Artificial Intelligence for Knowledge Discovery

Hatim Chahdi

Lecture 3 : Machine Learning

AxI Technologies

Course objectives

- The main objective of this course is to give you an overview of the process **of building a end-to-end Knowledge Discovery System**.
- Through the lectures, the labs and the project, you will gain an understanding and a hands-on experience in the different parts and the necessary components to set up such systems
- **Ontologies – Reasoning - Text Mining - Knowledge extraction et representation**

AxIC Technologies

Course overview

- Introduction
- Ontologies for knowledge management
- Deductive Reasoning for Ontologies
- Applied Machine Learning to Text
- Knowledge Extraction for Ontology Construction

Project : Text mining for Knowledge Extraction

Course logistics

Course planning :

- Lectures : 10,5h
- Tutorials : 14h
- Project : 10,5h

Prerequisites :

- Intermediate Java & Python
- Basic Statistics

AxIC Technologies

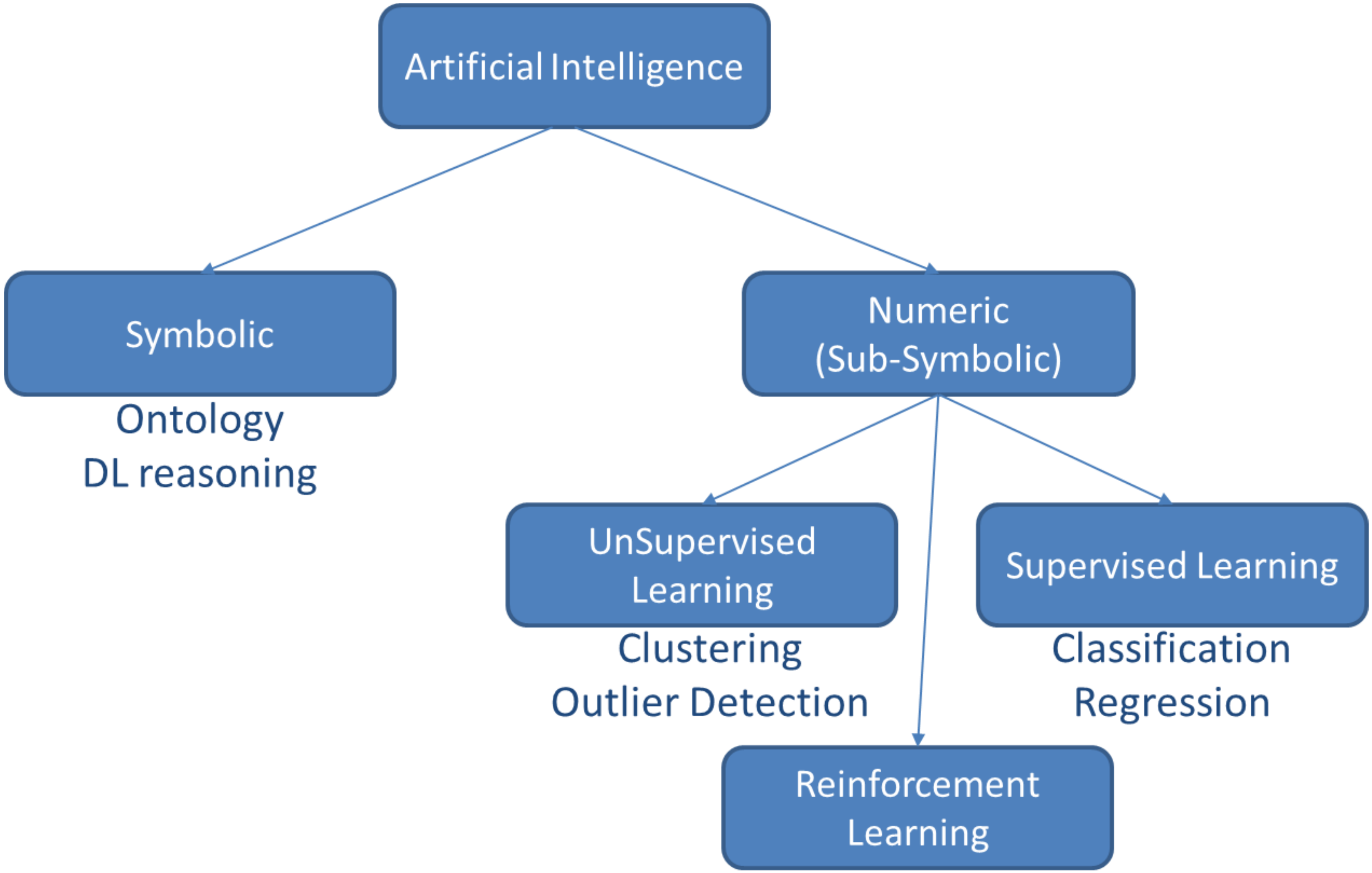
Lecture Outline:

- Overview of the knowledge extraction process
- Machine Learning techniques and concepts
- Overview of the Knowledge discovery process
- Conclusion and discussions about the project

Introduction :

- Recall of the course objective : Artificial Intelligence for Knowledge Discovery
  - Ontology => Knowledge Representation and Management
  - Machine Learning => Knowledge Extraction
  - Decision Support System => Using both to deliver insights

An Artificial Intelligence Vision



Supervised Learning :

- Classification
- Regression
- A lot of other variants :
  - Sequence generation
  - Syntax tree prediction
  - Object detection
  - Image segmentation

UnSupervised Learning :

- Clustering
- Dimensionality reduction

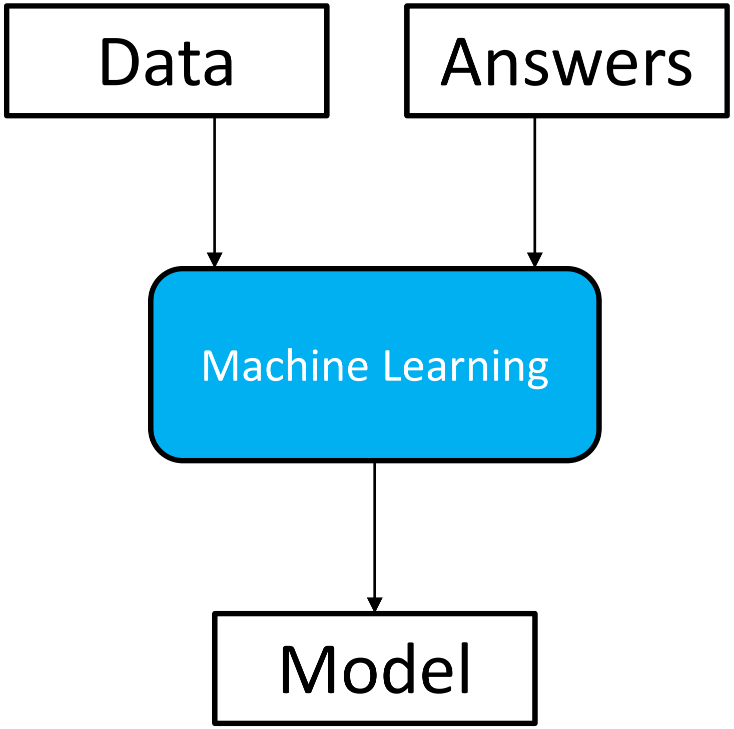
What will we see in the rest of this lecture :

- Classification algorithms

- Clustering algorithms

- The general framework of the machine learning process

Classification :



Supervised Learning : Definition

- The principle

  Given a collection of a training set containing a set of attributes (variables) and the corresponding output (class label), learn a representation (rules, model) that maps between the attributes and there labels

- Formal definition

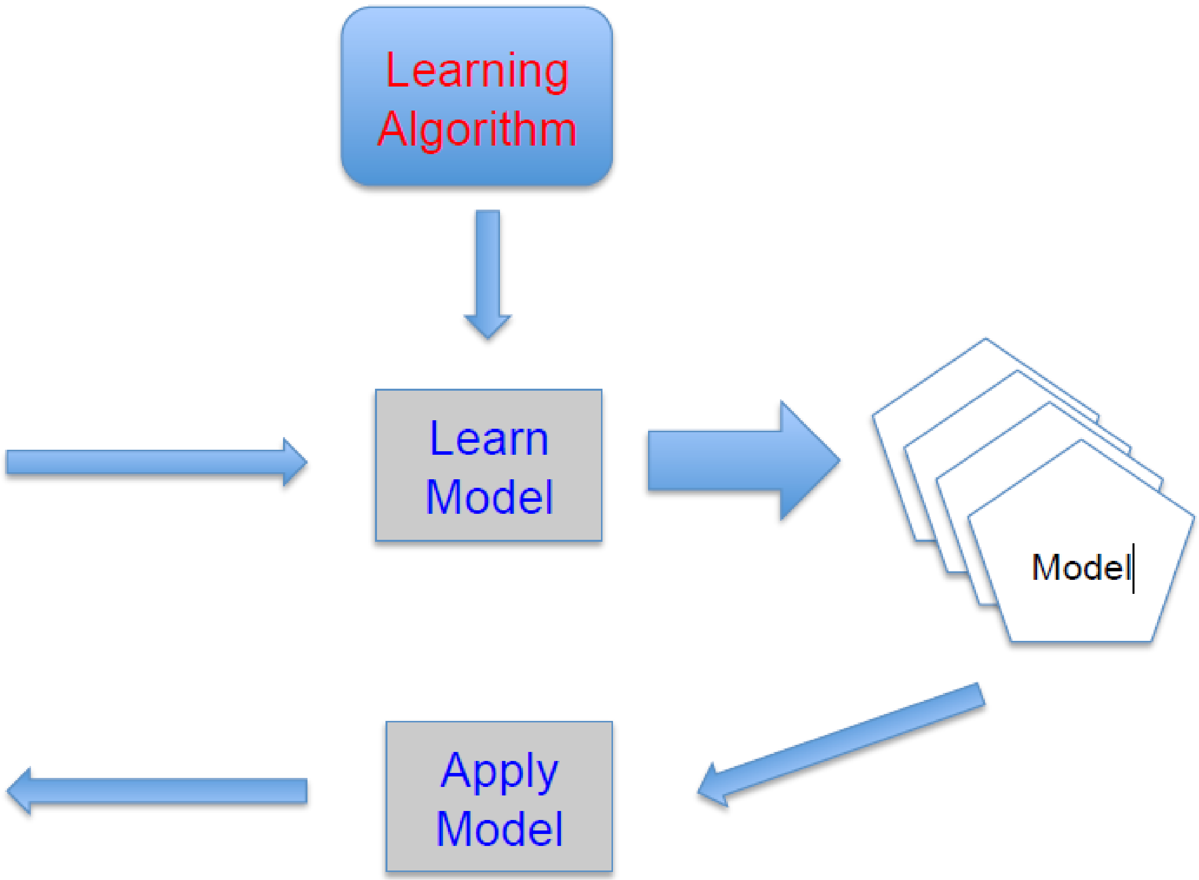
  "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E"

Tom M. Mitchell

Supervised Learning :

- Classification : The framework

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes



Tid	Refund	Marital Status	Taxable Income	Cheat
1	No	Married	80K	?
2	No	Single	100K	?
3	Yes	Single	90K	?
4	No	Married	120K	?
5	Yes	Divorced	130K	?

Supervised Learning :

- Classification examples :
  - Filtering spam from emails
  - Predicting tumor cells as benign or malignant
  - Classifying credit card transactions as legitimate or fraudulent
  - Classifying secondary structures of protein as alpha-helix, beta-sheet, or random coil
  - Categorizing news stories as finance (Topics)
  - Movies Recommendation
  - Named Entity Recognition

Supervised Learning :

- Classification techniques :
  - Decision Tree
  - Naïve Bayes
  - Instance Based Learning
  - Rule-based Methods
  - Neural Networks
  - Bayesian Belief Networks
  - Support Vector Machines

Supervised Learning :

Before we continue : Terminology (1/2)

- Sample or input : One data point that goes into the model
- Prediction or output : What comes out of your model
- Target – The truth : What your model should ideally have predicted
- Prediction error or loss value : measure of the distance between your model's prediction and the target
- Classes : A set of possible labels to choose from in a classification problem

Supervised Learning :

Before we continue : Terminology (2/2)

- Label : A specific instance of a class annotation in a classification problem
- Ground-truth or annotations : All targets for a dataset, typically collected by humans
- Binary classification : A classification task where each input sample should be categorized into two exclusive categories
- Multiclass classification : A classification task where each input sample should be categorized into more than two categories
- Multilabel classification : A classification task where each input sample can be assigned multiple labels.

Supervised Learning :

Google ML Glossary :

https://developers.google.com/machine-learning/glossary/

Decision Tree : (Principles)

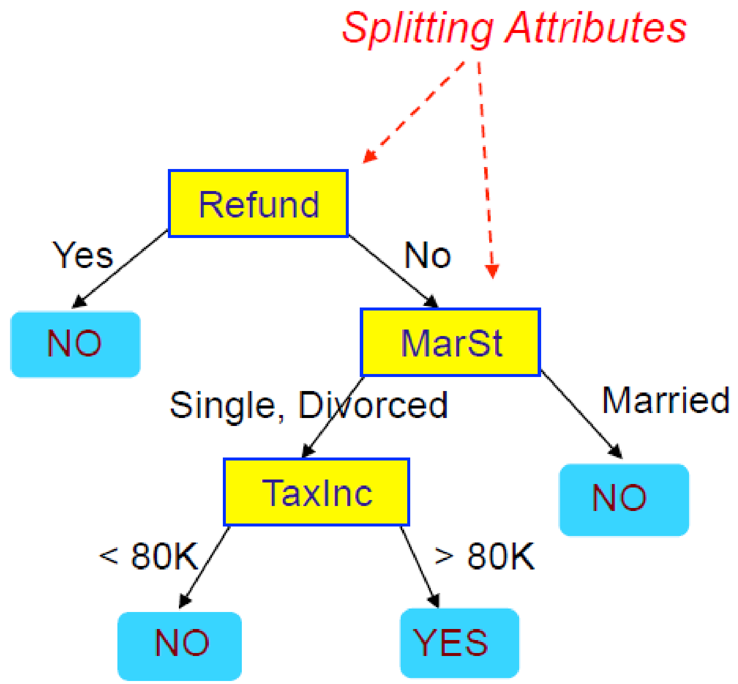
- Uses a tree structure to model the training set
- Classifies a new record following the path in the tree
- Inner nodes represent attributes and leaves nodes represent the class

Decision Tree : Example

categorical categorical continuous class

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Training Data

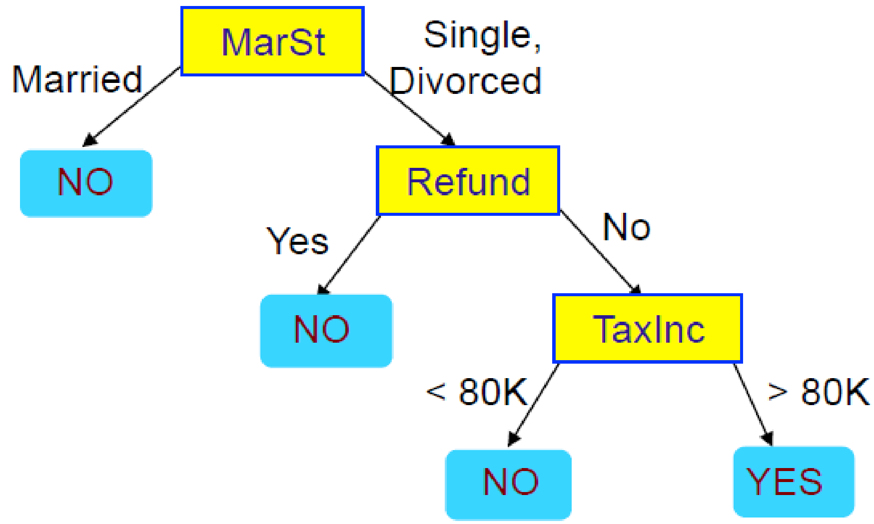


Model: Decision Tree

Decision Tree : Example of an other tree

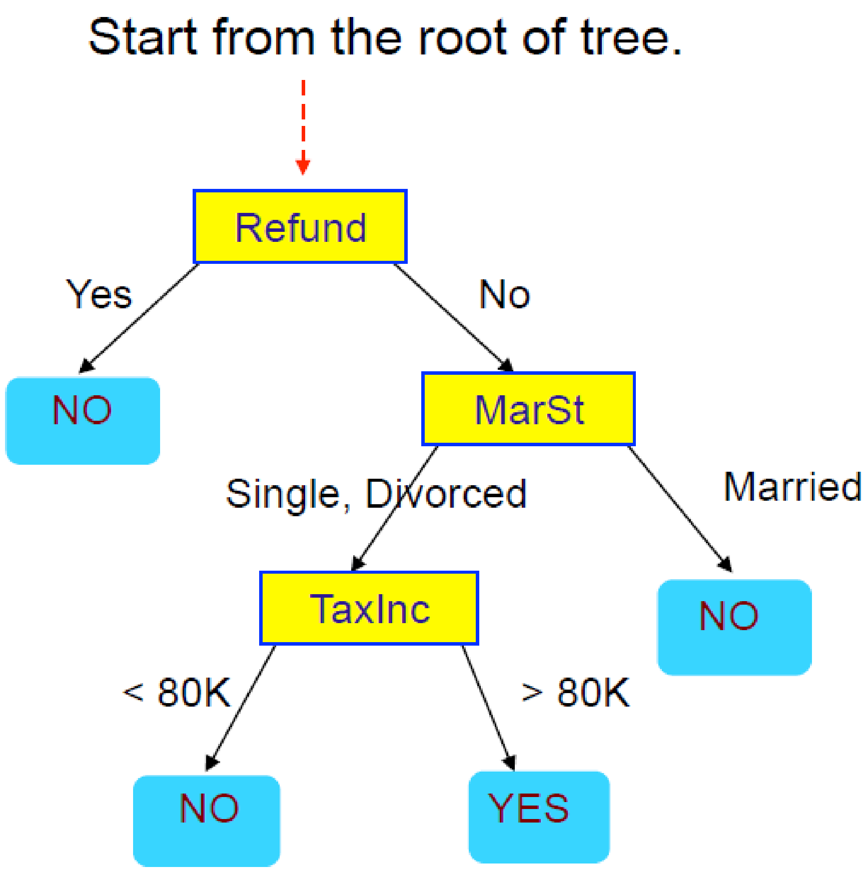
categorical categorical continuous class

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes



There could be more than one tree that fits the same data!

Decision Tree : Model application

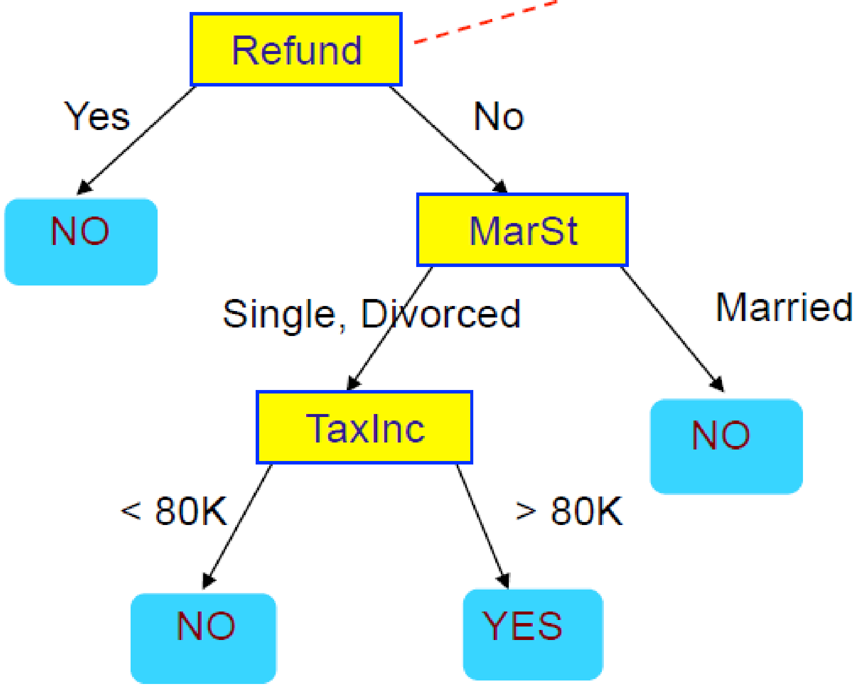


Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

Supervised Learning :

Decision Tree : Model application

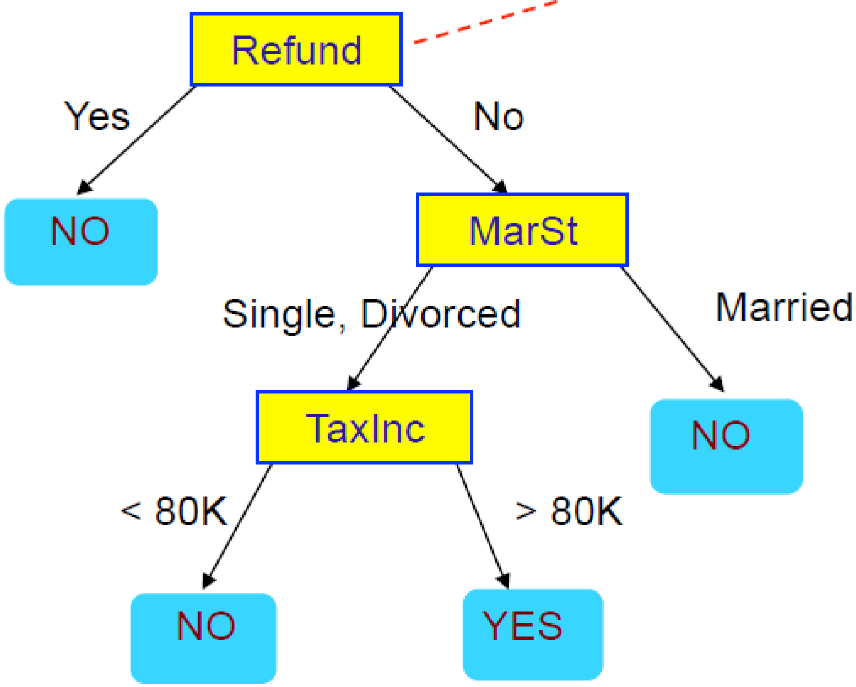


Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

Supervised Learning :

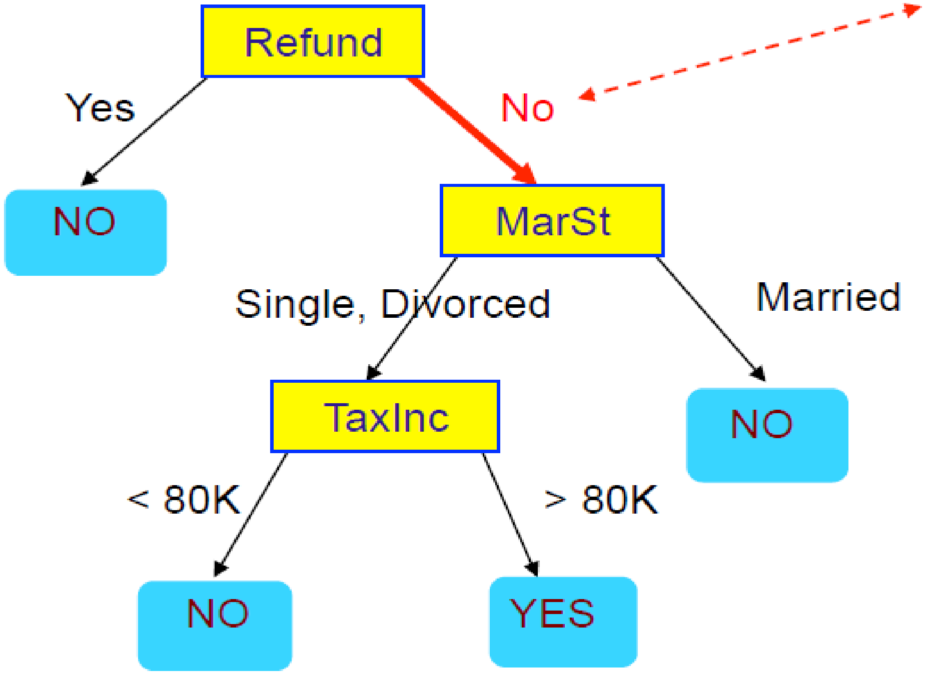
Decision Tree : Model application



Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

Decision Tree : Model application

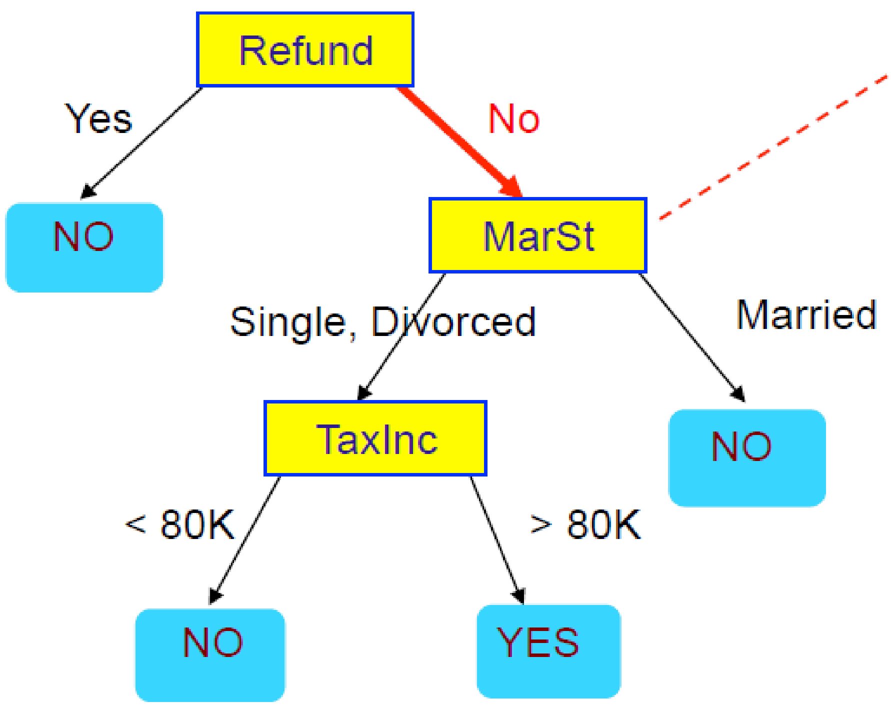


Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

Supervised Learning :

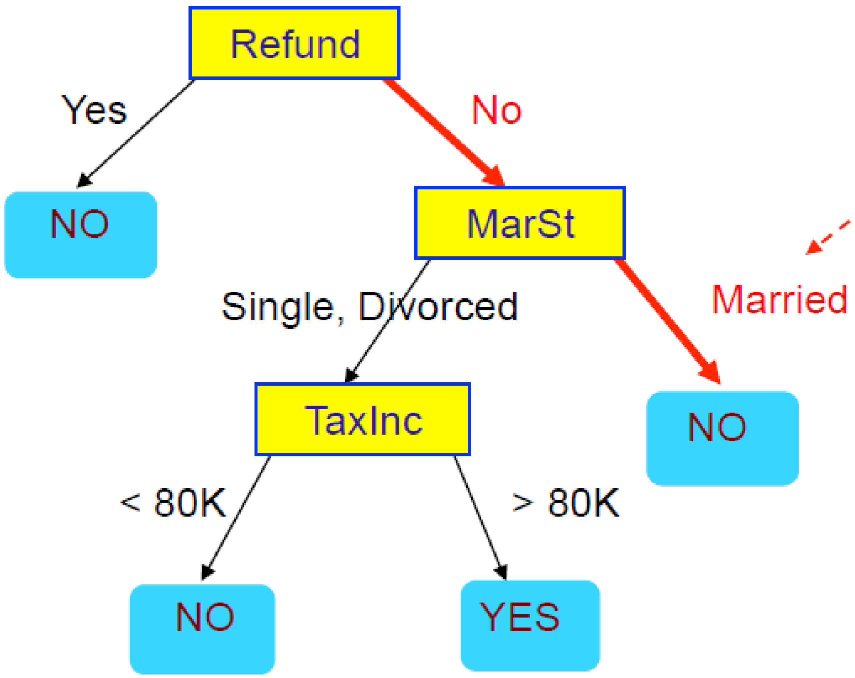
Decision Tree : Model application



Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

Decision Tree : Model application



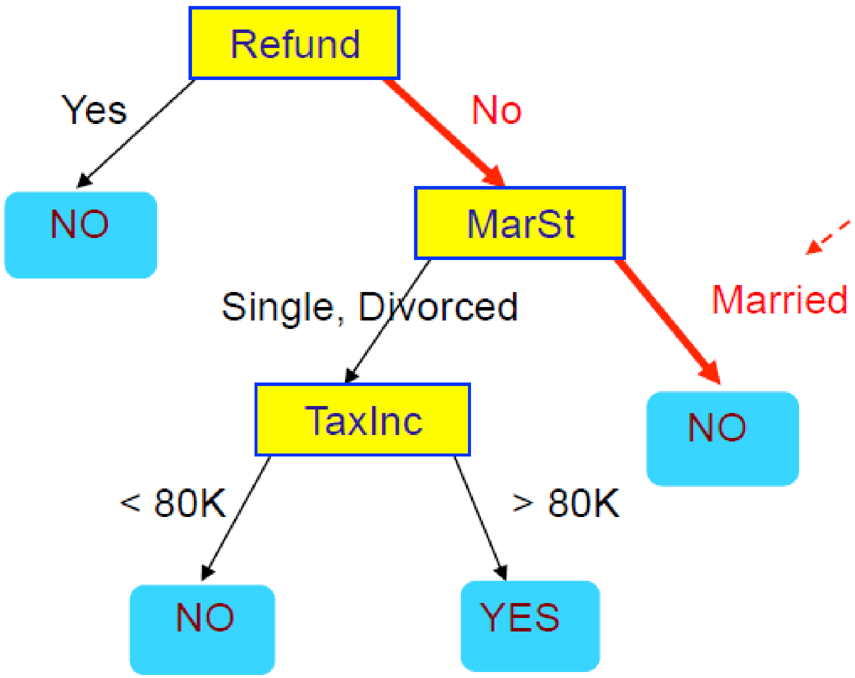
Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

Decision Tree : Model application

Test Data

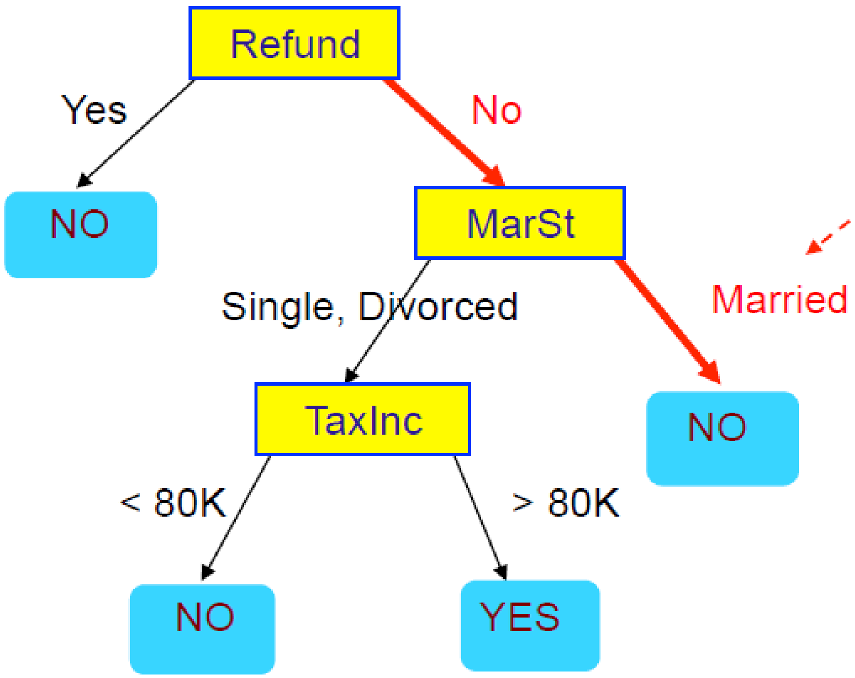
Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?



Decision Tree : Model application

Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?



Assign Cheat to “No”

Supervised Learning :

Decision Tree : Learning algorithms

- Hunt’s Algorithm (one of the earliest)
- CART
- ID3, C4.5
- SLIQ,SPRINT

Decision Tree Induction

- Greedy strategy : Split the records based on an attribute test that optimizes certain criterion
- Determine how to split the records :
  - How to specify the attribute test condition?
  - How to determine the best split?
- Determine when to stop splitting

Decision Tree Induction : How to Specify Test Condition?

- Depends on attribute types :

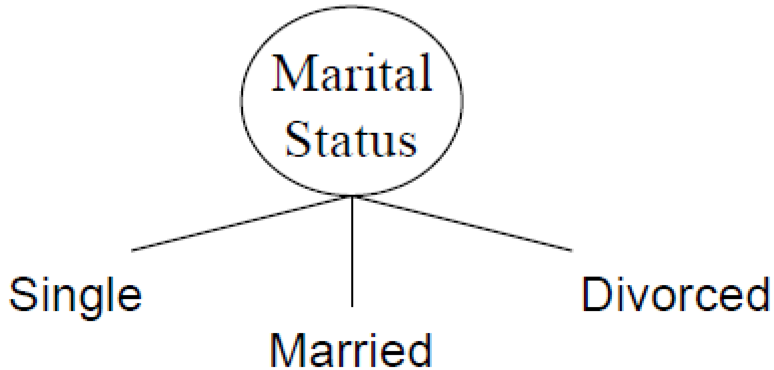
  Nominal, ordinal or continuous

- Depends on number of ways to split

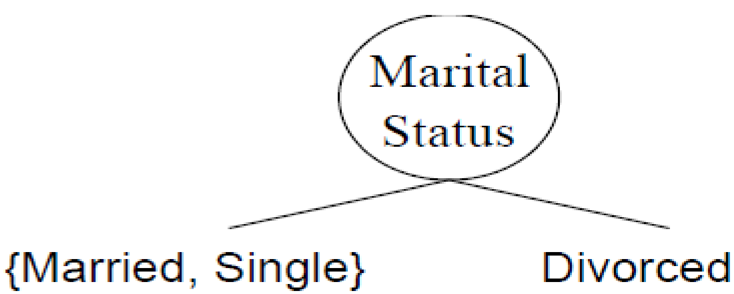
  2-way split, Multi-way split

Decision Tree Induction : Splitting Based on Nominal Attributes

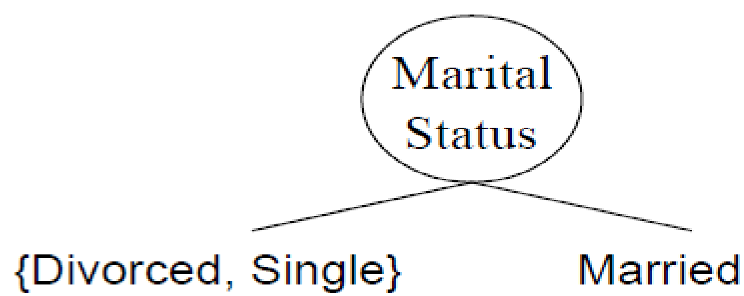
- Multi-way split: Use as many partitions as distinct values



- Binary split: Divides values into two subsets. Need to find optimal partitioning



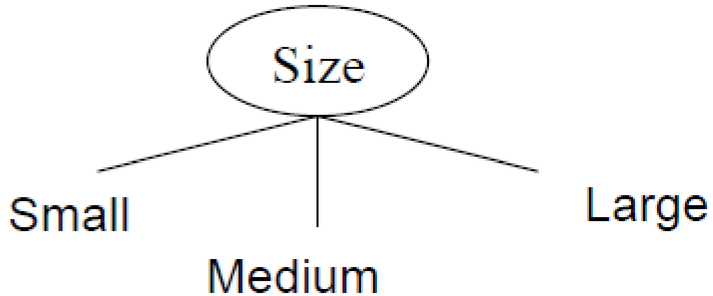
OR



Decision Tree Induction : Splitting Based on Ordinal Attributes

- We can imagine an attribute SIZE defined over the ordered set {Small, Medium, Large}

- Multi-way split: Use as many partitions as distinct values.



Decision Tree Induction : Splitting Based on Continuous Attributes

Different ways of handling

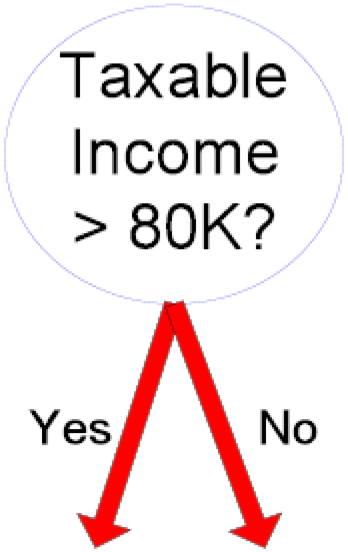
– Discretization to form an ordinal categorical attribute

- Static : discretize once at the beginning
- Dynamic : ranges can be found by equal interval bucketing, equal frequency bucketing (percentiles), or clustering

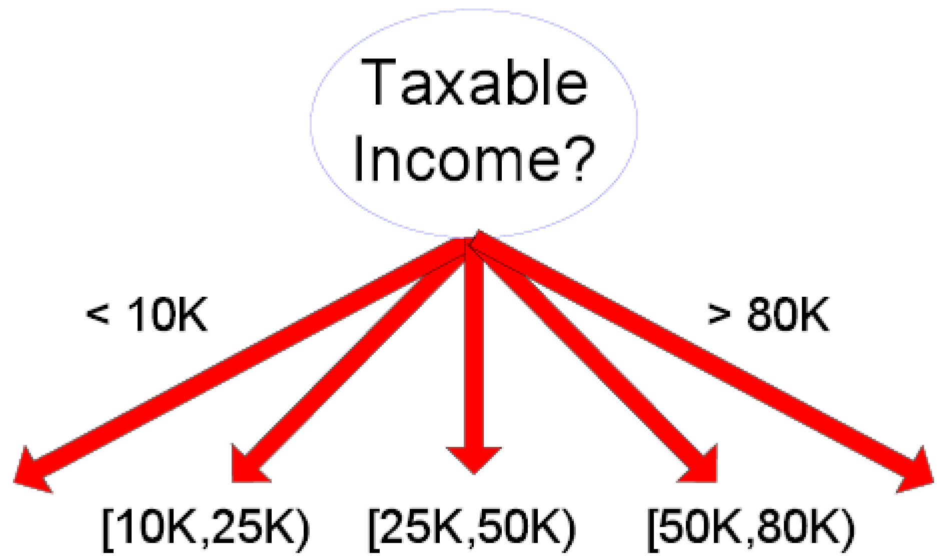
– Binary Decision: $(A < v)$ or $(A \geq v)$

- consider all possible splits and finds the best cut
- can be more compute intensive

Decision Tree Induction : Splitting Based on Continuous Attributes



(i) Binary split



(ii) Multi-way split

AxIO Technologies

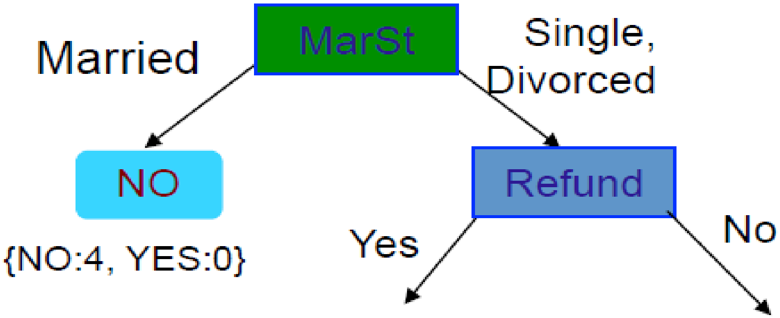
Decision Tree Induction : Best Split

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

	Yes	No
Married	0	4
Single, Divorced	3	3

Decision Tree Induction : Best Split

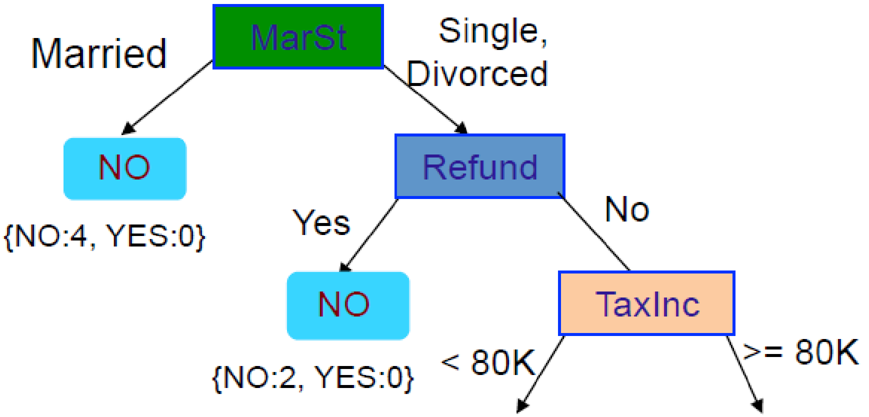
Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes



		YES	NO
Single, Divorced	**Refund = NO**	3	1
Single, Divorced	**Refund = Yes**	0	2

Decision Tree Induction : Best Split

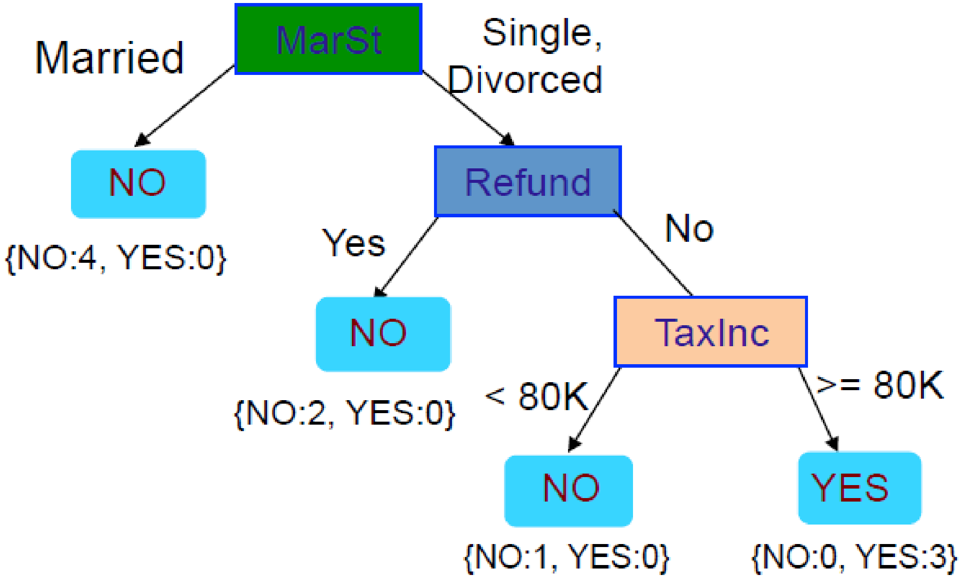
Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes



			YES	NO
Single, Divorced	**Refund = NO**	**TaxInc = < 80k**	0	1
Single, Divorced	**Refund = NO**	**TaxInc = >= 80k**	3	0

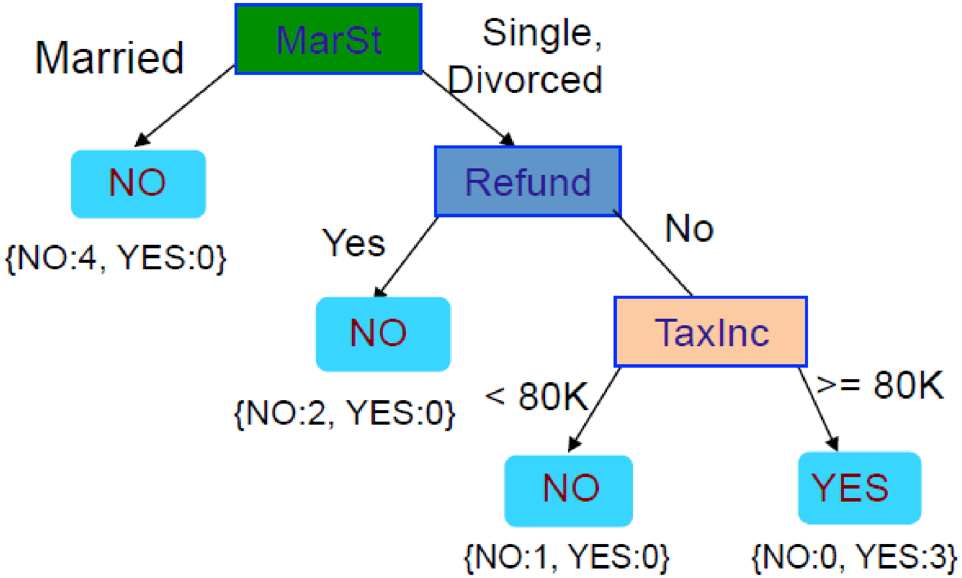
Decision Tree Induction : Best Split

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes



Decision Tree Induction : Best Split

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes



Decision Tree Induction : Stopping criteria

- Stop expanding a node when all the records belong to the same class

- Stop expanding a node when all the records have similar attribute values

Decision Tree Induction : Advantages

- Inexpensive to construct
- Extremely fast at classifying unknown records
- Easy to interpret for small-sized trees
- Accuracy is comparable to other classification
- Techniques for many simple data sets

Naïve Bayes

- Uses probability theory to model the training set
- Assumes independence between attributes
- Produces a model for each class

Supervised Learning :

Naïve Bayes : Principles

- Conditional Probability:

$$P(C \mid A) = \frac{P(A,C)}{P(A)}$$

$$P(A \mid C) = \frac{P(A,C)}{P(C)}$$

- Bayes theorem:

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

Naïve Bayes : Example

Given:

A doctor knows that meningitis causes stiff neck 50% of the time

Prior probability of any patient having meningitis is 1/50,000

Prior probability of any patient having stiff neck is 1/20

If a patient has stiff neck, what’s the probability he/she has meningitis?

$$P(M \mid S) = \frac{P(S \mid M)P(M)}{P(S)} = \frac{0.5 \times 1/50000}{1/20} = 0.0002$$

Supervised Learning :

Naïve Bayes :

Consider each attribute and class label as random variables

- Given a record with attributes (A1, A2,…,An)
  - Goal is to predict class C
  - Specifically, we want to find the value of C that maximizes P(C| A1, A2,…,An)
- Can we estimate P(C| A1, A2,…,An) directly from data?

Naïve Bayes : Approach

- Compute the posterior probability P(C | A1, A2, …, An) for all values of C using the Bayes theorem

$$P(C \mid A_1 A_2 \ldots A_n) = \frac{P(A_1 A_2 \ldots A_n \mid C) P(C)}{P(A_1 A_2 \ldots A_n)}$$

- Choose value of C that maximizes P(C | A1, A2, …, An)
- Equivalent to choosing value of C that maximizes P(A1, A2, …, An|C)P(C)
- How to estimate P(A1, A2, …, An | C)?

Naïve Bayes : Approach

- Assume independence among attributes Ai when class is given:

$$P(A1, A2, \ldots, An \mid C) = P(A1 \mid Cj)\, P(A2 \mid Cj) \ldots P(An \mid Cj)$$

Can estimate P(Ai| Cj) for all Ai and Cj

New point is classified to Cj if P(Cj) Π P(Ai| Cj) is maximal.

Naïve Bayes : Estimates Probabilities from data

Tid	Refund	Marital Status	Taxable Income	Evade
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

- Class: P(C) = Nc/N

  e.g., P(No) = 7/10, P(Yes) = 3/10

- For discrete attributes:

  P(Ai | Ck) = |Aik|/ Nc

  Where : |Aik| is number of instances having attribute Ai and belongs to class Ck

- Examples:

  P(Status=Married|No) = 4/7

  P(Refund=Yes|Yes)=0

Supervised Learning :

Naïve Bayes : Estimates Probabilities from data

For continuous attributes:

- Discretize the range into bins :
  - One ordinal attribute per bin
  - Violates independence assumption
- Two-way split: (A < v) or (A > v)
  - choose only one of the two splits as new attribute
- Probability density estimation:
  - Assume attribute follows a normal distribution
  - Use data to estimate parameters of distribution (e.g., mean and standard deviation)
  - Once probability distribution is known, can use it to estimate the conditional probability P(Ai|c)

AxI Technologies

Naïve Bayes : Estimates Probabilities from data

Tid	Refund	Marital Status	Taxable Income	Evade
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Compute :

- P(Status=Married|Yes) = ?
- P(Refund=Yes|No) = ?
- P(Status=Divorced|Yes) = ?
- P(TaxableInc > 80K|Yes) = ?
- P(TaxableInc > 80K|NO) = ?

Naïve Bayes : Estimates Probabilities from data

Tid	**Refund**	**Marital Status**	**Taxable Income**	**Evade**
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Compute :

- P(Status=Married|Yes) = 0/3
- P(Refund=Yes|No) = 3/7
- P(Status=Divorced|Yes) = 1/3
- P(TaxableInc > 80K|Yes) = 3/3
- P(TaxableInc > 80K|NO) = 4/7

Naïve Bayes : Estimates Probabilities from data (Exercice)

Given a Test Record: $X = (\text{Refund} = \text{No}, \text{Married}, \text{Income} >= 80\text{K})$

REFUND	P(Refund=Yes|No) = 3/7 P(Refund=No|No) = 4/7 P(Refund=Yes|Yes) = 0 P(Refund=No|Yes) = 1
MARITAL STATUS	P(Marital Status=Single|No) = 2/7 P(Marital Status=Divorced|No) = 1/7 P(Marital Status=Married|No) = 4/7 P(Marital Status=Single|Yes) = 2/7 P(Marital Status=Divorced|Yes) = 1/7 P(Marital Status=Married|Yes) = 0
TAXABLE INCOMING	P(TaxableInc >= 80K|Yes) = 3/3 P(TaxableInc >= 80K|NO) = 4/7 P(TaxableInc < 80K|Yes) = 0/3 P(TaxableInc < 80K|NO) = 3/7

Class=No	7/10
Class=Yes	3/10

$$P(C_j|A_1, \ldots, A_n) = P(A_1, \ldots, A_n |C_j) P(C_j) =$$
$$P(A_1| C_j)\ldots P(A_n| C_j) P(C_j)$$

AxIO Technologies

Naïve Bayes : Estimates Probabilities from data (Solution)

- **P(X|Class=No)** = P(Refund=No|Class=No)
  $\times$ P(Married| Class=No)
  $\times$ P(Income>=80K| Class=No)
  = 4/7 $\times$ 4/7 $\times$ 4/7 = **0.1865**

P(X|No)P(No) =
0.1865 * 0.7 = 0.1306

- **P(X|Class=Yes)** = P(Refund=No| Class=Yes)
  $\times$ P(Married| Class=Yes)
  $\times$ P(Income>=80K| Class=Yes)
  = 1 $\times$ 0 $\times$ 1 = **0**

P(X|Yes)P(Yes) =
0 * 0.3 = 0

Since P(X|No)P(No) > P(X|Yes)P(Yes)

Therefore P(No|X) > P(Yes|X)
=> **Class = No**

Naïve Bayes : Summary

- Robust to isolated noise points
- Model each class separately
- Robust to irrelevant attributes
- Use the whole set of attribute to perform classification
- Independence assumption may not hold for some attributes

Supervised Learning : Instance-Based Classifier (KNN)

- Lazy approach to classification
- Uses all the training set to perform classification
- Uses distances between training and test records

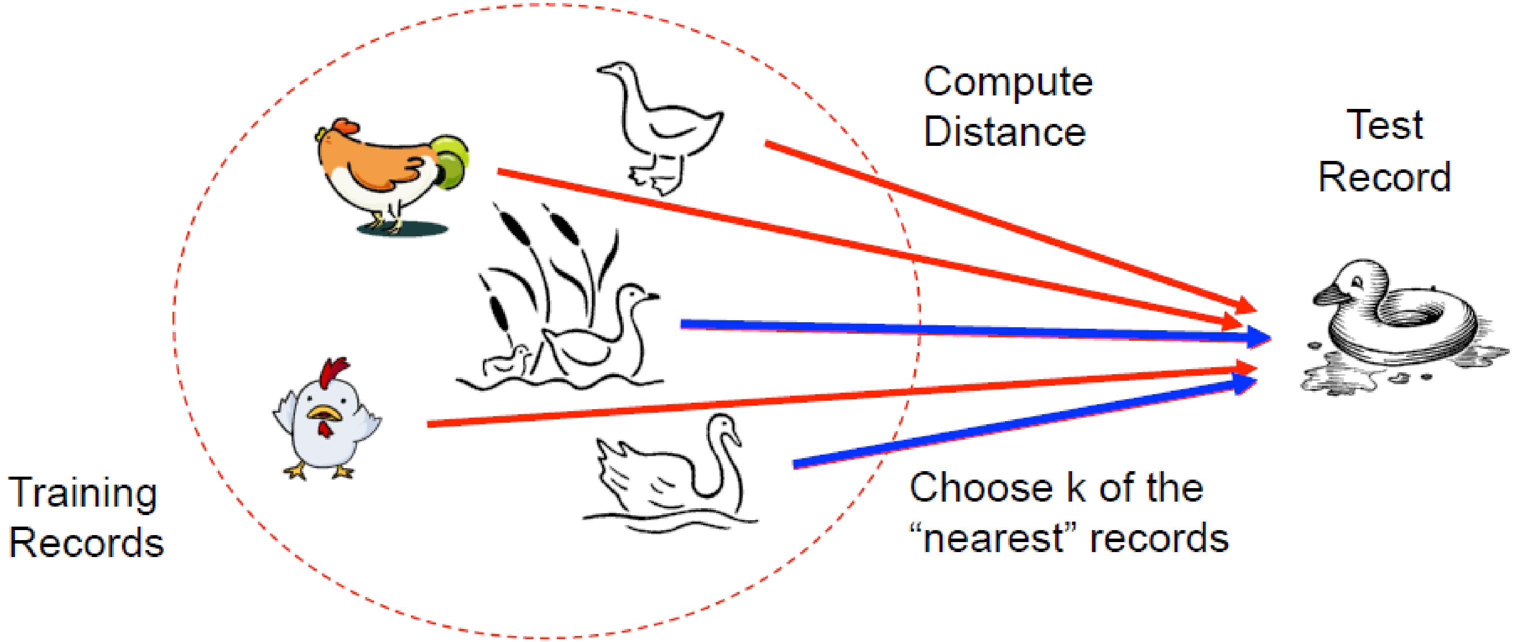
Supervised Learning : Instance-Based Classifier (KNN)

- Lazy approach to classification
- Uses all the training set to perform classification
- Uses distances between training and test records

Supervised Learning : Instance-Based Classifier (KNN)

Principle :

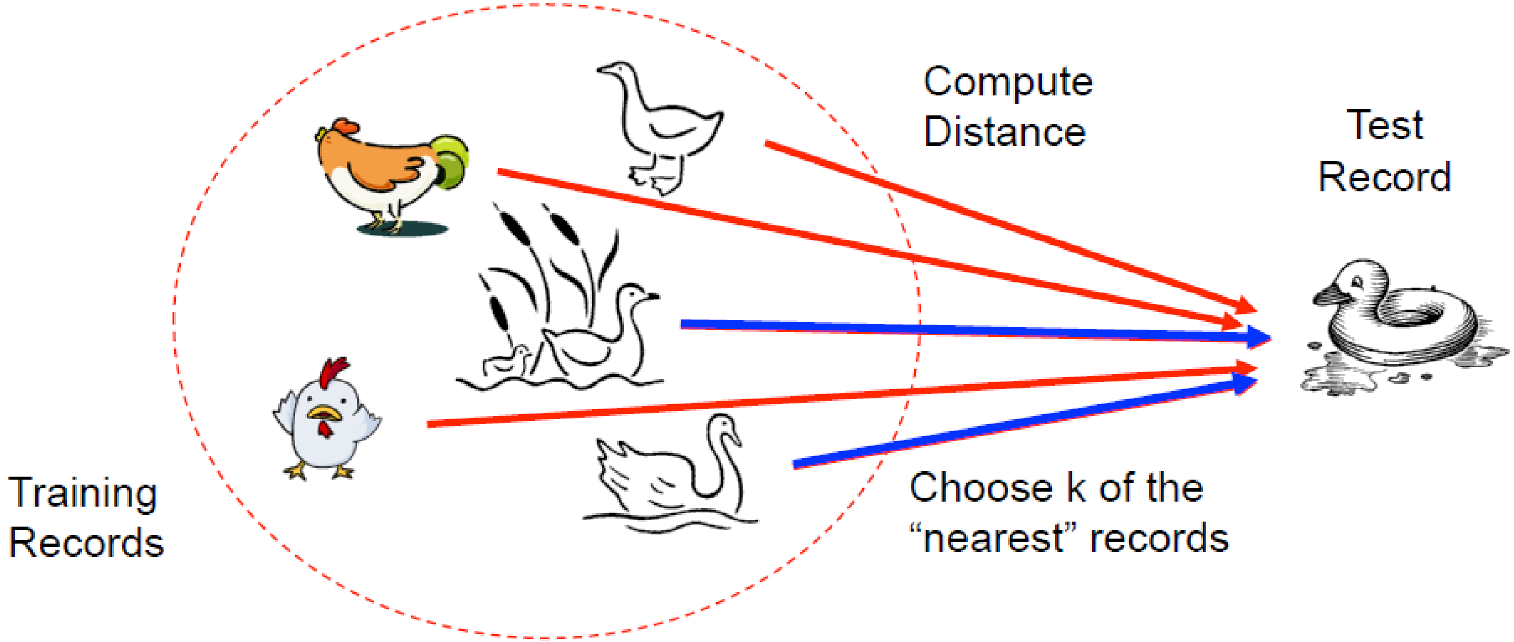
If it walks like a duck, quacks like a duck, then it's probably a duck



Supervised Learning : Instance-Based Classifier (KNN)

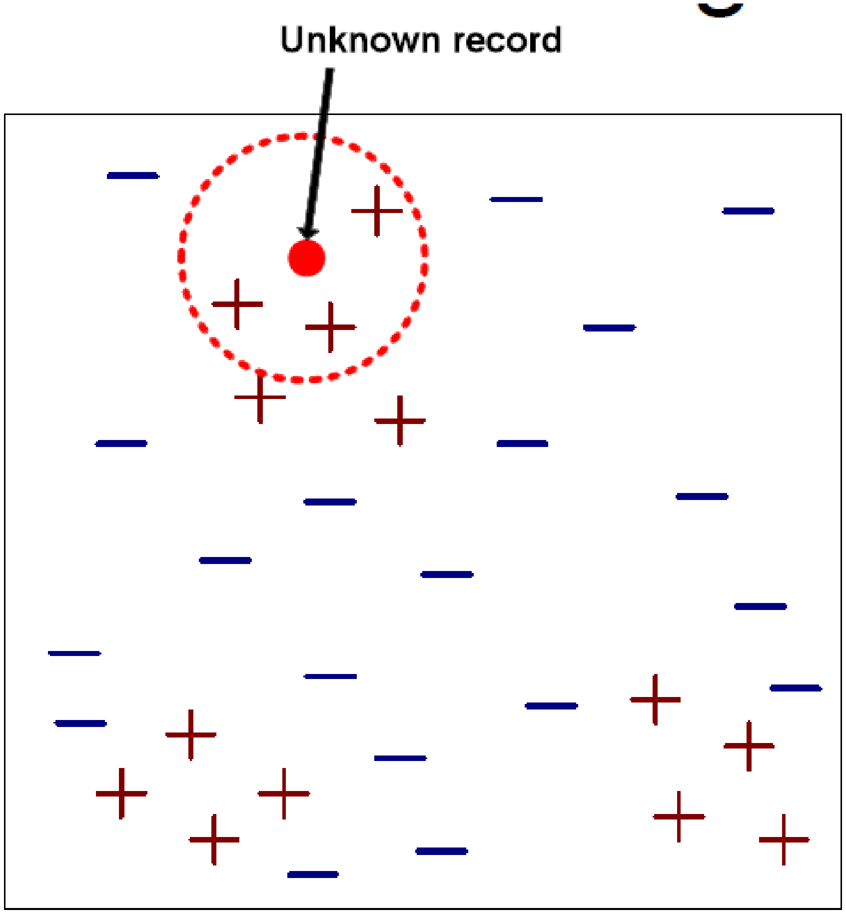
Principle :

If it walks like a duck, quacks like a duck, then it's probably a duck



Supervised Learning : Instance-Based Classifier (KNN)

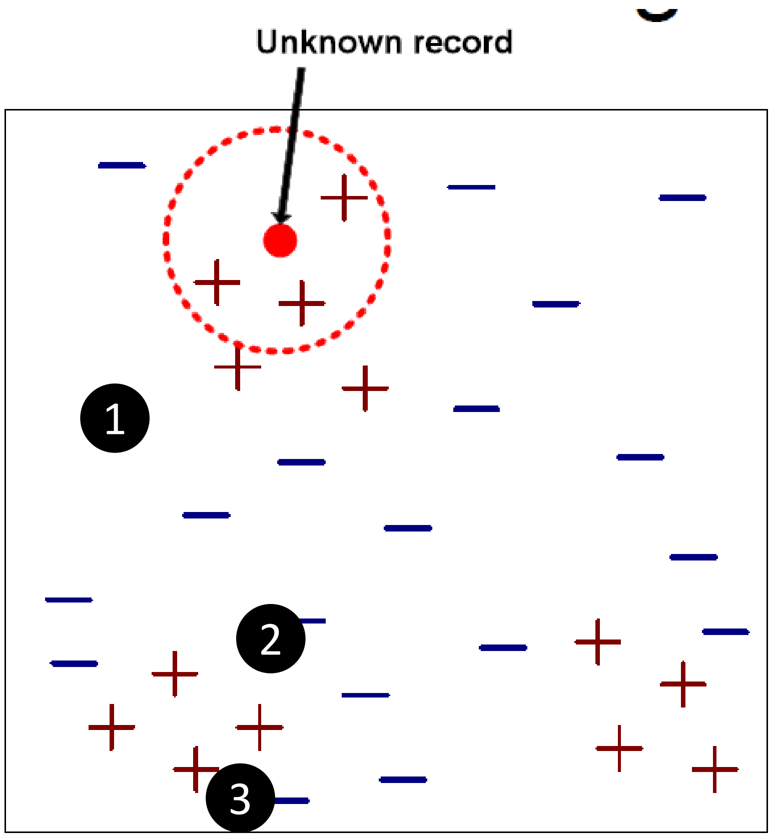
Principle :



- Requires three things
  - The set of stored records
  - Distance Metric to compute distance between records
  - The value of *k*, the number of nearest neighbors to retrieve
- To classify an unknown record:
  - Compute distance to other training records
  - Identify *k* nearest neighbors
  - Use class labels of nearest neighbors to determine the class label of unknown record (e.g., by taking majority vote)

Supervised Learning : Instance-Based Classifier (KNN)

Principle :



- Requires three things
  - The set of stored records
  - Distance Metric to compute distance between records
  - The value of *k*, the number of nearest neighbors to retrieve

- To classify an unknown record:
  - Compute distance to other training records
  - Identify *k* nearest neighbors
  - Use class labels of nearest neighbors to determine the class label of unknown record (e.g., by taking majority vote)

Exercise : Give the class attributed to each instance for k = 1, 2, or 5