University of Colorado
Boulder

CSCI 4502/5502
Data Mining

Fall 2020
Lecture 14 (Oct 15)

Reminder/Announcement

- Homework 4: due at 9:30am, Thursday, Oct 15
- Homework 5: due at 9:30am, Thursday, Oct 22
- Midterm exam: Thursday, Oct 29
  - availability survey: Tuesday, Oct 20
  - midterm review: Thursday, Oct 22
  - practice exam: Tuesday, Oct 27

Review: Chapter 8: Classification

- Basic concepts
- Decision tree induction
- Bayesian classification
- Rule-based classification
- Model evaluation and selection
- Improve classification accuracy
- Summary

University of Colorado
Boulder

Chapter 9: Advanced Classification Methods

Chap 9: Advanced Classification

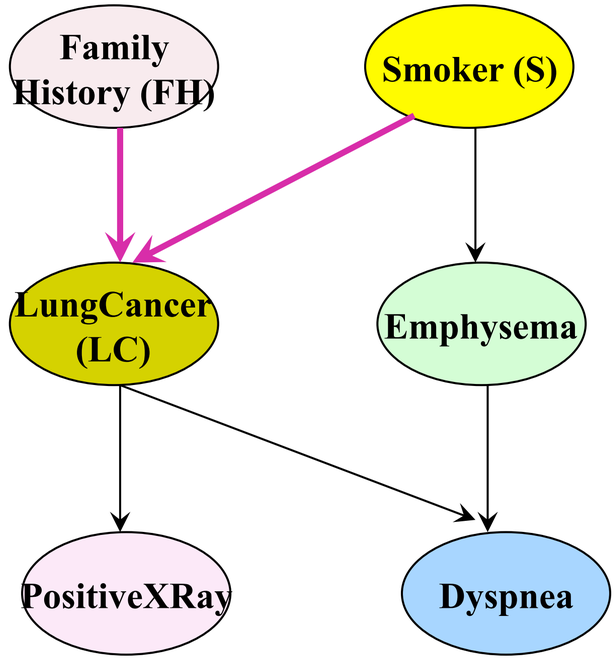
- Bayesian belief networks
- Backpropagation
- Support vector machines
- Lazy learning (or learning from your neighbors)
- Additional topics regarding classification
- Summary

Naïve Bayesian Classifier

- Advantage
  - easy to compute, good results in most cases
- Disadvantage
  - assumption: class conditional independence
  - dependencies exist in practice: e.g., hospital patients: age, family history, fever, cough, lung cancer, diabetes, etc.

Bayesian Belief Networks

- Subset of variables conditionally independent
- Causal model: a directed, acyclic graph
  - node, link, parent, CPTs



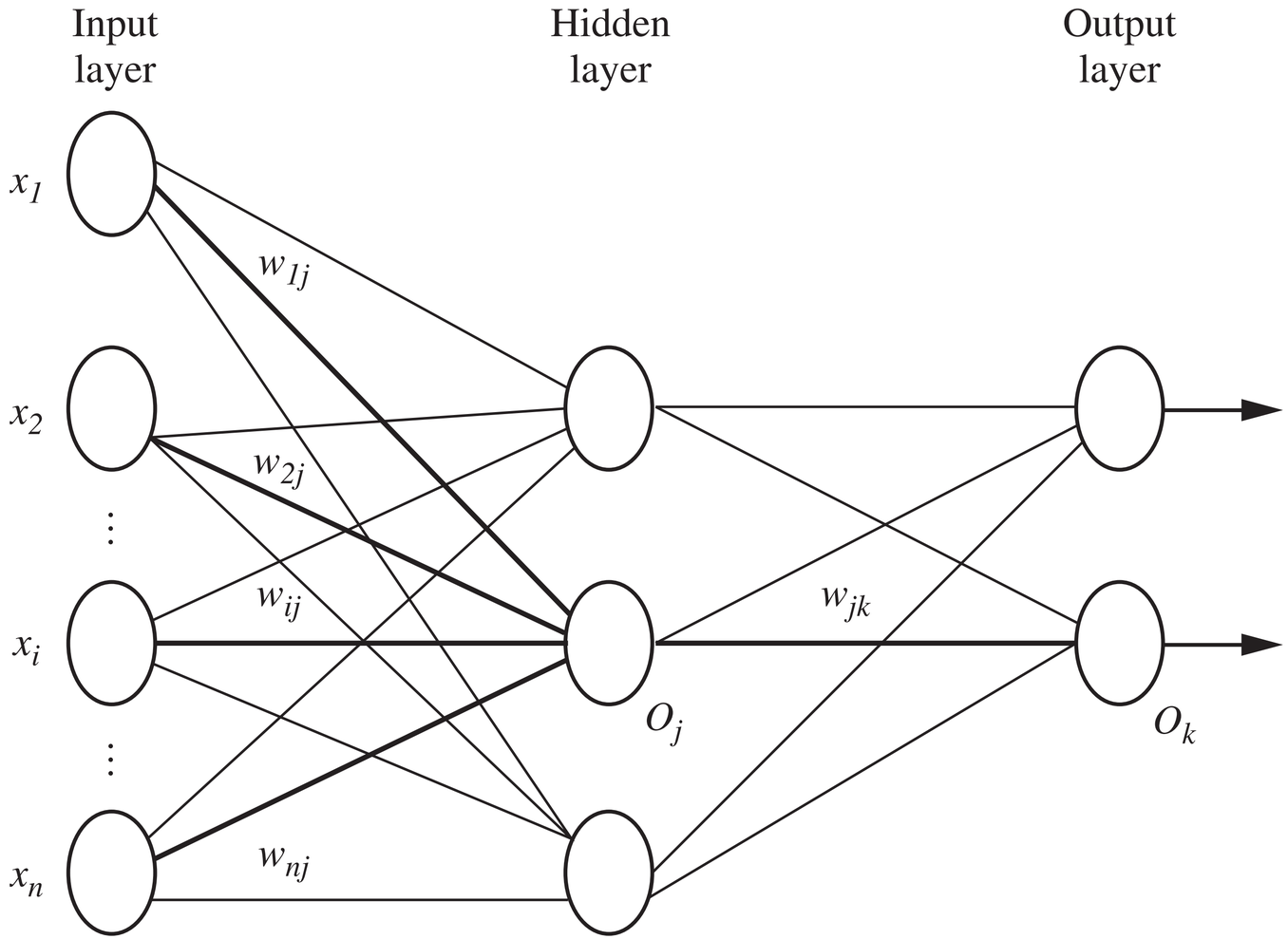
CPT: Conditional Probability Table for variable LungCancer:

	(FH, S)	(FH, ~S)	(~FH, S)	(~FH, ~S)
LC	0.8	0.5	0.7	0.1
~LC	0.2	0.5	0.3	0.9

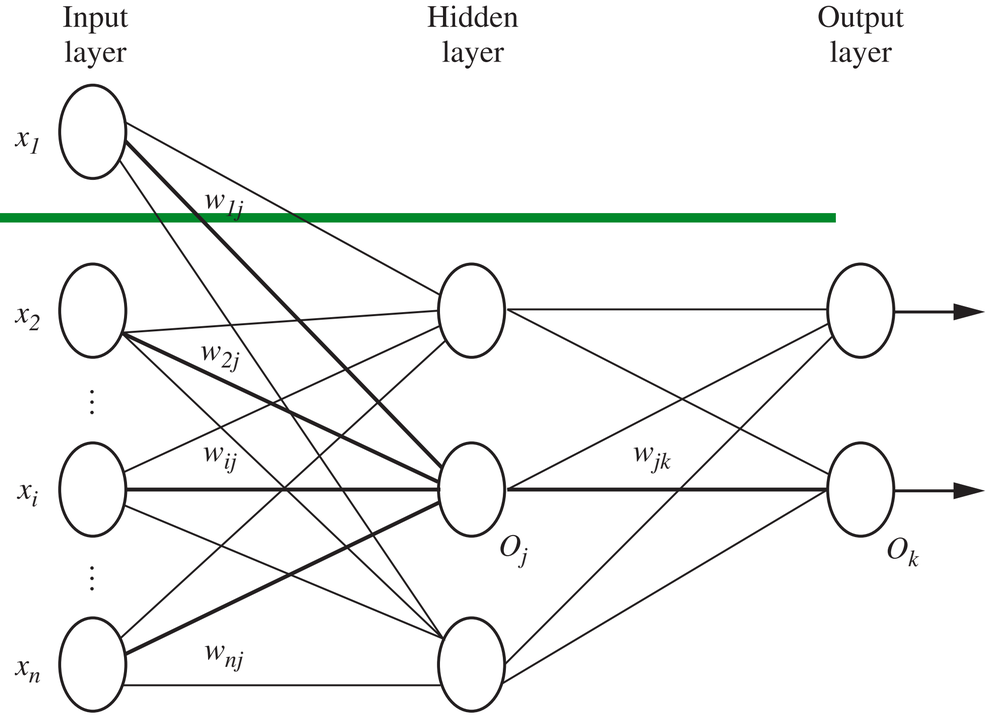
$$P(x_1, \cdots, x_n) = \prod_{i=1}^{n} P(x_i | Parents(Y_i))$$

Neural Network

- Connected input/out units
- Weighted connections
- Multi-layer
- Feed-forward
- Fully connected
- Backpropagation
- Adjust weights



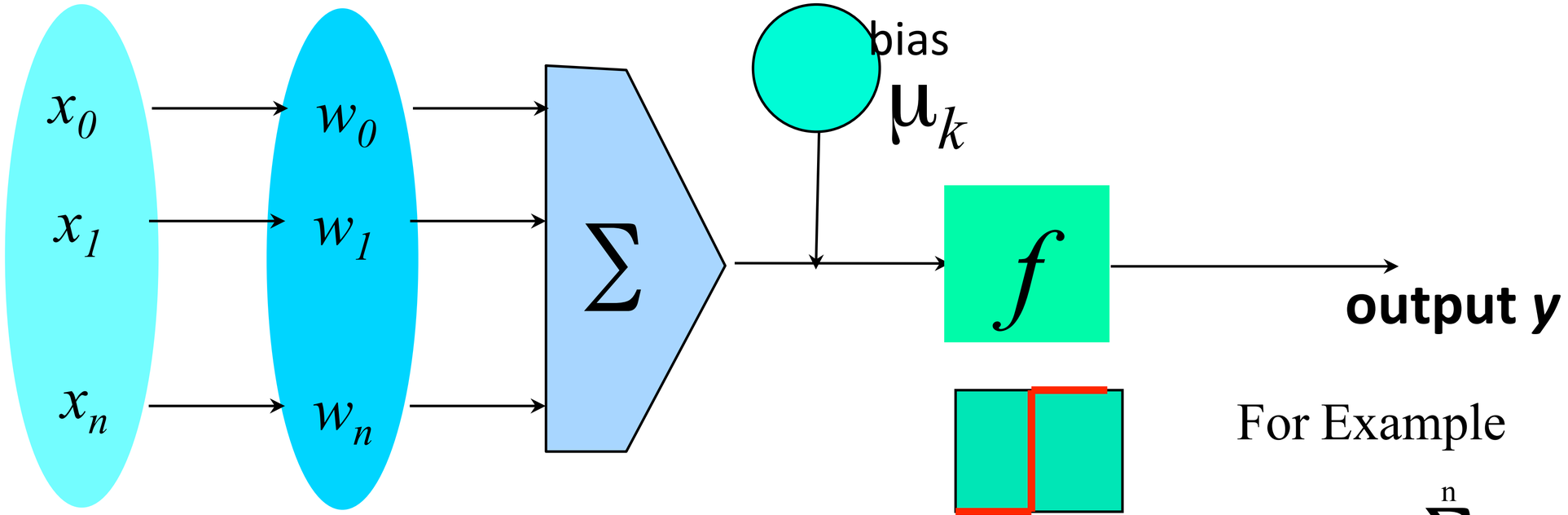
Network Topology



- # hidden layers, # units in each layer
- Input values
  - continuous: normalize to [0.0, 1.0]
  - discrete: one unit per attribute value, initially 0
- Output: 1 unit per class if more than two classes
- Trial-and-error: different topology, different initial weights

Neuron

- A hidden/output layer unit



Input vector x **weight vector w** **weighted sum** **Activation function**

For Example

$$y = \text{sign}(\sum_{i=0}^{n} w_i x_i - \mu_k)$$

Backpropagation

- Initialize weights, biases: small random numbers
- Propagate the inputs forward
- Backpropagate the error
- Terminating condition

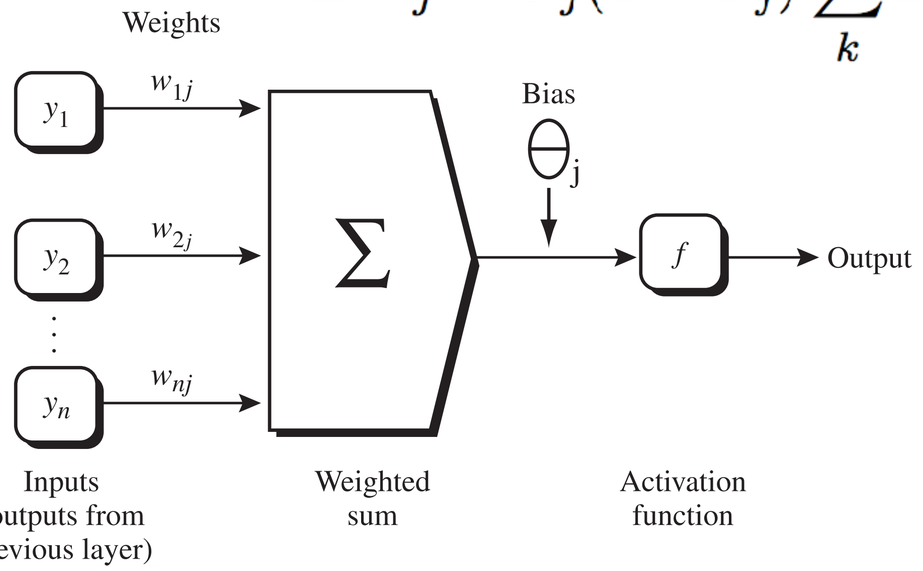
$$I_j = \sum_i w_{ij} O_i + \theta_j \qquad O_j = \frac{1}{1 + e^{-I_j}}$$

$$Err_j = O_j(1 - O_j)(T_j - O_j)$$

$$Err_j = O_j(1 - O_j) \sum_k Err_k W_{jk}$$

$$w_{ij} = w_{ij} + (l) Err_j O_i$$

$$\theta_j = \theta_j + (l) Err_j$$



Neural Network as Classifier

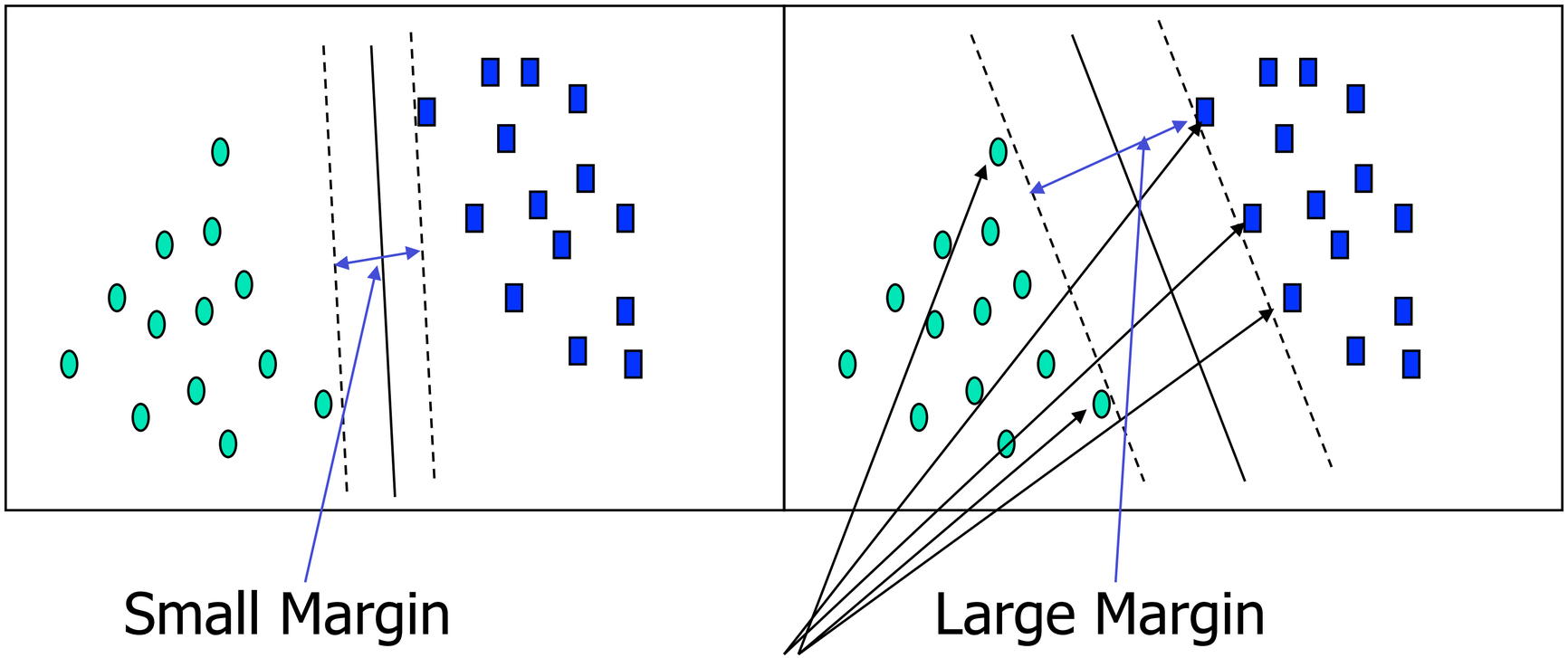
- Weakness
  - long training time; parameters determined empirically; poor interpretability
- Strength
  - high tolerance to noisy data; can classify untrained patterns; well-suited for continuous-valued inputs & outputs; success on a wide array of real-world data; inherently parallel; rule extraction

Chap 9: Advanced Classification

- Bayesian belief networks
- Backpropagation
- Support vector machines
- Lazy learning (or learning from your neighbors)
- Additional topics regarding classification
- Summary

SVM Example

- Linearly separable data

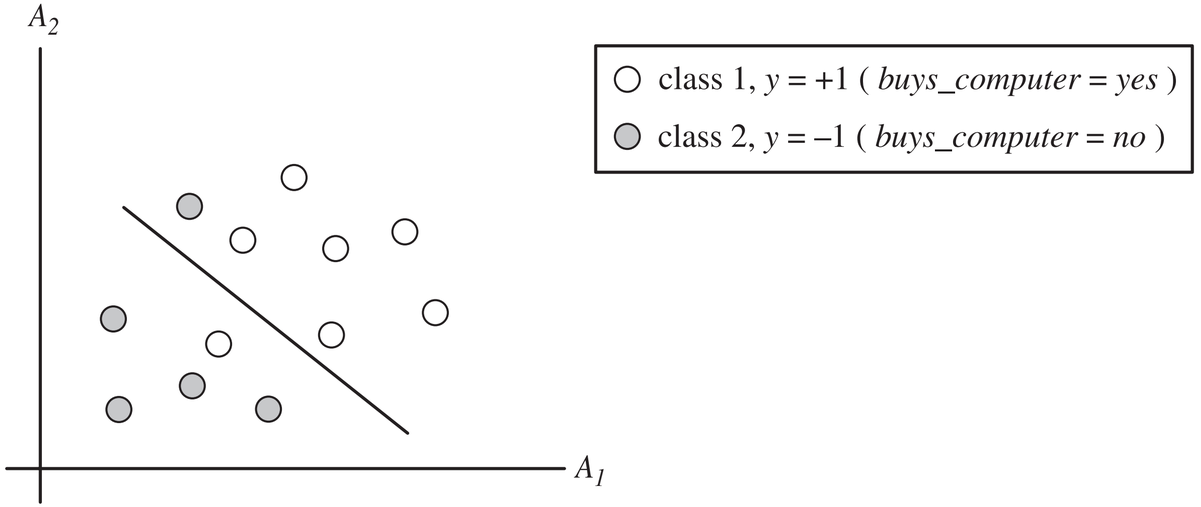


SVM: Linearly Separable Data

- D: $(X_1, y_1), ..., (X_{|D|}, y_{|D|})$, X_i is n-dimensional
- A separating hyperplane
  - $W * X + b = 0$
- Find maximum marginal hyperplane (MMH)
- Hyperplanes H_1 and H_2
  - sides of the margin
- Support vectors
  - training tuples fall on H_1 H_2

SVM: Linearly Inseparable

- Transform data into a higher dimension
- Search for optimal linear separating hyperplane in the new space
- Computing dot product on the transformed data mathematically equivalent to applying a kernel function to original data
  - $K(X_i, X_j)$
  - $= \varnothing(X_i) \cdot \varnothing(X_j)$



Support Vector Machines

- Classification for both linear & nonlinear data
- Transforms data to higher dimension using nonlinear mapping
- Searches for optimal linear separation hyperplane in the new dimension
- SVM finds this hyperplane using support vectors ("essential" training tuples) and margins (defined by the support vectors)

Chap 9: Advanced Classification

- Bayesian belief networks
- Backpropagation
- Support vector machines
- Lazy learning (or learning from your neighbors)
- Additional topics regarding classification
- Summary

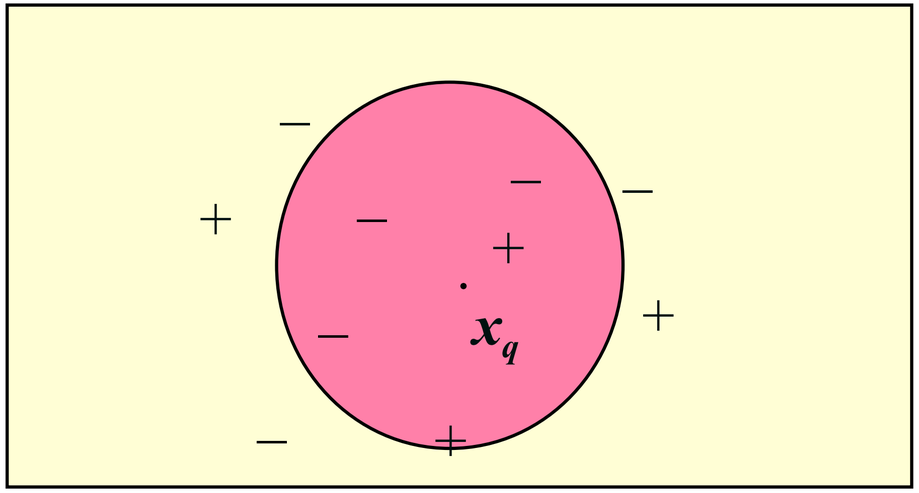
Lazy vs. Eager Learning

- Eager learning: constructs classification model on training data before receiving test data
- Lazy learning: stores training data and delays processing until a new instance must be classified

- Lazy: less time in training, more time in predicting
- Lazy: require efficient storage techniques
- Lazy: richer hypothesis space using many local linear functions for implicit global approximation

k-Nearest-Neighbor Classifiers

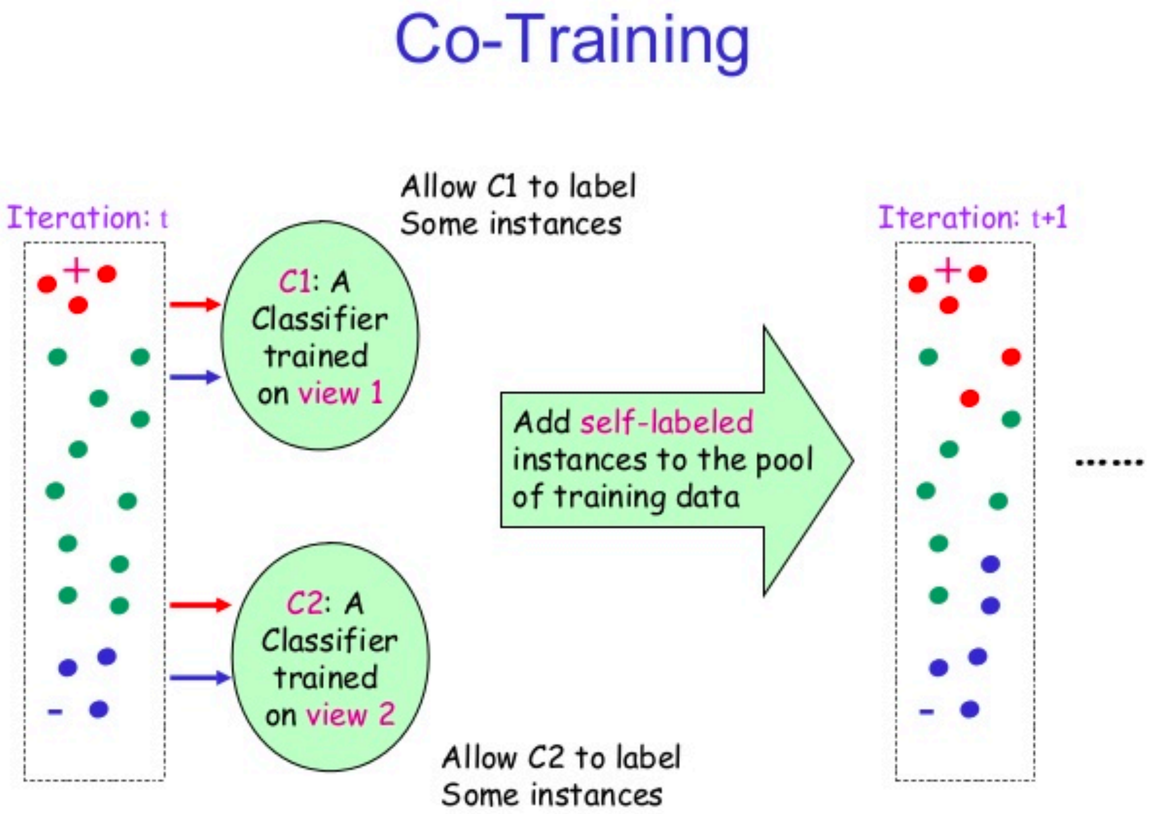
- Find k-nearest-neighbors of a test tuple
- Discrete-valued: most common values
- Continuous-valued: average of k values
- Give greater weight to closer neighbors
- Indexing for nearest-neighbor search



Chap 9: Advanced Classification

- Bayesian belief networks
- Backpropagation
- Support vector machines
- Lazy learning (or learning from your neighbors)
- Additional topics regarding classification
- Summary

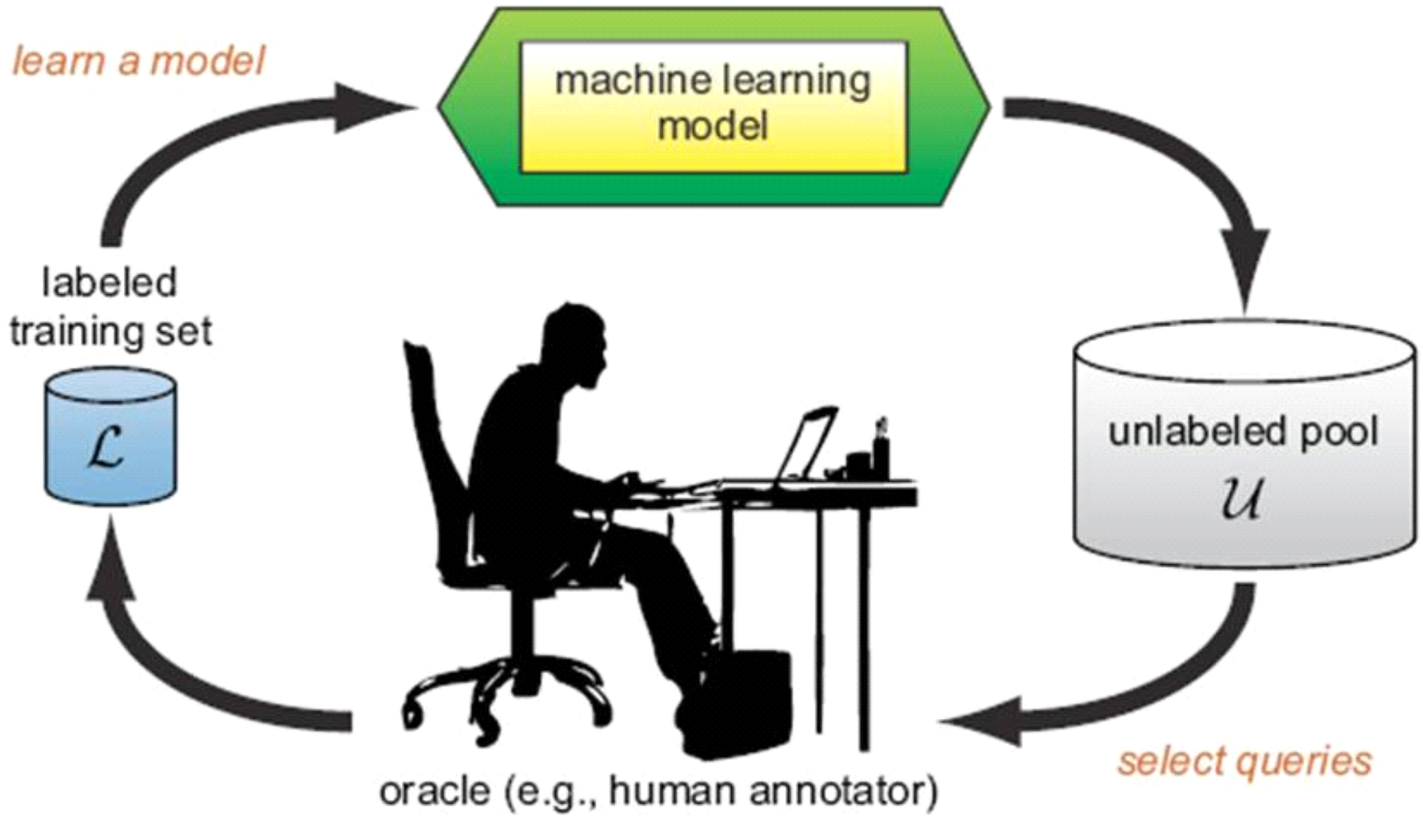
- Multiclass classification
  - one-versus-all (OVA)
  - all-versus-all (AVA)
- Semi-supervised training
  - self-training
  - cotraining



https://image.slidesharecdn.com/semisupervised-learning563/95/semisupervised-learning-25-728.jpg?cb=1272280467

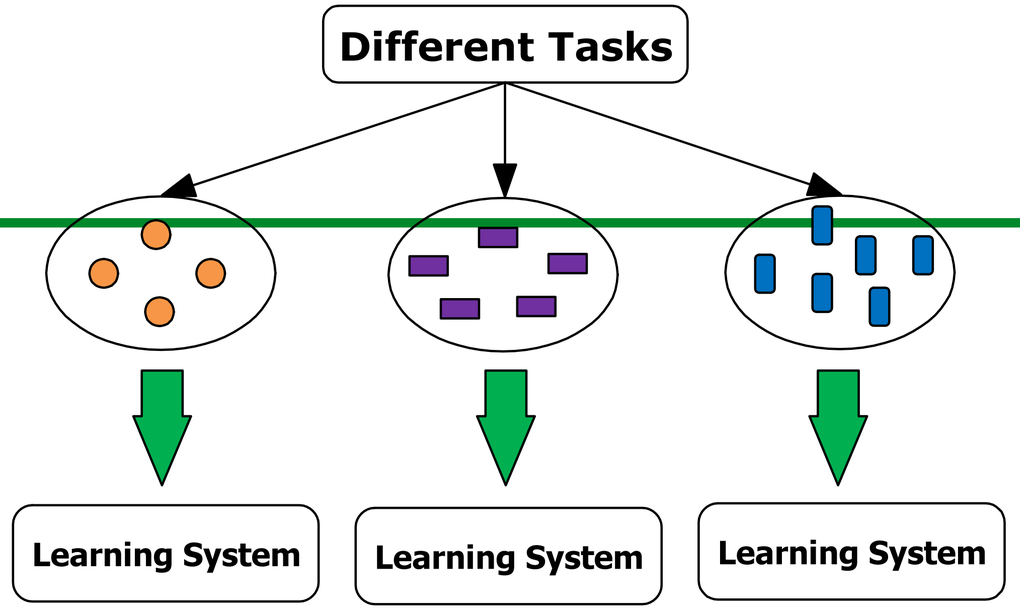
Active Learning

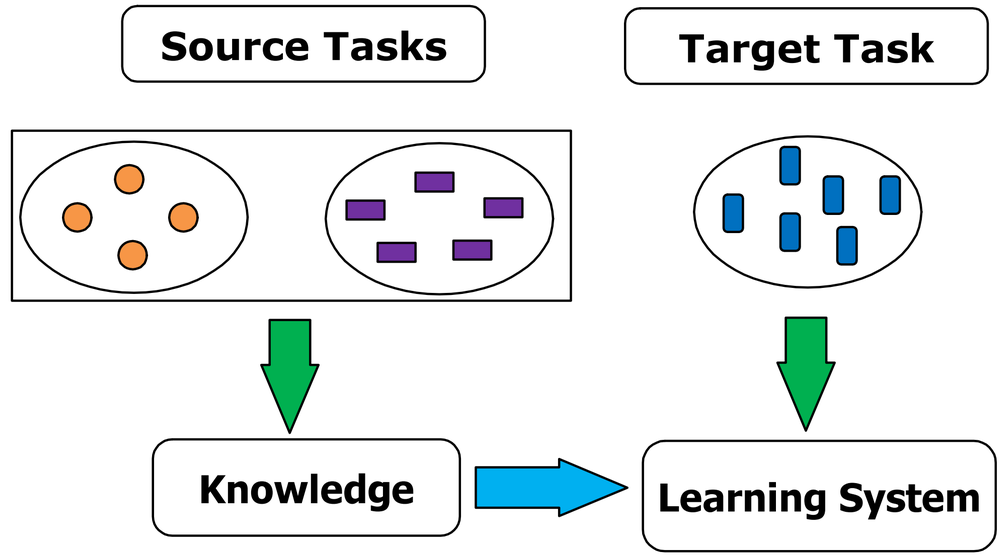
- Achieve high accuracy using as few labeled instances as possible
- Uncertainty sampling, vision space



Transfer Learning

- Utilize knowledge learned in source tasks for target task
- E.g., sentiment classification: camera reviews => TV reviews
- E.g., cars => trucks





Chap 9: Advanced Classification

- Bayesian belief networks
- Backpropagation
- Support vector machines
- Lazy learning (or learning from your neighbors)
- Additional topics regarding classification
- Summary

University of Colorado
Boulder

Chapter 10:
Cluster Analysis

What is Cluster Analysis?

- Cluster: a collection of data objects
  - similar to one another within a cluster
  - dissimilar to objects in other clusters
- Cluster analysis
  - group similar data objects into clusters
  - similarity measure & clustering algorithm
- Unsupervised learning
  - no predefined classes

Requirements of Clustering

- Scalability
- Different types of attributes
- Clusters with arbitrary shape
- Minimal domain knowledge for parameters
- Noisy data
- Incremental, insensitive to input order
- High dimensionality
- Constraint-based clustering
- Interpretability and usability

Major Clustering Methods (1)

- Partitioning methods
  - construct k partitions, iterative relocation
- Hierarchical methods
  - hierarchical decomposition, split/ merge
- Density-based methods
  - connectivity and density functions
- Grid-based methods
  - quantize into cells, multi-granularity grid

Major Clustering Methods (2)

- Model-based methods
  - hypothesized cluster model, best fit
- Clustering high-dimensional data
  - subspace clustering
- frequent-pattern-based clustering
- Constraint-based clustering
  - user-specified or application-oriented constraints

Chapter 10: Cluster Analysis

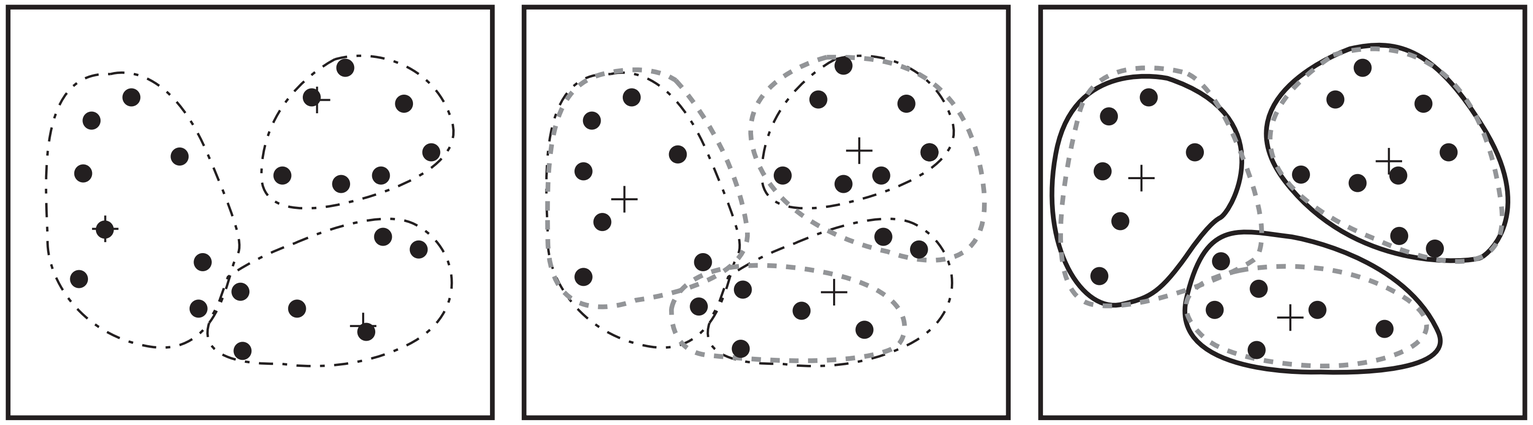
- Basic concepts
- Partitioning methods
- Hierarchical methods
- Density-based methods
- Grid-based methods
- Evaluation of clustering
- Summary

Partitioning Methods

- Given a data set D of n objects
- Given k, find a partition of k clusters that optimizes the chosen partition criterion
- Global optimal: enumerate all partitions
- Heuristic methods
  - k-means: cluster represented by mean (centroid)
  - k-medoids: cluster represented by medoid (object closest to centroid)

k-Means Clustering (1)

- Partition objects into k nonempty clusters
- Compute mean (centroid) of each cluster
- Assign each object to closest centroid
- Repeat till no more assignment changes



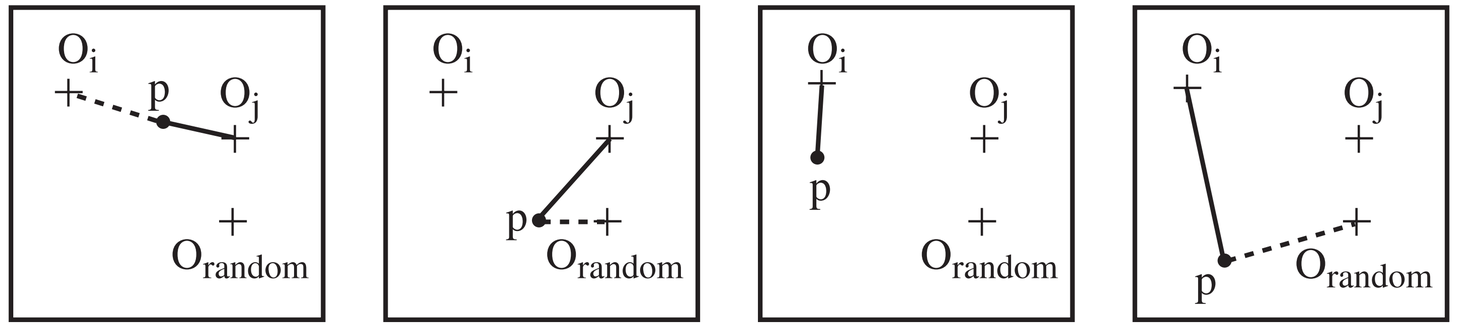
(a) (b) (c)

k-Means Clustering (2)

- Relatively efficient: O(nkt)
  - n: #objects, k: #clusters, t: #iterations
- Often terminates at local optimal
- Applicable only when centroid is defined
- Need to specify k in advance
- Not suitable for discovering clusters with non-convex shapes
- Sensitive to noise and outliers

k-Medoids Clustering (1)

- k-means method is sensitive to outliers
  - substantially distort the distribution of data
- k-medoids: find representative objects (medoids)



k-Medoids Clustering (2)

- PAM (Partitioning Around Medoids)
  - starts from an initial set of medoids
  - iteratively replace a medoid w/ a non-medoid if it reduces the total distance
  - effective for small data sets, does not scale, $O(k(n-k)^2)$ for each iteration
- CLARA: apply PAM on multiple sampled sets
- CLARANS: use randomized sample to search for neighboring solutions