MACHINE LEARNING

Syllabus Scheme

Subject Code	Subject Name	Credits
MCADLE5042	**Machine Learning**	**04**

Subject Code	Subject Name	Teaching Scheme			Credits Assigned			
		Theory	Pract.	Tut	Theory	Pract.	Tut	Total
MCADLE5042	**Machine Learning**	**04**	**--**	**--**	**04**	**--**	**--**	**04**

Subject Code	Subject Name	Examination Scheme							
		Theory Marks				TW	Pract.	Oral	Total
		Internal Assessment			End Semester Exam				
		Test1 (T1)	Test2(T2)	Average of T1 & T2					
MCADLE5042	**Machine Learning**	**20**	**20**	**20**	**80**	**--**	**--**	**--**	**100**

CEO & CO

Pre-requisites:

Understanding of basic computer science concepts, data structures and good understanding of Mathematical Concepts is required.

Course Educational Objectives (CEO): At the end of the course, the students will be able to

CEODLE5042. 1	Understand Machine Learning and its techniques.
CEODLE5042. 2	Study regression, classification with AdaBoost and clustering methods.
CEODLE5042. 3	Understand support vector machine, Dimensionality reduction, Anomaly Detection, Recommender Systems

Course Outcomes (CO): At the end of the course, the students will be able to

MCADLE5042.1	Analyze the Machine Learning techniques.
MCADLE5042.2	Apply regression, classification with AdaBoost and clustering methods to real world applications.
MCADLE5042.3	Describe support vector machine, Dimensionality reduction, Anomaly Detection, Recommender Systems

Syllabus-ML

Sr. No.	Module	Detailed Contents	Hrs
1	**Understand Machine Learning**	Introduction to Machine Learning, Overview of Machine Learning, Key Terminology and task of ML, Applications of ML, Software Tools, Introduction to Big Data and Machine Learning, Hypothesis space, Estimate hypothesis accuracy, Hypothesis testing	**06**
2	**Supervised Learning-Classification**	Introduction to Supervised Learning: Classification, Decision Tree Representation- Appropriate problem for Decision Learning, Decision Tree Algorithm, Hyperspace Search in Decision Tree Naive Bayes- Bayes Theorem , Classifying with Bayes Decision Theory , Conditional Probability, Bayesian Belief Network	**08**
3	**Supervised Learning-Regression**	Regression: Linear Regression- Predicting numerical value, Finding best fit line with linear regression, Regression Tree- Using CART for regression Logistic Regression - Classification with Logistic Regression and the Sigmoid Function	**08**

Syllabus-ML

4	**Support Vector Machine**	Introduction : Separating data with maximum margin, Finding the maximum margin, Effective optimization with SMO algorithm	**08**
5	**Improving classification with the AdaBoost**	Classifier using multiple samples of the data set, Improving classifier by focusing on error, weak learner with a decision stump, Implementing the AdaBoost algorithm, Classifying with AdaBoost	**08**
6	Unsupervised Learning	Clustering: Learning from unclassified data –Introduction to clustering, K- Mean Clustering, Expectation-Maximization Algorithm(EM algorithm),Hierarchical Clustering, Supervised Learning after clustering	**08**
7	**Additional Core Techniques**	Dimensionality reduction- Dimensionality reduction techniques, Principal component analysis, Anomaly Detection, Recommender Systems	**06**

References:

Reference:

- Machine Learning in Action By Peter Harrington By Manning
- Machine Learning, T. Mitchell, McGraw-Hill, 1997.
- Introduction to Machine LearningBy EthemAlpaydin,MIT Press
- Understanding Machine Learning From Theory to Algorithms By ShaiShalev-Shwartz and Shai Ben David, Cambridge University Press
- Data Mining Concepts and Techniques, J. Han and Kamber

Web References:

- http://www.infoworld.com/article/2853707/robotics/11-open-source-tools-machine-learning.html#slide12
- http://www.ibm.com/developerworks/library/os-recommender1/

What is Learning?

- *"Learning denotes changes in a system that ... enable a system to do the same task … more efficiently the next time."* - Herbert Simon
- *"Learning is constructing or modifying representations of what is being experienced."* - Ryszard Michalski
- *"Learning is making useful changes in our minds."* - Marvin Minsky

"Machine learning refers to a system capable of the autonomous acquisition and integration of knowledge."

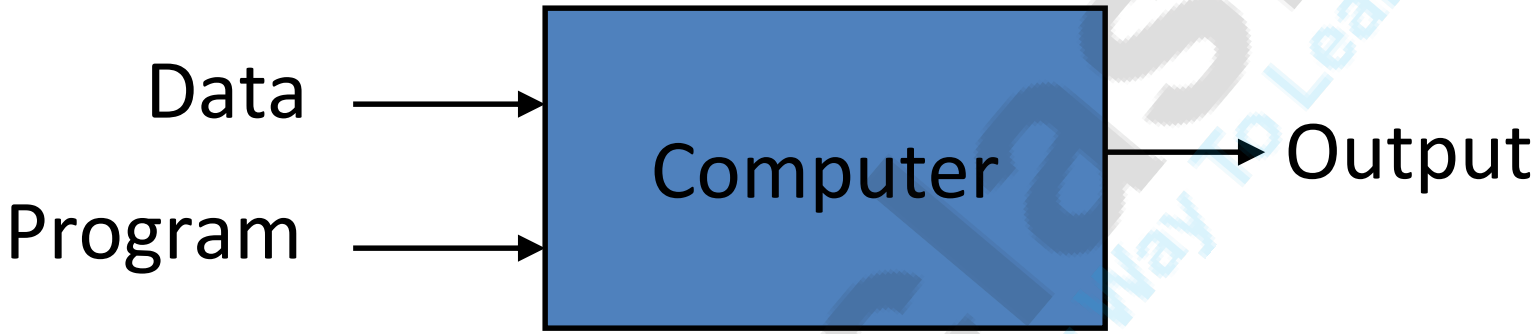
ML-More Meanings

- **Machine learning** is a subset of artificial intelligence in the field of computer science that often uses statistical techniques to give computers the ability to "learn" (i.e., progressively improve performance on a specific task) with data, without being explicitly programmed.

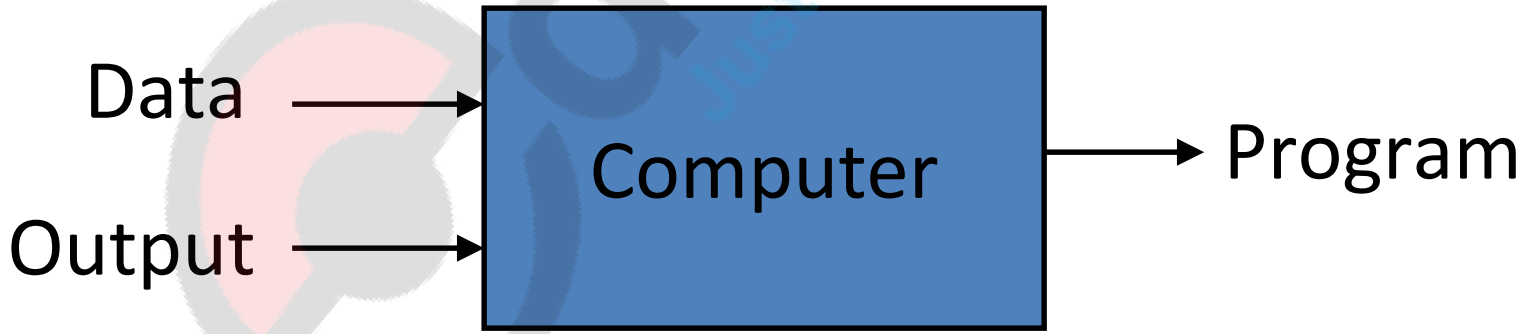
- Machine learning is programming computers to optimize a performance criterion using example data or past experience.

- **Machine learning** is an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed.

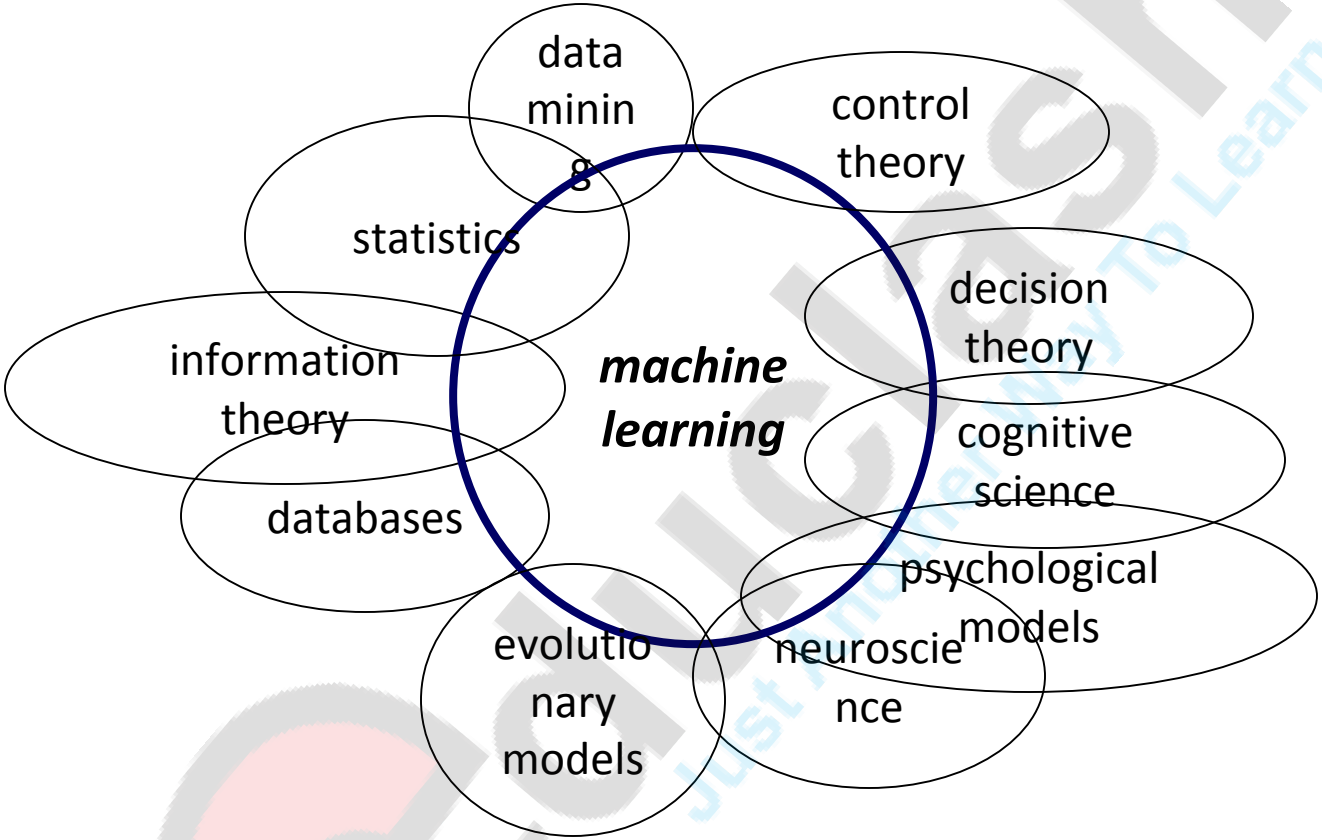
Traditional Programming



Machine Learning

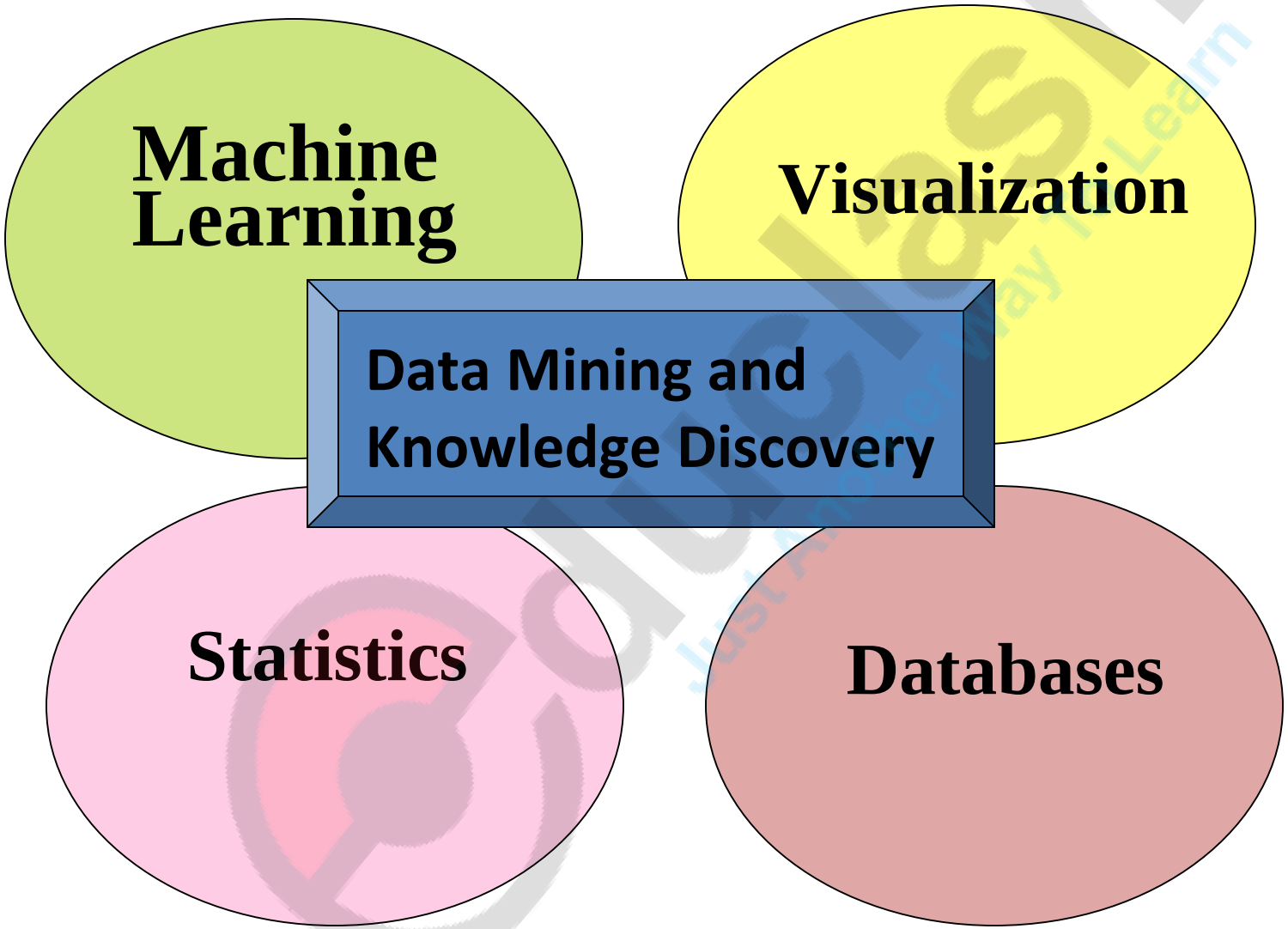


Related Fields



Machine learning is primarily concerned with the accuracy and effectiveness of the *computer system*.

Related Fields



What We Talk About When We Talk About"Learning"

- Learning general models from a data of particular examples
- Data is cheap and abundant (data warehouses, data marts); knowledge is expensive.
- Example in retail: Customer transactions to consumer behavior:

  People who bought "Da Vinci Code" also bought "The Five People You Meet in Heaven" (www.amazon.com)

- Build a model that is *a good and useful approximation* to the data.

Data Mining/KDD

Definition := *"KDD is the non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data"*

Applications:

- Retail: Market basket analysis, Customer relationship management (CRM)
- Finance: Credit scoring, fraud detection
- Manufacturing: Optimization, troubleshooting
- Medicine: Medical diagnosis
- Telecommunications: Quality of service optimization
- Web mining: Search engines

Big Data and Machine Learning

- Big data analytics is the process of collecting and analyzing the large volume of data sets (called Big Data) to discover useful hidden patterns and other information like customer choices, market trends that can help organizations make more informed and customer oriented business decisions.
- Big data is a term that describes the data characterized by 3Vs: the extreme volume of data, the wide variety of data types and the velocity at which the data must be processed.
- Big data can be analyzed for insights that lead to better decisions and strategic business moves.

Machine Learning

- Machine learning is a field of AI (Artificial Intelligence)
- to increase their accuracy for the expecting outcomes.
  - You know those movie/show recommendations you get on Netflix or Amazon? Machine learning does this for you.
  - How does Uber/Ola determine the price of your cab ride? How do they minimize the wait time once you hail a car? How do these services optimally match you with other passengers to minimize detours? The answer to all these questions is Machine Learning.
  - How can a financial institution determine if a transaction is fraudulent or not? In most cases, it is difficult for humans to manually review each transaction because of its very high daily transaction volume. Instead, AI is used to create systems that learn from the available data to check what types of transactions are fraudulent.
  - Ever wondered what's the technology behind the self-driving Google car? Again the answer is machine learning.
- Now we know What Big Data vs Machine Learning are, but to decide which one to use at which place we need to see the difference between both.

#1. Data Use

Big Data	Machine Learning
Big data can be used for a variety of purposes, including financial research, collecting sales data etc.	Machine learning is the technology behind self-driving cars and advance recommendation engines.

#2. Foundations for Learning

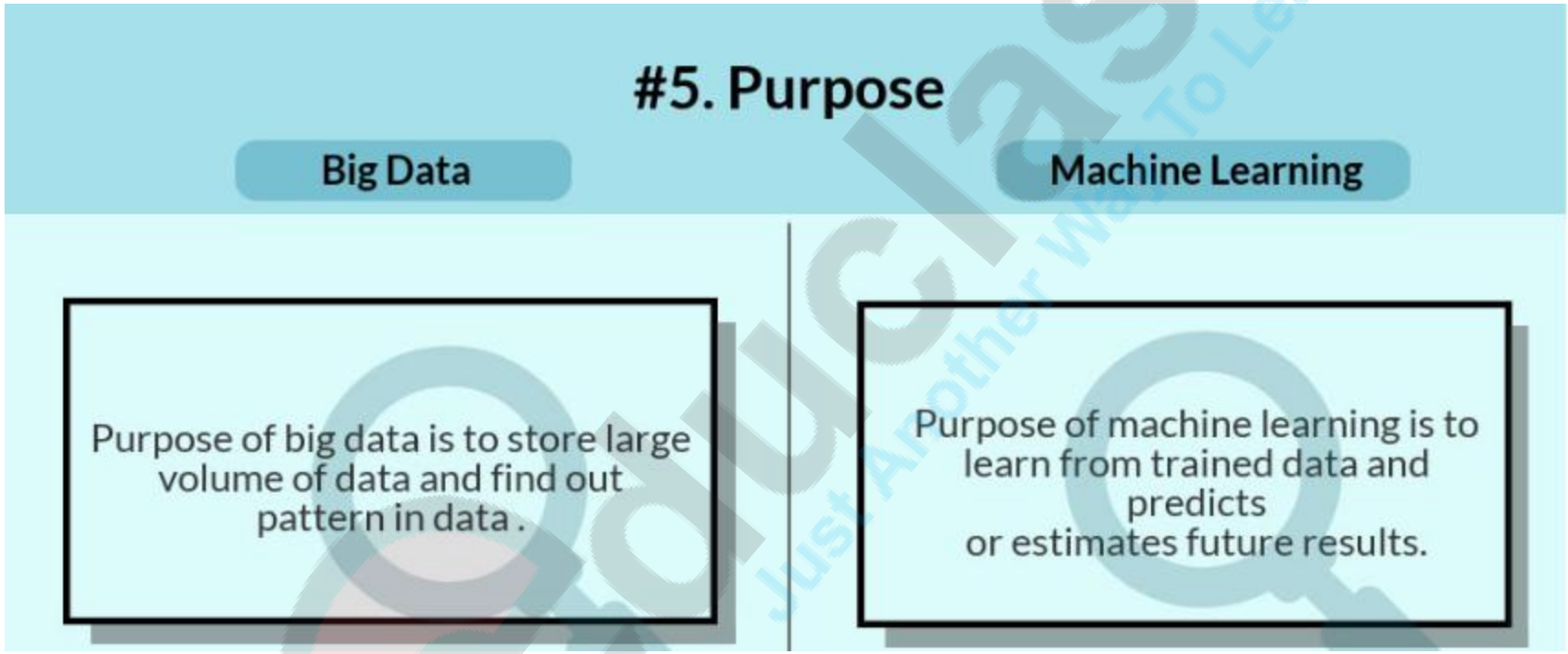
Big Data	Machine Learning
Big data analytics pulls from existing information to look for emerging patterns that can help shape our decision-making processes.	On the other hand, Machine learning can learn from the existing data and provide the foundation required for a machine to teach itself.

#3. Pattern Recognition

Big Data	Machine Learning
	
Big data analytics can reveal some patterns through classifications and sequence analysis.	However, machine learning takes this concept a one step ahead by using the same algorithms that big data analytics uses to automatically learn from the collected data.

#4. Data Volume

Big Data	Machine Learning
	
Big data as the name suggest tends to be interested in large-scale datasets where the problem is dealing with the large volume of data.	ML tends to be more interested in small datasets where over-fitting is the problem.


#5. Purpose
Big Data
Machine Learning
Purpose of big data is to store large volume of data and find out pattern in data .
Purpose of machine learning is to learn from trained data and predicts
or estimates future results.

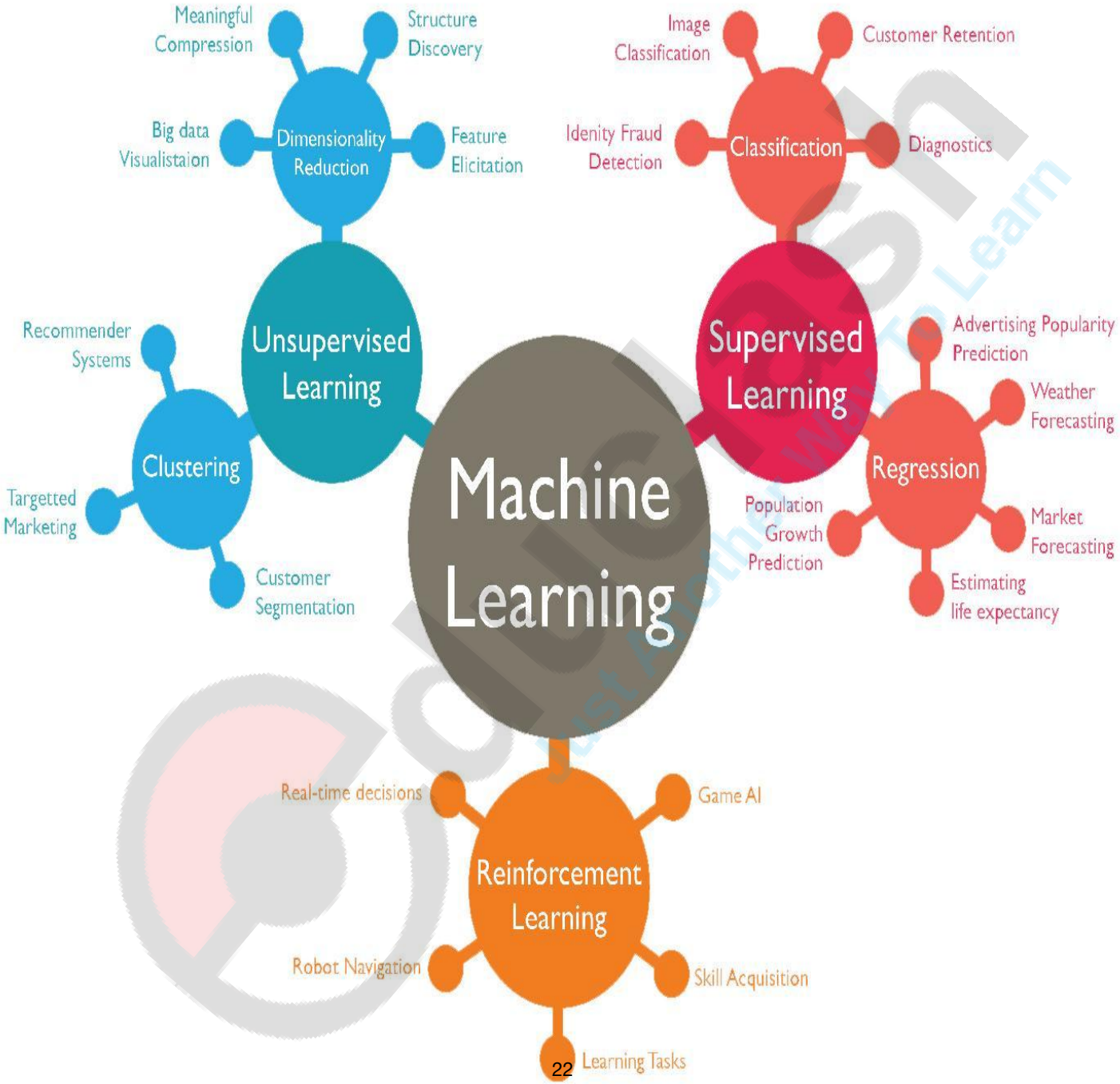
Examples of ML problems

- Face detection: find faces in images(if face is present or not).
- Spam filtering: identify email msg as spam or not.
- OCR: categorize images of handwritten characters by letters represented.
- Fraud detection: identify credit card transactions that would be fraudulent in nature.
- Weather prediction: predict, whether or not it will rain tomorrow.
- Medical diagnosis: diagnose a patient as as sufferer or non-sufferer of some disease.
- Stock Market: classification (stock to go up or down), regression (how much will the stock goes up)
- Real estate: predict how much a house will sell for.
- Automatic recommender systems (collaborative filtering)

Learning Methodologies

- **Learning from labelled data (supervised learning) eg. Classification, regression, prediction**
- **Learning from unlabelled data (unsupervised learning) eg. Clustering, visualization, dimensionality reduction.**

  Data unlabelled and algo learn to inherent structure from input data.
  - Competitive learning: Input pattern matched against the node with most similar weights(Winner-take-all)
- **Semi-supervised learning** Eg: photo
  - » Large amt of data and only some data is labelled.
- **Reinforcement Learning**


Machine Learning
Unsupervised Learning
Dimensionality Reduction
Meaningful Compression
Structure Discovery
Big data Visualistaion
Feature Elicitation
Clustering
Recommender Systems
Targetted Marketing
Customer Segmentation
Supervised Learning
Classification
Image Classification
Customer Retention
Idenity Fraud Detection
Diagnostics
Regression
Advertising Popularity Prediction
Weather Forecasting
Population Growth Prediction
Market Forecasting
Estimating life expectancy
Reinforcement Learning
Real-time decisions
Game AI
Robot Navigation
Skill Acquisition
Learning Tasks

Key tasks of machine learning

Supervised learning tasks	
k-Nearest Neighbors	Linear
Naive Bayes	Locally weighted linear
Support vector machines	Ridge
Decision trees	Lasso
Unsupervised learning tasks	
k-Means	Expectation maximization
DBSCAN	Parzen window

Table 1.2 Common algorithms used to perform classification, regression, clustering, and density estimation tasks

How to choose the right algorithm

- If you're trying to predict or forecast a target value, then you need to look into supervised learning.
- discrete value like Yes/No, 1/2/3, A/B/C, or Red/Yellow/Black?
- If so, then you want to look into classification.
- If the target value can take on a number of values, say any value from 0.00 to 100.00, or -999 to 999, or + to -, then you need to look into regression.
- Else then unsupervised learning

How to choose the right algorithm

- trying to fit your data into some discrete groups?
- you should look into clustering.
- Do you need to have some numerical estimate of how strong the fit is into each group?
- If you answer yes, then you probably should look into a density estimation algorithm.

Steps in developing a machine learning application

- *Collect data.*
- *Prepare the input data.*
- *Analyze the input data.*
- *Train the algorithm.*
- *Test the algorithm*
- *Use it.*

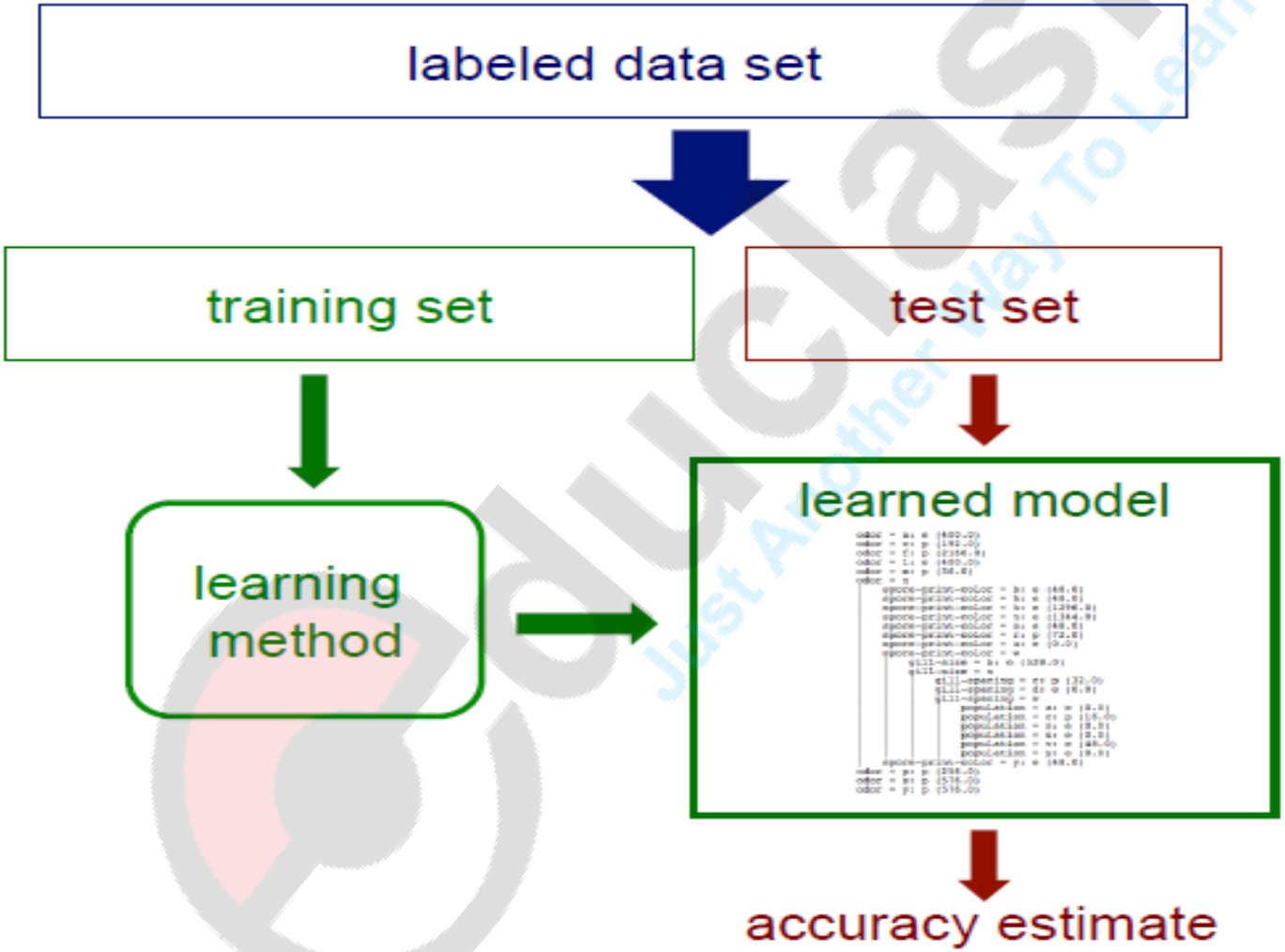
Machine Learning / Data Mining Application areas

- Science
  - astronomy, bioinformatics, drug discovery, …
- Business
  - CRM (Customer Relationship management), fraud detection, e-commerce, manufacturing, sports/entertainment, telecom, targeted marketing, health care, …
- Web:
  - search engines, advertising, web and text mining, …
- Government
  - surveillance (?|), crime detection, profiling tax cheaters, …

Software Tools – Machine Learning

- Scikit-learn Developers. Scikit-learn.
- Super Data Science
- The Shogun Team. Shogun.
- Accord.Net Framework. Project: Accord Framework/AForge.net
- **Apache Singa :** *http://singa.apache.org/en/index.html*
- Apache **Software** Foundation. Apache Mahout
- Apache **Software** Foundation. Spark Mllib
  - practical machine learning scalable and easy. (all algms) *http://spark.apache.org/mllib/*
- **TensorFlow by Google**
- Cloudera Oryx2 built on Apache Spark and Apache Kafka for real-time large scale machine learning.
- **Amazon Machine Learning (AML)** *https://aws.amazon.com/machine-learning/*

Hypothesis space



Hypothesis Space Example

Example

Suppose an example with four binary features and one binary output variable. Below is a set of observations:

- x1 x2 x3 x4 | y ----------
- 0 0 0 1 | 0
- 0 1 0 1 | 0
- 1 1 0 0 | 1
- 0 0 1 0 | 1

This set of observations can be used by a **machine learning (ML) algorithm** to learn a function **f** that is able to predict a value y for any input from the **input space**.

- We are searching for the ground truth f(x) = y that explains the relation between x and y for all possible inputs in the correct way.
- The function f has to be chosen from the **hypothesis space**.

To get a better idea:

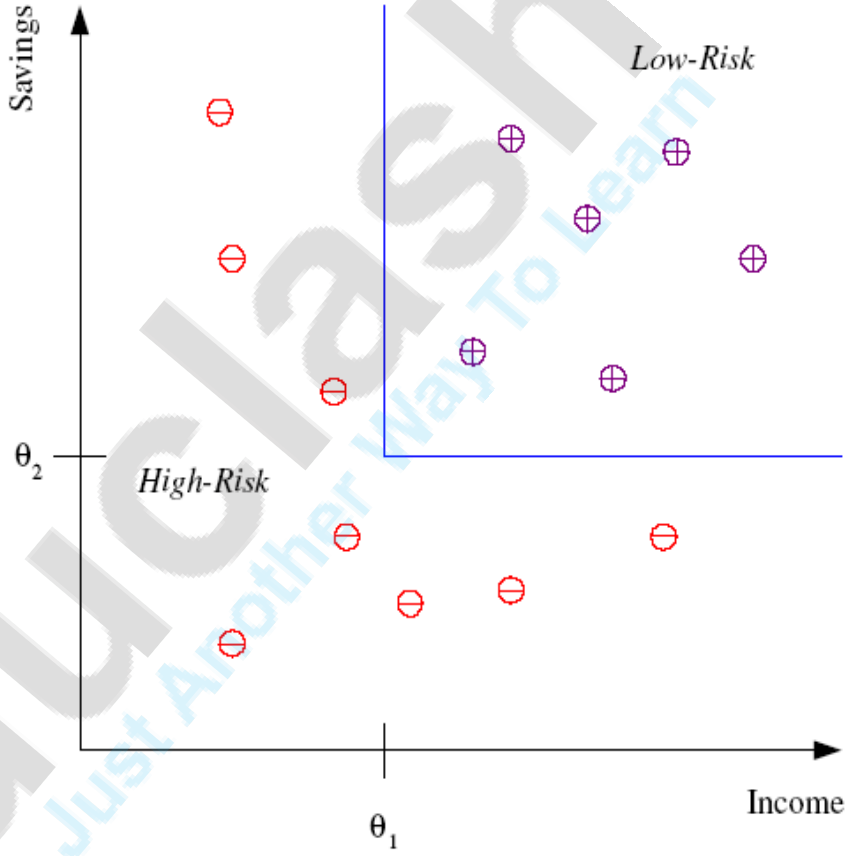
- The input space is in the above given example , its the number of possible inputs.
- The hypothesis space is because for each set of features of the input space two outcomes (0 and 1) are possible.
- The ML algorithm helps us to find **one function**, sometimes also referred as hypothesis, from the relatively large hypothesis space.

Applications

- Association Analysis
- Supervised Learning
  - Classification
  - Regression/Prediction
- Unsupervised Learning
- Reinforcement Learning

Classification

- Example: 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

Model

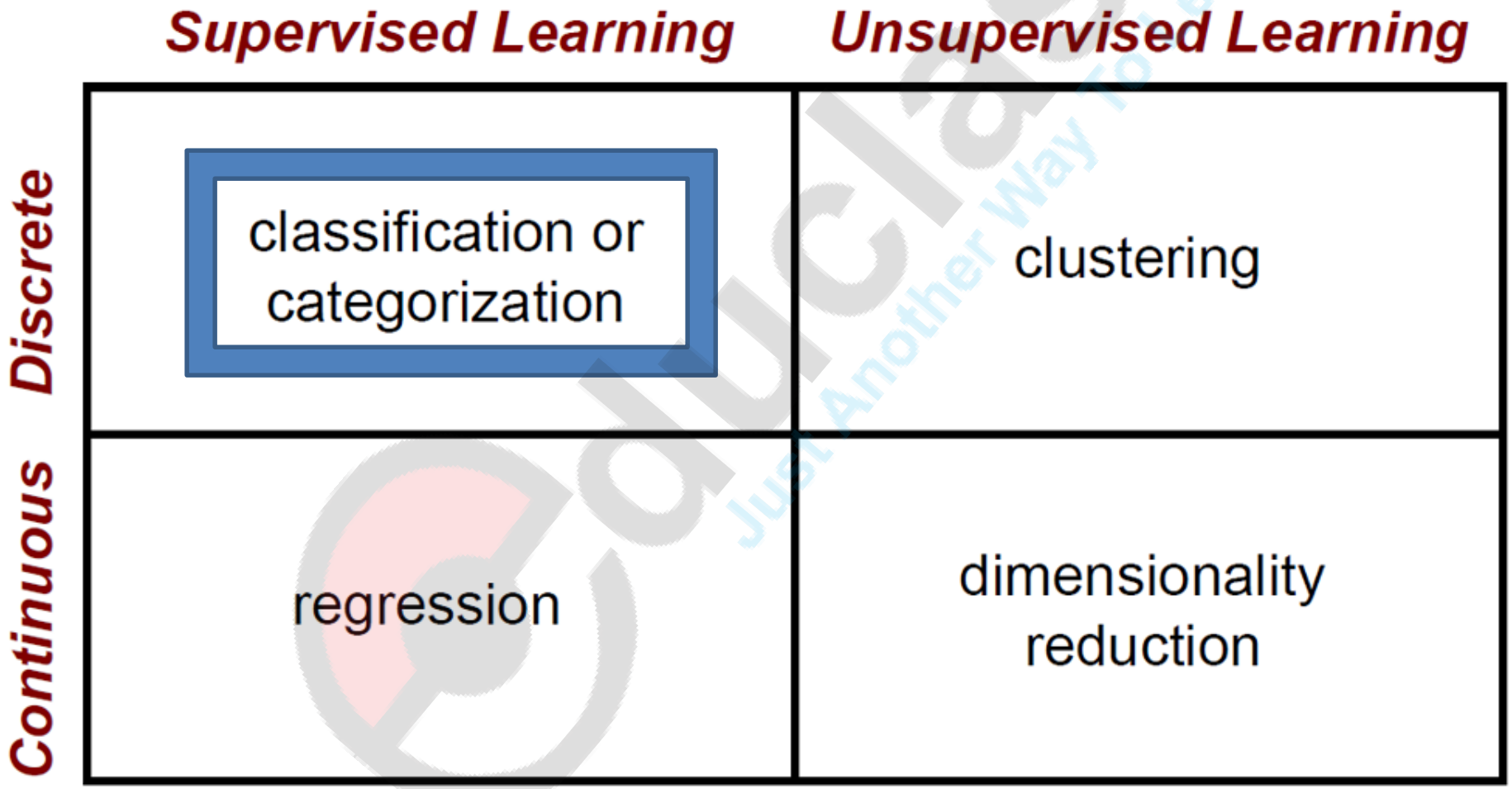
Classification: Applications

- Pattern recognition
- Face recognition: Pose, lighting, make-up, hair style
- Character recognition: Different handwriting styles.
- Speech recognition:
  - Use of a dictionary or the syntax of the language.
  - Sensor fusion: Combine multiple modalities; eg, visual (lip image) and acoustic for speech
- Medical diagnosis: From symptoms to illnesses
- Web Advertizing: Predict if a user clicks on an ad on the Internet.

Reinforcement Learning

- Topics:
  - Policies: what actions should an agent take in a particular situation
  - Utility estimation: how good is a state (→used by policy)
- No supervised output but delayed reward
- Credit assignment problem (what was responsible for the outcome)
- Applications:
  - Game playing
  - Robot in a maze
  - Multiple agents, partial observability, ...

Machine Learning Problems



The machine learning framework

- Apply a prediction function to a feature representation of the image to get the desired output:

f() = “apple”

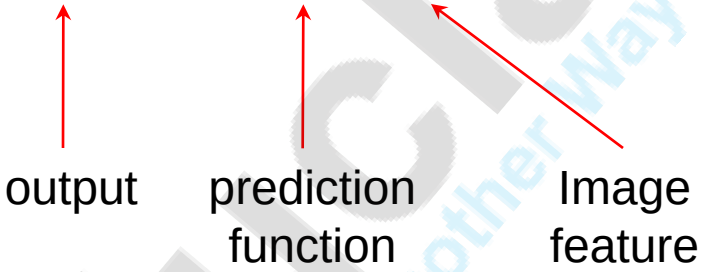
f() = “tomato”

f() = “cow”

Slide credit: L. Lazebnik

The machine learning framework

$$y = f(\mathbf{x})$$



- **Training:** given a *training set* of labeled examples $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$, estimate the prediction function f by minimizing the prediction error on the training set
- **Testing:** apply f to a never before seen *test example* $\mathbf{x}$ and output the predicted value $y = f(\mathbf{x})$

Slide credit: L. Lazebnik

Steps

Trainin g

Training Images

Training Labels

Image Features

Training

Learned model

Testin g

Test

Image Features

Learned model

Prediction

Slide credit: D. Hoiem and L. Lazebnik

Supervised Learning- Classification

- Introduction to Supervised Learning:
- Classification, Decision Tree Representation- Appropriate
- problem for Decision Learning, Decision Tree Algorithm,
- Hyperspace Search in Decision Tree
- Naive Bayes- Bayes Theorem , Classifying with Bayes Decision
- Theory , Conditional Probability, Bayesian Belief Network

Basics :

- Supervised learning asks the machine to learn from our data when we specify a target variable.
- This reduces the machine's task to only divining some pattern from the input data to get the target variable.

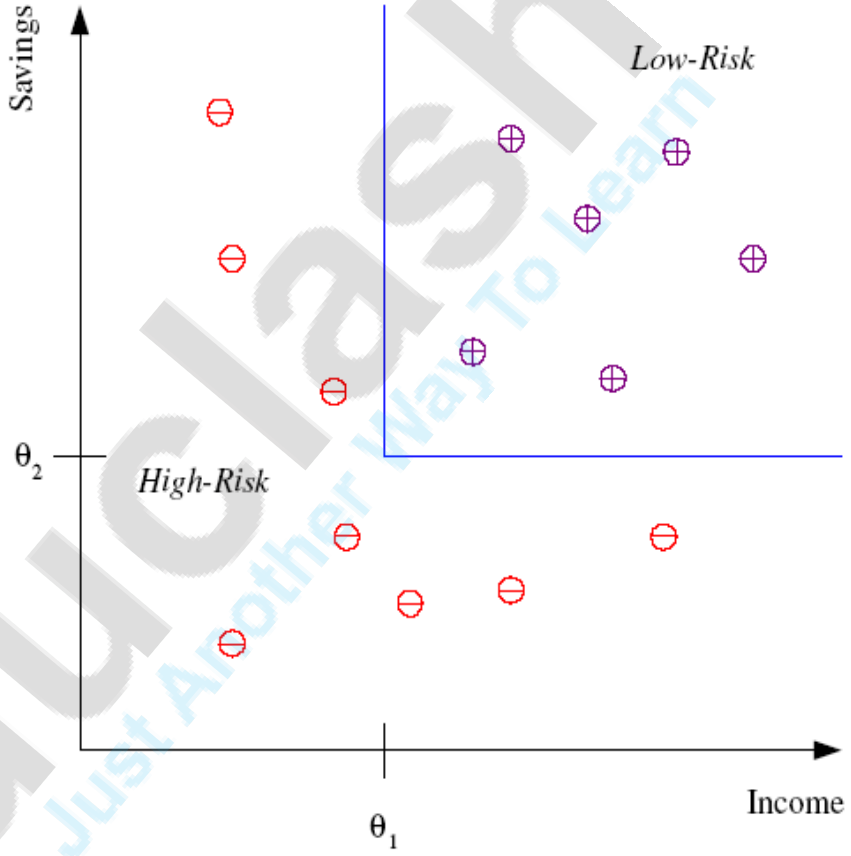
- We address two cases of the target variable. The first case occurs when the target Variable can take only nominal values: true or false;
- The second case of classification occurs when the target variable can take an infinite number of numeric values, such as 0.100, 42.001, 1000.743, ...
- This case is called regression.

I. Classification vs. Prediction

- Classification
  - predicts categorical class labels (discrete or nominal)
  - classifies data (constructs a model) based on the training set and the values (class labels) in a classifying attribute and uses it in classifying new data
- Prediction
  - models continuous-valued functions, i.e., predicts unknown or missing values
- Typical applications
  - Credit approval
  - Target marketing
  - Medical diagnosis
  - Fraud detection

Classification

- Example: Credit scoring
- Differentiating between low-risk and high-risk customers from their *income* and *savings*



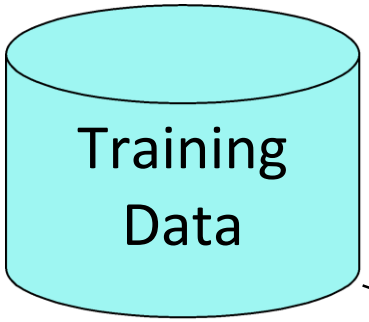
Discriminant: IF *income* > θ_1 AND *savings* > θ_2
THEN low-risk ELSE high-risk

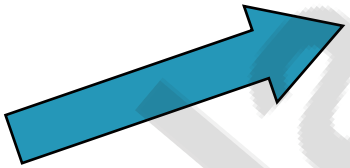
Model

Classification—A Two-Step Process

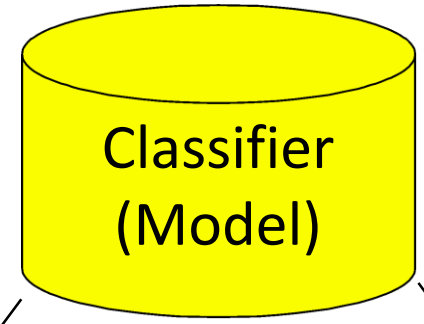
- Model construction: describing a set of predetermined classes
  - Each tuple/sample is assumed to belong to a predefined class, as determined by the class label attribute
  - The set of tuples used for model construction is training set
  - The model is represented as classification rules, decision trees, or mathematical formulae
- Model usage: for classifying future or unknown objects
  - Estimate accuracy of the model
    - The known label of test sample is compared with the classified result from the model
    - Accuracy rate is the percentage of test set samples that are correctly classified by the model
    - Test set is independent of training set, otherwise over-fitting will occur
  - If the accuracy is acceptable, use the model to classify data tuples whose class labels are not known

Classification Process (1): Model Construction





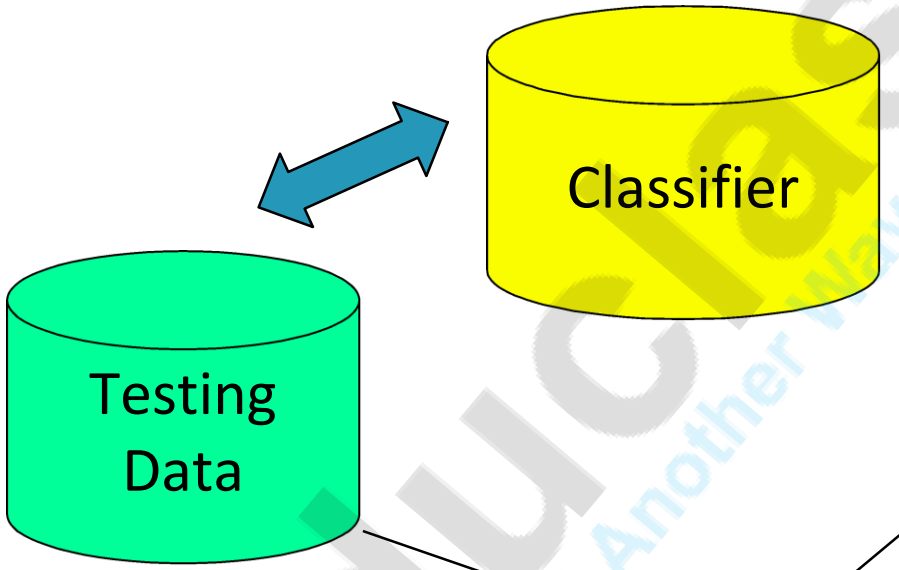
Classification
Algorithms



NAME	RANK	YEARS	TENURED
Mike	Assistant Prof	3	no
Mary	Assistant Prof	7	yes
Bill	Professor	2	yes
Jim	Associate Prof	7	yes
Dave	Assistant Prof	6	no
Anne	Associate Prof	3	no

IF rank = ‘professor’
OR years > 6
THEN tenured = ‘yes’

Classification Process (2): Use the Model in Prediction



NAME	RANK	YEARS	TENURED
Tom	Assistant Prof	2	no
Merlisa	Associate Prof	7	no
George	Professor	5	yes
Joseph	Assistant Prof	7	yes

(Jeff, Professor, 4)

Tenured ?

Supervised vs. Unsupervised Learning

- Supervised learning (classification)
  - Supervision: The training data (observations, measurements, etc.) are accompanied by labels indicating the class of the observations
  - New data is classified based on the training set
- Unsupervised learning (clustering)
  - The class labels of training data is unknown
  - Given a set of measurements, observations, etc. with the aim of establishing the existence of classes or clusters in the data

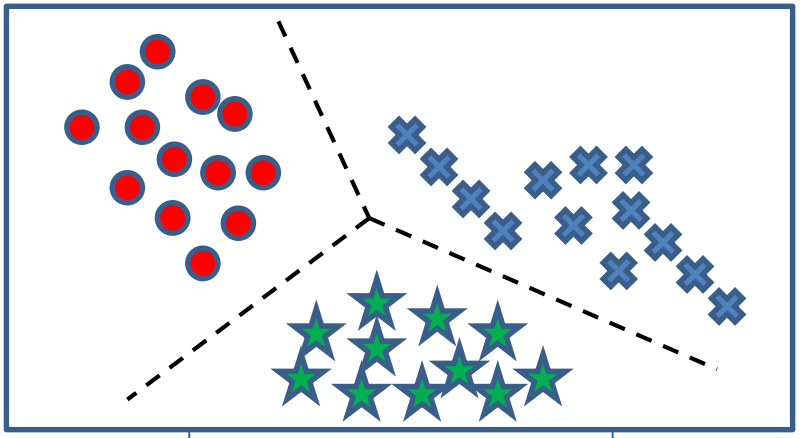
II. Issues Regarding Classification and Prediction (1): Data Preparation

- Data cleaning
  - Preprocess data in order to reduce noise and handle missing values
- Relevance analysis (feature selection)
  - Remove the irrelevant or redundant attributes
- Data transformation
  - Generalize and/or normalize data

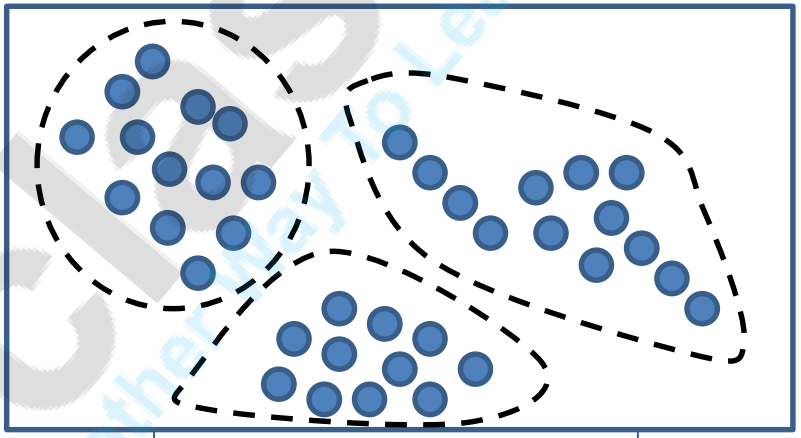
Algorithms

- **Supervised learning**
  - Prediction
  - Classification (discrete labels), Regression (real values)
- **Unsupervised learning**
  - Clustering
  - Probability distribution estimation
  - Finding association (in features)
  - Dimension reduction
- **Semi-supervised learning**
- **Reinforcement learning**
  - Decision making (robot, chess machine)

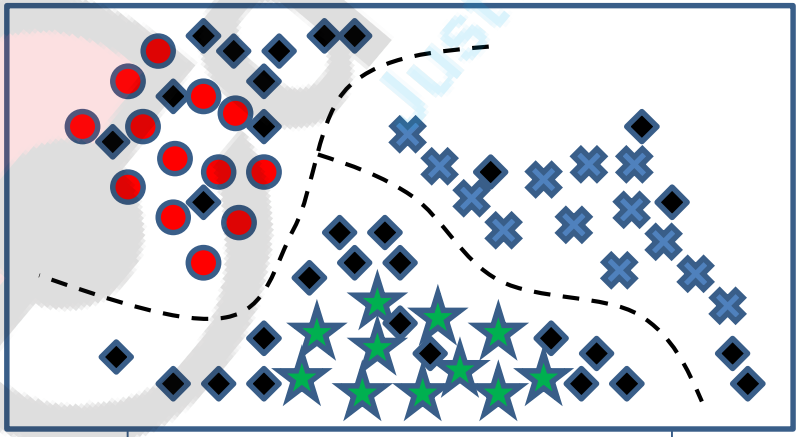
Algorithms



Supervised learning



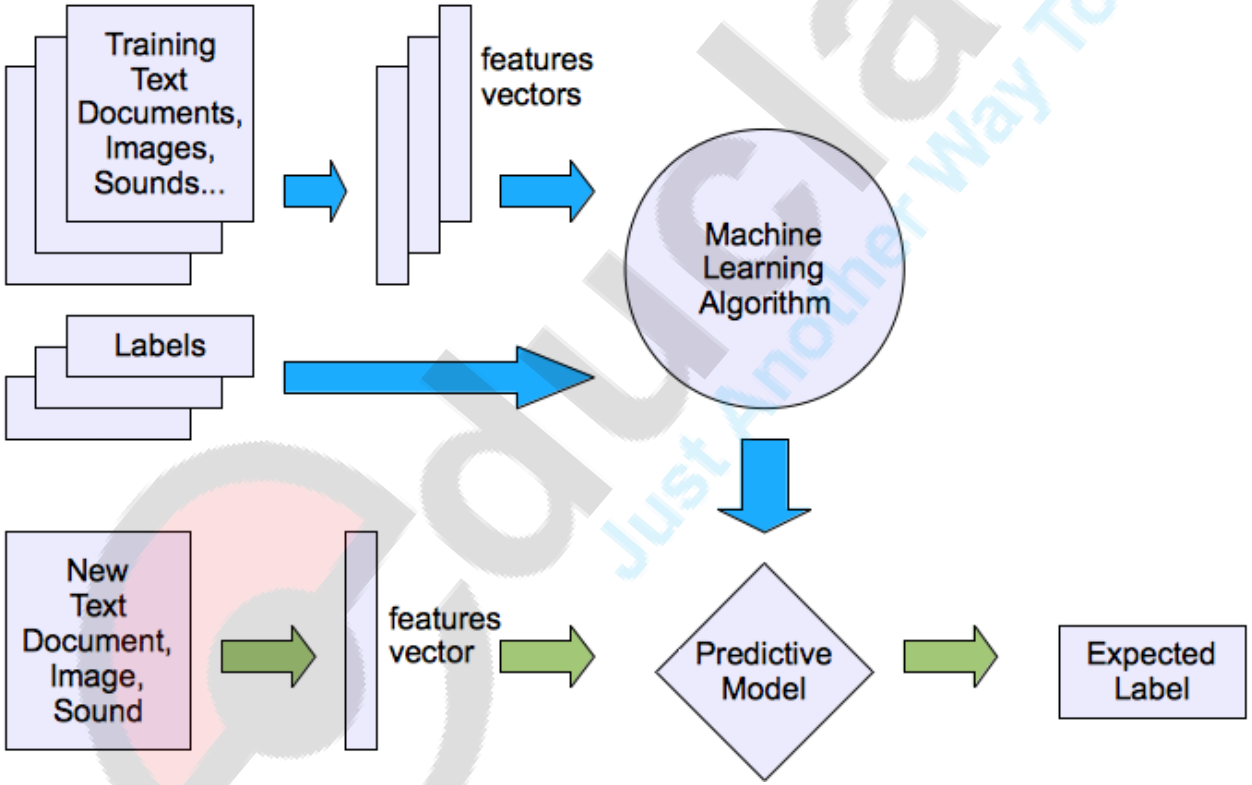
Unsupervised learning



Semi-supervised learning

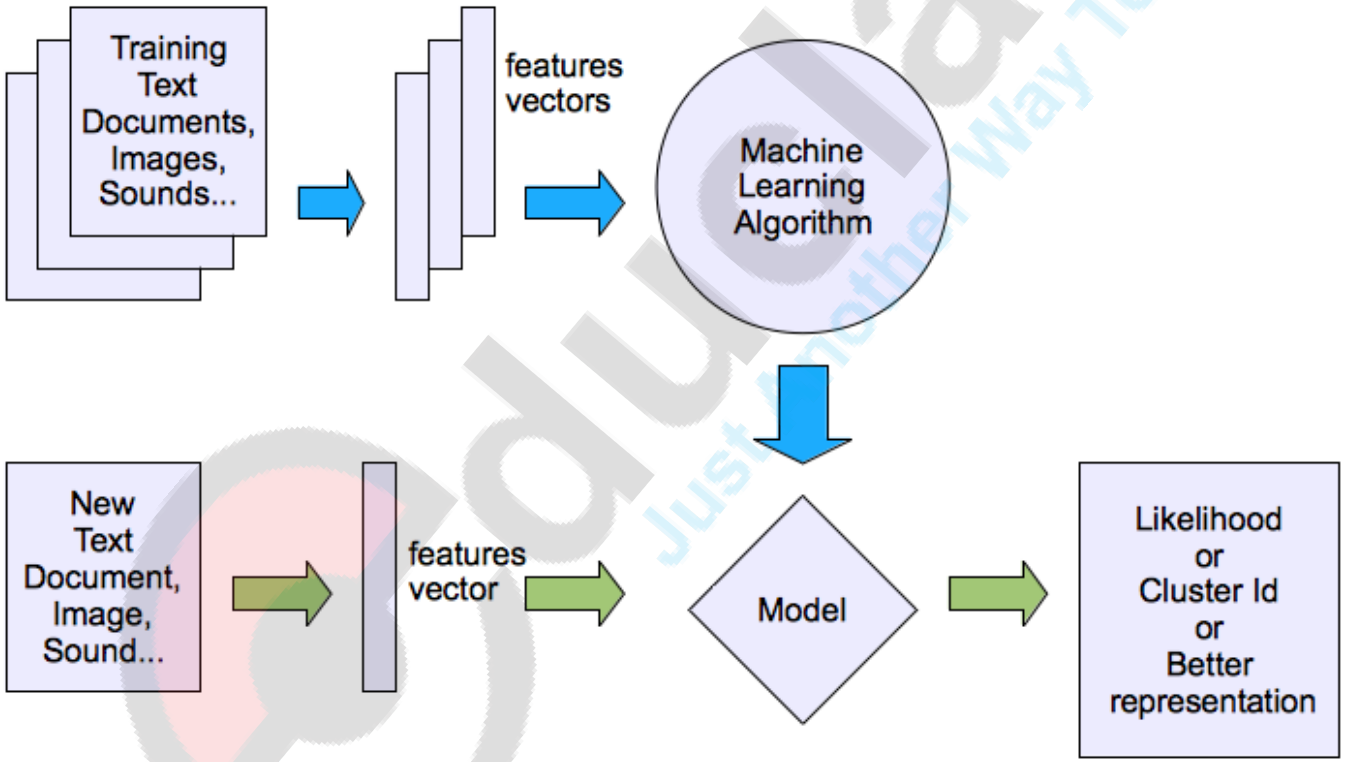
Machine learning structure

- Supervised learning



Machine learning structure

- Unsupervised learning



Classification by Decision Tree Induction

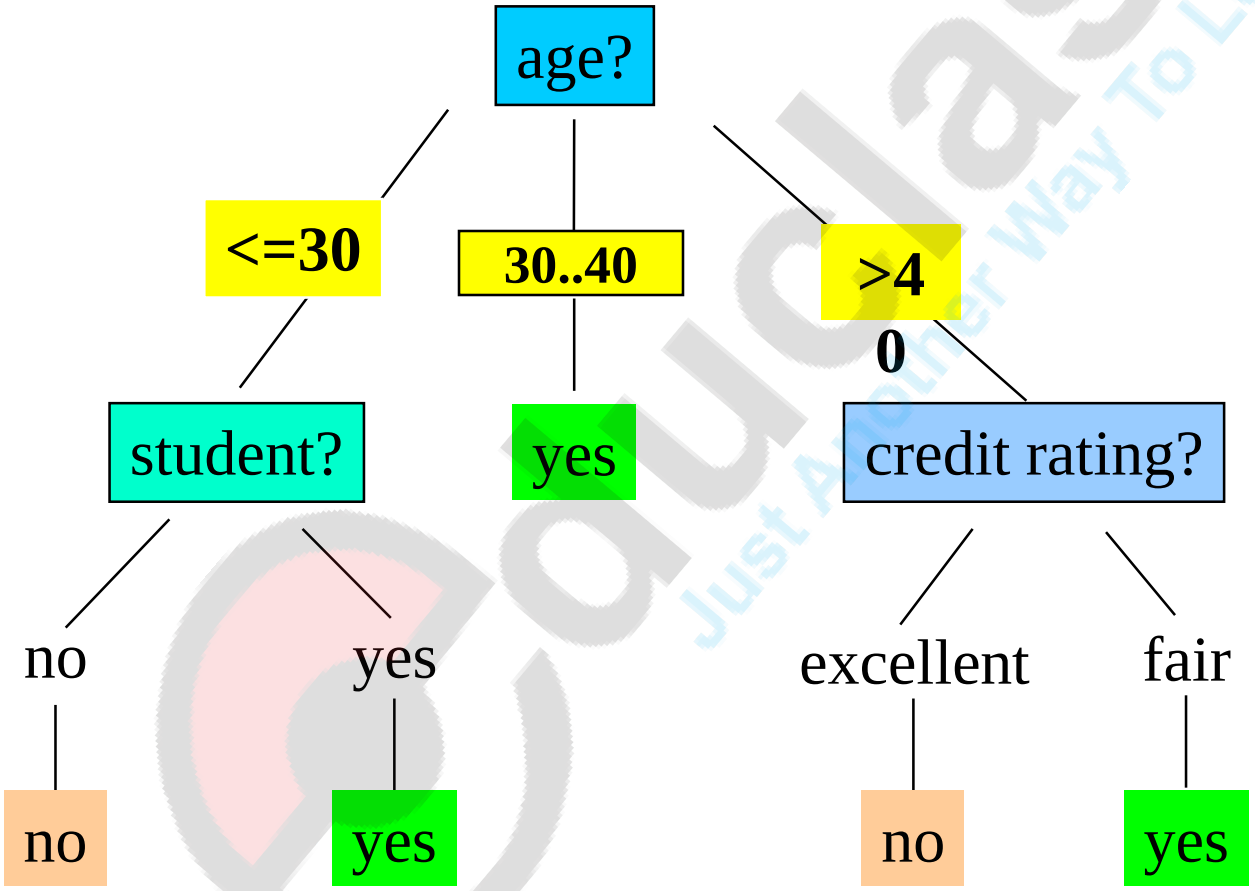
- **Decision tree learning** uses a **decision tree** (as a predictive model) to go from observations about an item (represented in the branches) to conclusions about the item's target value (represented in the leaves). It is one of the predictive modelling approaches used in statistics, data mining and **machine learning**.
  - A flow-chart-like tree structure
  - Internal node denotes a test on an attribute
  - Branch represents an outcome of the test
  - Leaf nodes represent class labels or class distribution
- Decision tree generation consists of two phases
  - Tree construction
    - At start, all the training examples are at the root
    - Partition examples recursively based on selected attributes
  - Tree pruning
    - Identify and remove branches that reflect noise or outliers
- Use of decision tree: Classifying an unknown sample
  - Test the attribute values of the sample against the decision tree

Training Dataset

This follows an example from Quinlan's ID3

age	income	student	credit_rating	buys_compter
<=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

Output: A Decision Tree for "*buys_computer*"



Algorithm for Decision Tree Induction

- Basic algorithm (a greedy algorithm)
  - Tree is constructed in a top-down recursive divide-and-conquer manner
  - At start, all the training examples are at the root
  - Attributes are categorical (if continuous-valued, they are discretized in advance)
  - Examples are partitioned recursively based on selected attributes
  - Test attributes are selected on the basis of a heuristic or statistical measure (e.g., information gain)
- Conditions for stopping partitioning
  - All samples for a given node belong to the same class
  - There are no remaining attributes for further partitioning – majority voting is employed for classifying the leaf
  - There are no samples left

Information Gain (ID3/C4.5)

- Select the attribute with the highest information gain
- Assume there are two classes, *P* and *N*
  - Let the set of examples *S* contain *p* elements of class *P* and *n* elements of class *N*
  - The amount of information, needed to decide if an arbitrary example in *S* belongs to *P* or *N* is defined as

$$I(p,n) = -\frac{p}{p+n}\log_2\frac{p}{p+n} - \frac{n}{p+n}\log_2\frac{n}{p+n}$$

Information Gain in Decision Tree Induction

- Assume that using attribute A a set *S* will be partitioned into sets $\{S_1, S_2, \ldots, S_v\}$
  - If S_i contains p_i examples of *P* and n_i examples of *N*, the entropy, or the expected information needed to classify objects in all subtrees S_i is

$$E(A) = \sum_{i=1}^{v} \frac{p_i + n_i}{p + n} I(p_i, n_i)$$

- The encoding information that would be gained by branching on *A*

$$Gain(A) = I(p, n) - E(A)$$

Attribute Selection by Information Gain Computation

- Class P: buys_computer = “yes”
- Class N: buys_computer = “no”
- I(p, n) = I(9, 5) =0.940
- Compute the entropy for *age*:

age	p_i	n_i	$I(p_i, n_i)$
<=30	2	3	0.971
30…40	4	0	0
>40	3	2	0.971

$$E(age) = \frac{5}{14} I(2,3) + \frac{4}{14} I(4,0) + \frac{5}{14} I(3,2) = 0.69$$

Hence

$$Gain(age) = I(p,n) - E(age)$$

Similarly

$$Gain(income) = 0.029$$
$$Gain(student) = 0.151$$
$$Gain(credit\_rating) = 0.048$$

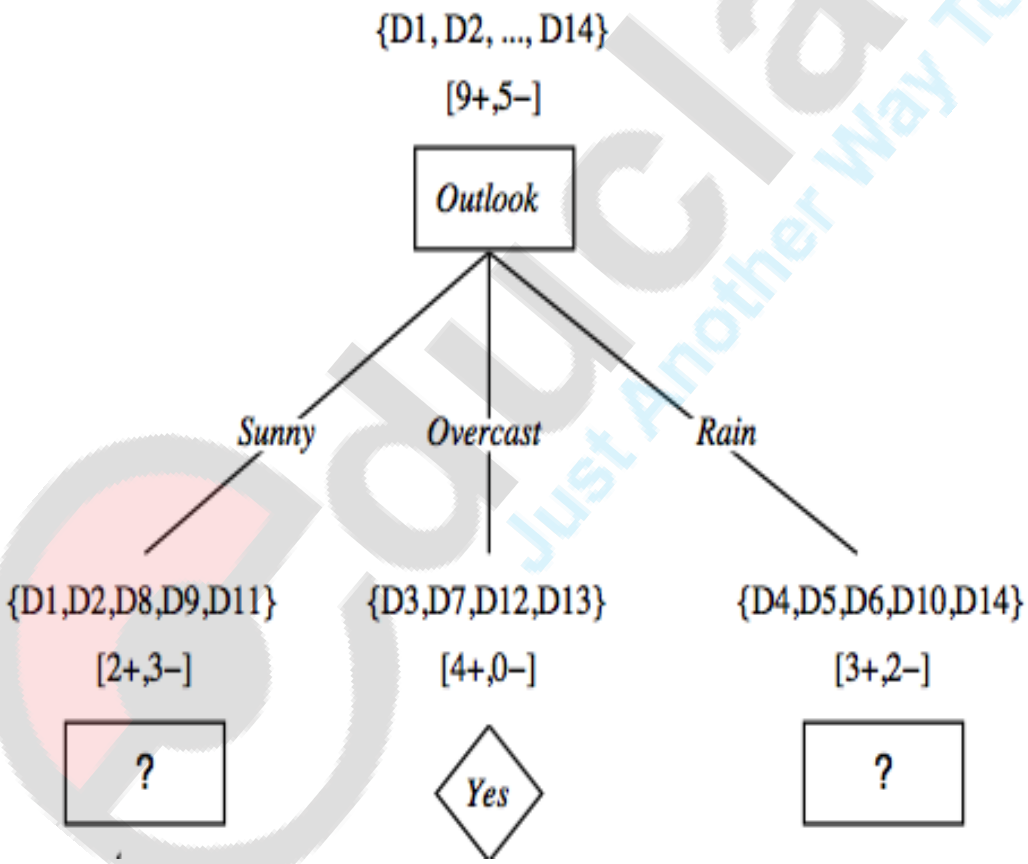
Example

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

First step: which attribute to test at the root?

- Which attribute should be tested at the root?
  - $Gain(S, Outlook) = 0.246$
  - $Gain(S, Humidity) = 0.151$
  - $Gain(S, Wind) = 0.084$
  - $Gain(S, Temperature) = 0.029$
- *Outlook* provides the best prediction for the target
- Lets grow the tree:
  - add to the tree a successor for each possible value of *Outlook*
  - partition the training samples according to the value of *Outlook*

After first step



Second step

- Working on *Outlook=Sunny* node:

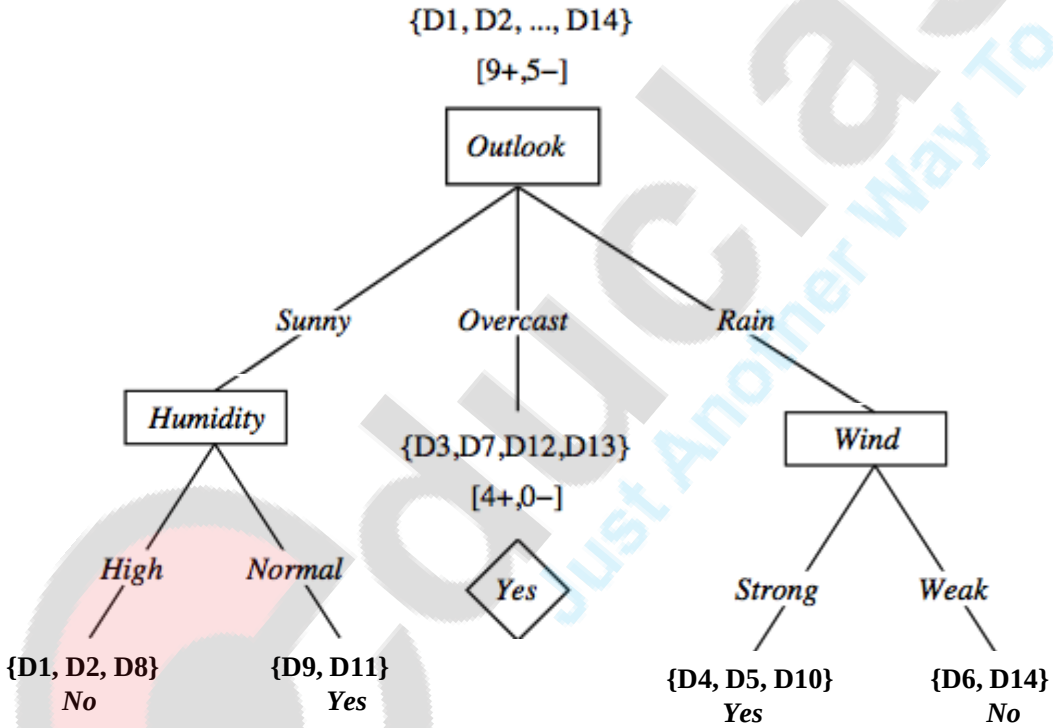
  $Gain(S_{Sunny}, Humidity) = 0.970 - 3/5 \times 0.0 - 2/5 \times 0.0 = 0.970$

  $Gain(S_{Sunny}, Wind) = 0.970 - 2/5 \times 1.0 - 3.5 \times 0.918 = 0\ .019$

  $Gain(S_{Sunny}, Temp.) = 0.970 - 2/5 \times 0.0 - 2/5 \times 1.0 - 1/5 \times 0.0 = 0.570$

- *Humidity* provides the best prediction for the target
- Lets grow the tree:
  - add to the tree a successor for each possible value of *Humidity*
  - partition the training samples according to the value of *Humidity*

Second and third steps



Extracting Classification Rules from Trees

- Represent the knowledge in the form of IF-THEN rules
- One rule is created for each path from the root to a leaf
- Each attribute-value pair along a path forms a conjunction
- The leaf node holds the class prediction
- Rules are easier for humans to understand
- Example

IF *age* = "<=30" AND *student* = "*no*" THEN *buys_computer* = "*no*"

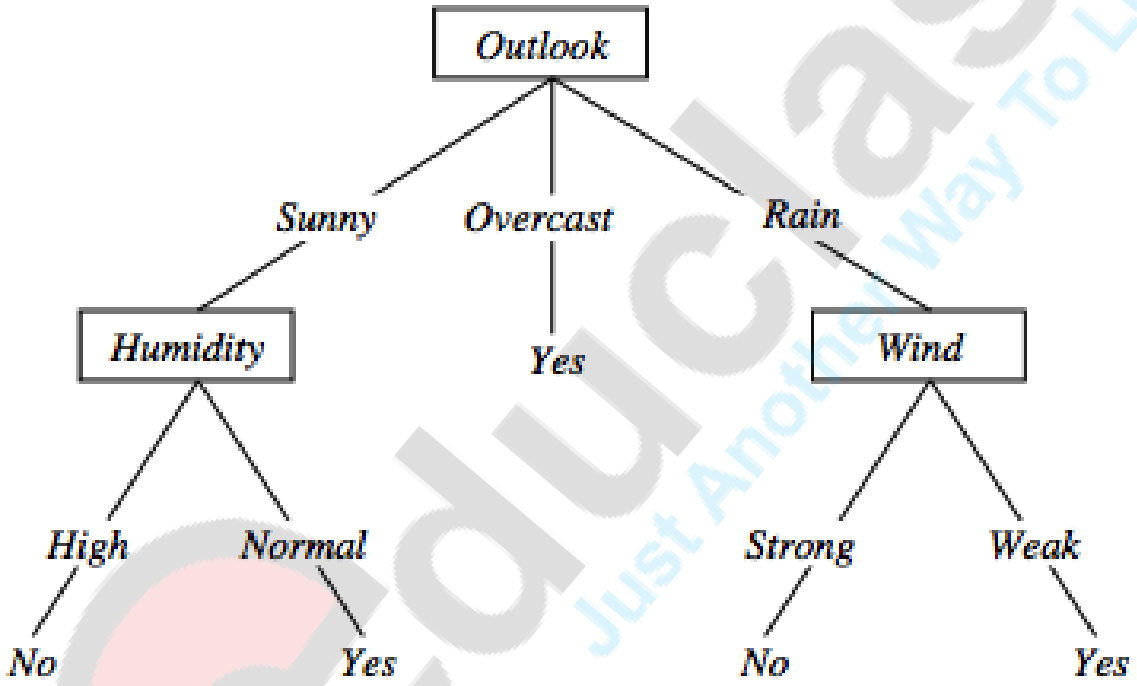
IF *age* = "<=30" AND *student* = "*yes*" THEN *buys_computer* = "*yes*"

IF *age* = "31…40" THEN *buys_computer* = "*yes*"

IF *age* = ">40" AND *credit_rating* = "*excellent*" THEN *buys_computer* = "*yes*"

IF *age* = ">40" AND *credit_rating* = "*fair*" THEN *buys_computer* = "*no*"

Converting to rules



$(Outlook=Sunny) \wedge (Humidity=High) \Rightarrow (PlayTennis=No)$

Avoid Overfitting in Classification

- The generated tree may overfit the training data
  - Too many branches, some may reflect anomalies due to noise or outliers
  - Result is in poor accuracy for unseen samples
- Two approaches to avoid overfitting
  - Prepruning: Halt tree construction early—do not split a node if this would result in the goodness measure falling below a threshold
    - Difficult to choose an appropriate threshold
  - Postpruning: Remove branches from a "fully grown" tree—get a sequence of progressively pruned trees
    - Use a set of data different from the training data to decide which is the "best pruned tree"

Overfitting: definition

- Building trees that "adapt too much" to the training examples may lead to "overfitting".
- Consider error of hypothesis h over
  - training data: $error_D(h)$ empirical error
  - entire distribution X of data: $error_X(h)$ expected error
- Hypothesis *h overfits* training data if there is an alternative hypothesis $h' \in H$ such that

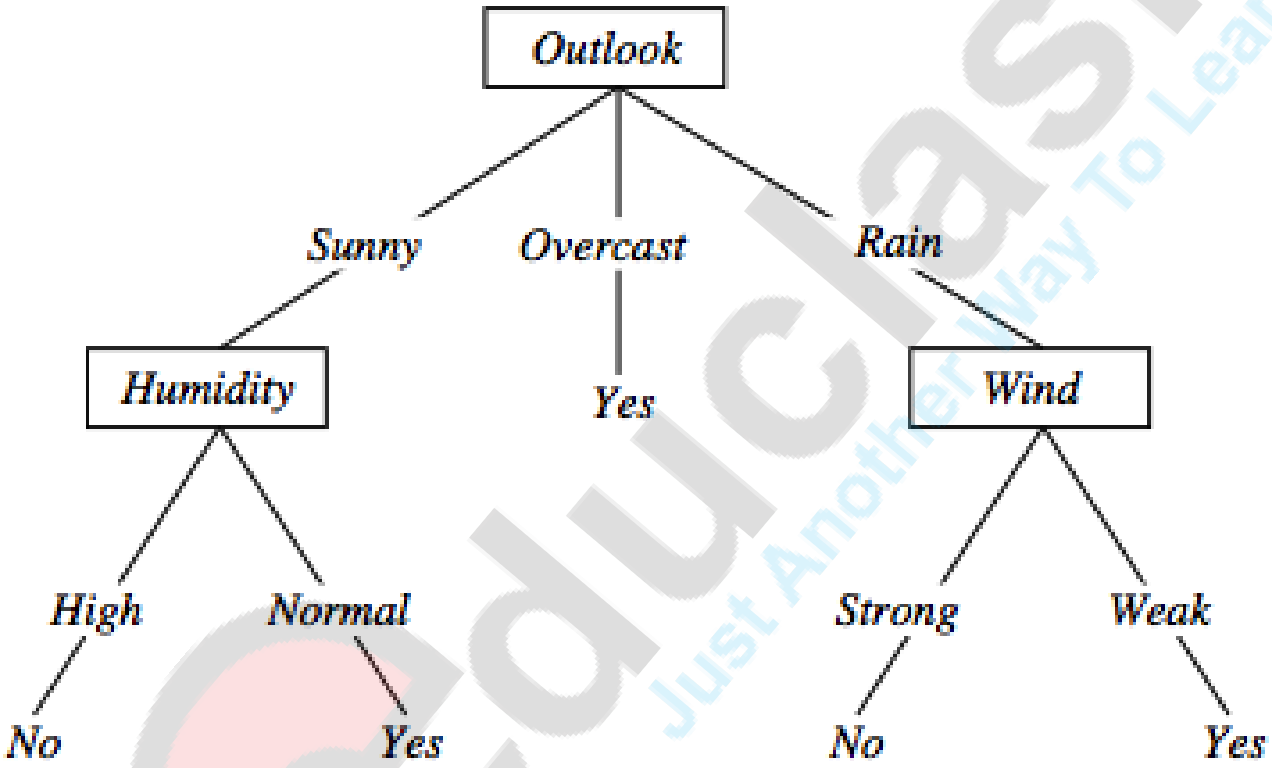
$$error_D(h) < error_D(h') \quad \text{and}$$
$$error_X(h') < error_X(h)$$

i.e. h' behaves better over unseen data

Example

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No
D15	Sunny	Hot	Normal	Strong	No

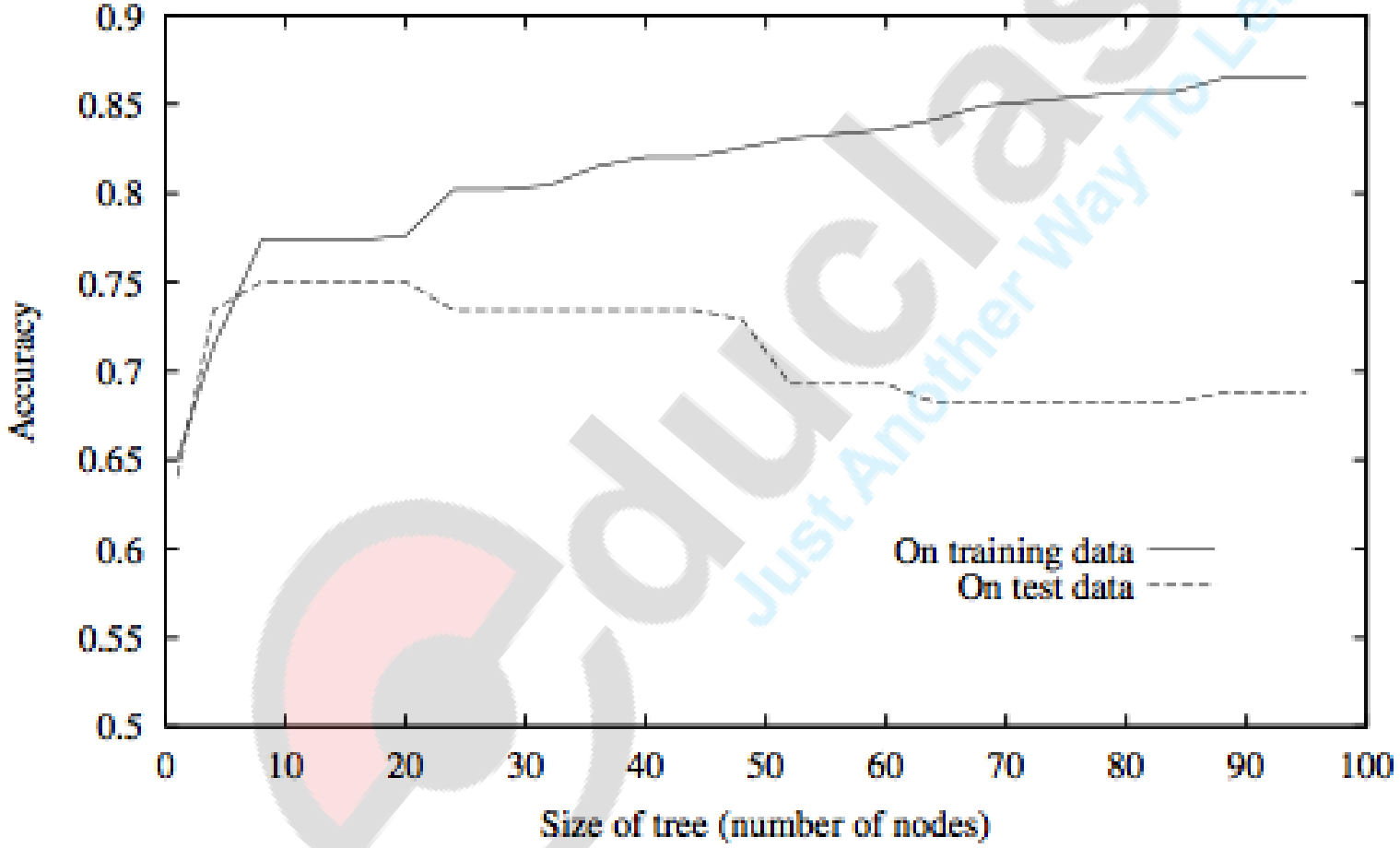
Overfitting in decision trees



⟨*Outlook=Sunny, Temp=Hot, Humidity=Normal, Wind=Strong, PlayTennis=No*⟩

New noisy example causes splitting of second leaf node.

Overfitting in decision tree learning



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

Bayesian Classification: Why?

- Probabilistic learning: Calculate explicit probabilities for hypothesis, among the most practical approaches to certain types of learning problems
- Incremental: Each training example can incrementally increase/decrease the probability that a hypothesis is correct. Prior knowledge can be combined with observed data.
- Probabilistic prediction: Predict multiple hypotheses, weighted by their probabilities

Bayesian Theorem

- Given training data *D, posteriori probability of a hypothesis h, P(h|D)* follows the Bayes theorem

$$P(h|D) = \frac{P(D|h)P(h)}{P(D)}$$

- Practical difficulty: require initial knowledge of many probabilities, significant computational cost

Training Set

Outlook	Temperature	Humidity	Windy	Class
sunny	hot	high	false	N
sunny	hot	high	true	N
overcast	hot	high	false	P
rain	mild	high	false	P
rain	cool	normal	false	P
rain	cool	normal	true	N
overcast	cool	normal	true	P
sunny	mild	high	false	N
sunny	cool	normal	false	P
rain	mild	normal	false	P
sunny	mild	normal	true	P
overcast	mild	high	true	P
overcast	hot	normal	false	P
rain	mild	high	true	N

Naive Bayesian Classifier (II)

- Given a training set, we can compute the probabilities

Outlook	P	N
sunny	2/9	3/5
overcast	4/9	0
rain	3/9	2/5
Tempreature		
hot	2/9	2/5
mild	4/9	2/5
cool	3/9	1/5

Humidity	P	N
high	3/9	4/5
normal	6/9	1/5
Windy		
true	3/9	3/5
false	6/9	2/5

Bayesian classification

- The classification problem may be formalized using a-posteriori probabilities:
- P(C|X) = prob. that the sample tuple $X=<x_1,\ldots,x_k>$ is of class C.

- E.g. P(class=N | outlook=sunny,windy=true,…)

- Idea: assign to sample X the class label C such that P(C|X) is maximal

Play-tennis example: estimating $P(x_i|C)$

Outlook	Temperature	Humidity	Windy	Class
sunny	hot	high	false	N
sunny	hot	high	true	N
overcast	hot	high	false	P
rain	mild	high	false	P
rain	cool	normal	false	P
rain	cool	normal	true	N
overcast	cool	normal	true	P
sunny	mild	high	false	N
sunny	cool	normal	false	P
rain	mild	normal	false	P
sunny	mild	normal	true	P
overcast	mild	high	true	P
overcast	hot	normal	false	P
rain	mild	high	true	N

P(p) = 9/14
P(n) = 5/14

outlook	
P(sunny|p) = 2/9	**P(sunny|n) = 3/5**
P(overcast|p) = 4/9	**P(overcast|n) = 0**
P(rain|p) = 3/9	**P(rain|n) = 2/5**
temperature	
P(hot|p) = 2/9	**P(hot|n) = 2/5**
P(mild|p) = 4/9	**P(mild|n) = 2/5**
P(cool|p) = 3/9	**P(cool|n) = 1/5**
humidity	
P(high|p) = 3/9	**P(high|n) = 4/5**
P(normal|p) = 6/9	**P(normal|n) = 2/5**
windy	
P(true|p) = 3/9	**P(true|n) = 3/5**
P(false|p) = 6/9	**P(false|n) = 2/5**

Play-tennis example: classifying X

- An unseen sample X = <rain, hot, high, false>

- $P(X|p)\cdot P(p) =$
  $P(rain|p)\cdot P(hot|p)\cdot P(high|p)\cdot P(false|p)\cdot P(p) =$
  $3/9\cdot 2/9\cdot 3/9\cdot 6/9\cdot 9/14 = 0.010582$
- $P(X|n)\cdot P(n) =$
  $P(rain|n)\cdot P(hot|n)\cdot P(high|n)\cdot P(false|n)\cdot P(n) =$
  $2/5\cdot 2/5\cdot 4/5\cdot 2/5\cdot 5/14 = 0.018286$

- Sample X is classified in class n (don't play)

Car Theft Example

Example No.	Color	Type	Origin	Stolen?
1	Red	Sports	Domestic	Yes
2	Red	Sports	Domestic	No
3	Red	Sports	Domestic	Yes
4	Yellow	Sports	Domestic	No
5	Yellow	Sports	Imported	Yes
6	Yellow	SUV	Imported	No
7	Yellow	SUV	Imported	Yes
8	Yellow	SUV	Domestic	No
9	Red	SUV	Imported	No
10	Red	Sports	Imported	Yes

Classify (Red Domestic SUV)

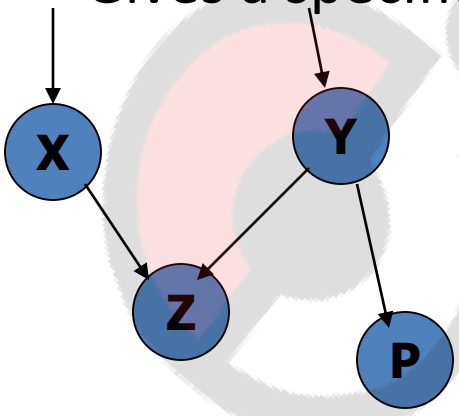
Bayesian Belief Network

- Bayesian Belief networks describe conditional independence among **subsets** of variables

- allows combining prior knowledge about (in)dependencies among variables with observed training data

Bayesian Networks

- Bayesian belief network allows a *subset* of the variables conditionally independent
- A graphical model of causal relationships
  - Represents dependency among the variables
  - Gives a specification of joint probability distribution



- ❑Nodes: random variables
- ❑Links: dependency
- ❑X,Y are the parents of Z, and Y is the parent of P
- ❑No dependency between Z and P
- ❑Has no loops or cycles

Conditional Independence

– Once we know that the patient has cavity we do not expect the probability of the probe catching to depend on the presence of toothache

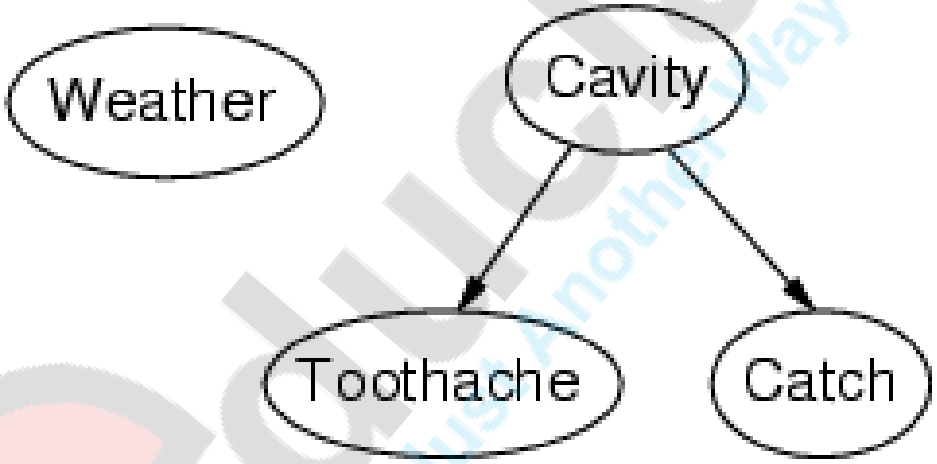
$$P(catch \mid cavity \wedge toothache) = P(catch \mid cavity)$$
$$P(toothache \mid cavity \wedge catch) = P(toothache \mid cavity)$$

– Independence between a and b

$$P(a \mid b) = P(a)$$
$$P(b \mid a) = P(b)$$

Example

- Topology of network encodes conditional independence assertions:



- *Weather* is independent of the other variables
- *Toothache* and *Catch* are conditionally independent given *Cavity*

Example

- I'm at work, neighbor John calls to say my alarm is ringing, but neighbor Mary doesn't call. Sometimes it's set off by minor earthquakes. Is there a burglar?

- Variables: *Burglary*, *Earthquake*, *Alarm*, *JohnCalls*, *MaryCalls*

- Network topology reflects "causal" knowledge:

  - A burglar can set the alarm off
  - An earthquake can set the alarm off
  - The alarm can cause Mary to call
  - The alarm can cause John to call

Belief Networks