer = "no") = 0.6 × 0.4 × 0.2 × 0.4 = 0.019

$P(X|C_i)$*$P(C_i)$: P(X|buys_computer = "yes") * P(buys_computer = "yes") = 0.028
P(X|buys_computer = "no") * P(buys_computer = "no") = 0.007

Therefore, X belongs to class ("buys_computer = yes").

Reference

- The detail material related to this lecture can be found in

Data Mining: Concepts and Techniques, (3rd Edn.), Jiawei Han, Micheline Kamber, Morgan Kaufmann, 2015.

Introduction to Data Mining, Pang-Ning Tan, Michael Steinbach, and Vipin Kumar, Addison-Wesley, 2014

Any question?

ctanujit@gmail.com