Bioinformatics

Clusters and networks

Martin Saturka

http://www.bioplexity.org/lectures/

EBI version 0.4

Learning on profiles

Supervised and unsupervised methods on expression data.

- approximation, clustering, classification, inference.
- crisp and fuzzy relation models. graphical models.

Main topics

- vector approaches
  - SVM, ANN, kernels
  - classification
- data organization
  - clustering technics
  - map generation
- regulatory systems
  - Bayesian networks
  - algebraic description

Inference types

- reasoning
  - logical
    - standard logical rules based reasoning
  - statistical
    - frequent co-occurence based reasoning

- deduction (logic, recursion)
  - $A, A \rightarrow B \vdash B$
- induction (frequentist statistics)
  - many $A, B \vdash A \sim B$
  - few $A, \neg B \vdash A \rightarrow B$
- abduction (Bayesian statistics)
  - $A_1 \rightarrow B, ..., A_n \rightarrow B, B \vdash A_i$

Machine learning methods

- supervised learning methods
  with known correct outputs on training data
  - approximation
    - measured data to (continuous) output function
      growth rate $\longrightarrow$ nutrition supply regulation
  - classification
    - measured data to discrete output function
      expression profiles $\longrightarrow$ illness **diagnosis**
  - regression
    - continuous measured data (cor)relations
      a gene expression magnitude $\longrightarrow$ growth rate

- unsupervised learning methods
  without known desired outputs on used data
  - data granulation
    - internal data organization and distribution
  - data visualization
    - overall outer view onto the internal data

MLE

- maximal likelihood estimation
  - $L(y \mid X) = \Pr(X \mid Y = y)$
  - conditional probability as a function of the unkown condition with known outcome, a reverse view on probability

- Bernoulli trials example

$$L(\theta \mid H = 11, T = 10) \equiv \Pr(H = 11, T = 10 \mid p = \theta) = \binom{21}{11}\theta^{11}(1-\theta)^{10}$$
$$0 = \partial/\partial\theta\, L(\theta \mid H, T) = \binom{21}{11}\theta^{10}(1-\theta)^{9}(11 - 21\theta) \rightarrow \theta = 11/21$$

- used when without a better model
  - maximizations inside dynamic programming technics $\surd$
  - variance estimation leads to biased sample variance $\times$

Regression

- linear regression
  - least squares
    - for homoskedastic distributions
    - sample mean the best estimation
  - least absolute deviations
    - robust version
    - sample median a safe estimation

$$\min_{\bar{x}\in R}\left(\sum_i |x_i-\bar{x}|^2\right) \quad \Big| \quad \min_{\bar{x}\in R}\left(\sum_i |x_i-\bar{x}|\right)$$

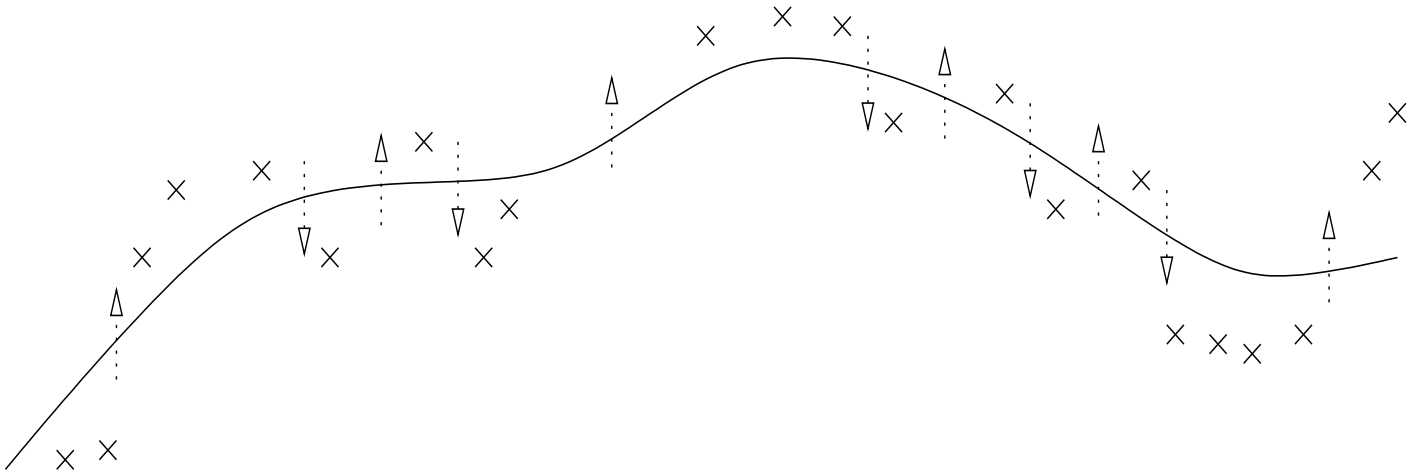
$$\Downarrow \qquad\qquad \Big| \qquad \Downarrow$$

$$\left(\sum_i |x_i-\bar{x}|^2\right)' = 0 \quad \Big| \quad \left(\sum_i |x_i-\bar{x}|\right)' = 0$$

$$\longrightarrow \text{ arithmetic mean} \quad \Big| \quad \longrightarrow \text{ median}$$

$$\bar{x} = \sum_i x_i/n \quad \Big| \quad \#x_i : x_i < \bar{x} = \#x_i : x_i > \bar{x}$$

Parametrization



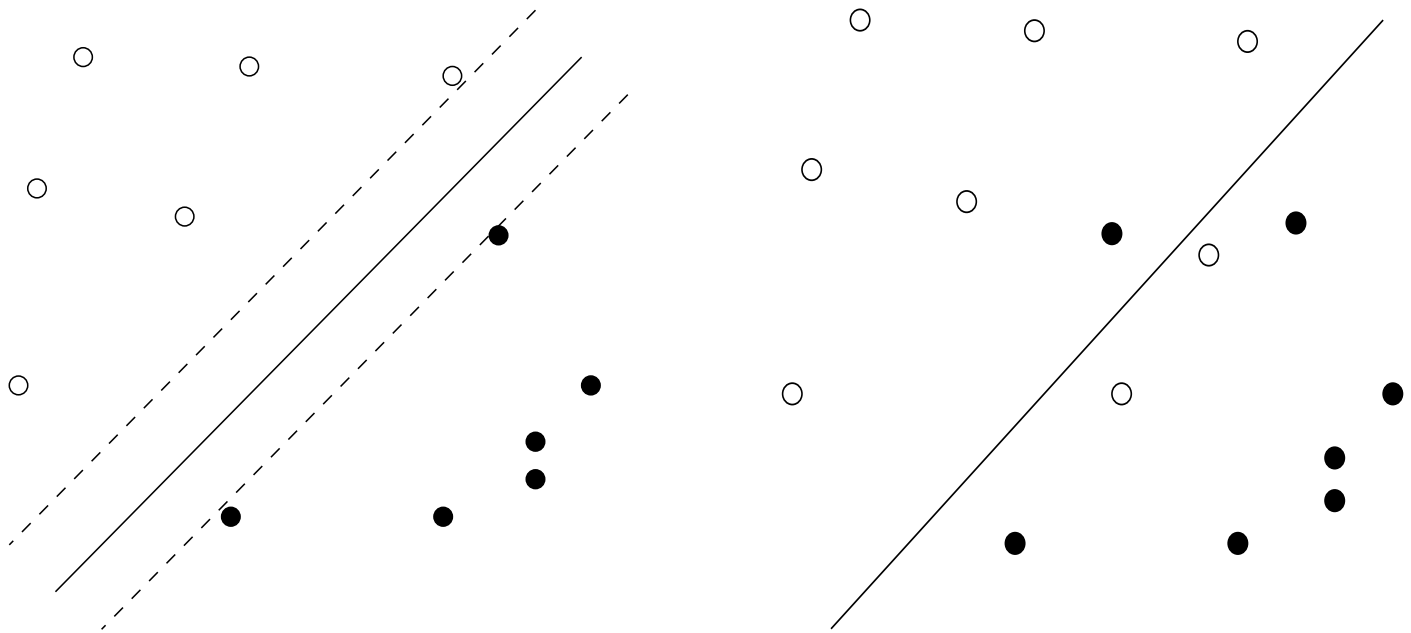
- parametrized curve crossing
  - assumption: equal amounts of above / below points
  - the same probabilities to cross / not to cross the curve
  - cross count distribution approaches normal distribution
  - over/under-fitting if not in $(N-1)/2 \pm \sqrt{(N-1)/2}$

Distinctions

empirical risk minimization for discrimination

- metamethodology
  - boosting
    - increasing weights of wrong result training cases
  - probably approximately correct learning
    - to achieve high probabilities to make convenient predictions
- particular methods
  - support vector machines
  - artificial neural networks
  - case-based reasoning
  - nearest neighbour algorithm
  - (naive) bayes classifier
  - decision trees
  - random forests

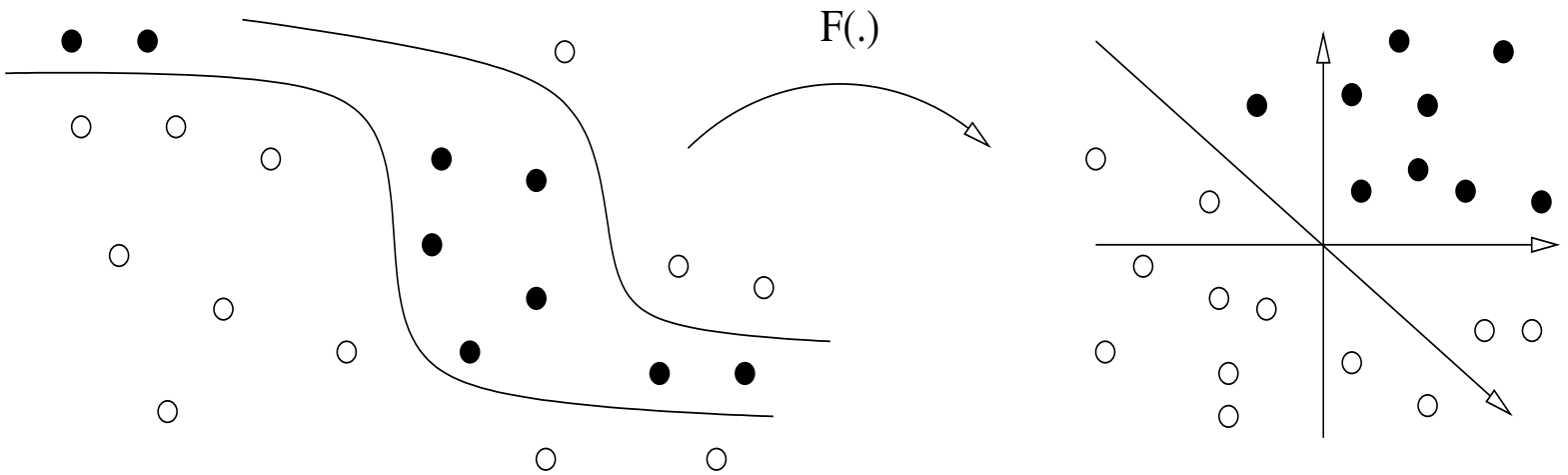
Support vector machines



- linear classifier
  - maximal margin linear separation
  - minimal distances for misclassified

Kernel methods

non-linear into linear separations in higher dimensional space



- linear discriminant given by dot product $< F(x_i), F(x_j) >$
- back into low-dimensional space by a kernel $K(x_i, x_j)$

VC dimension

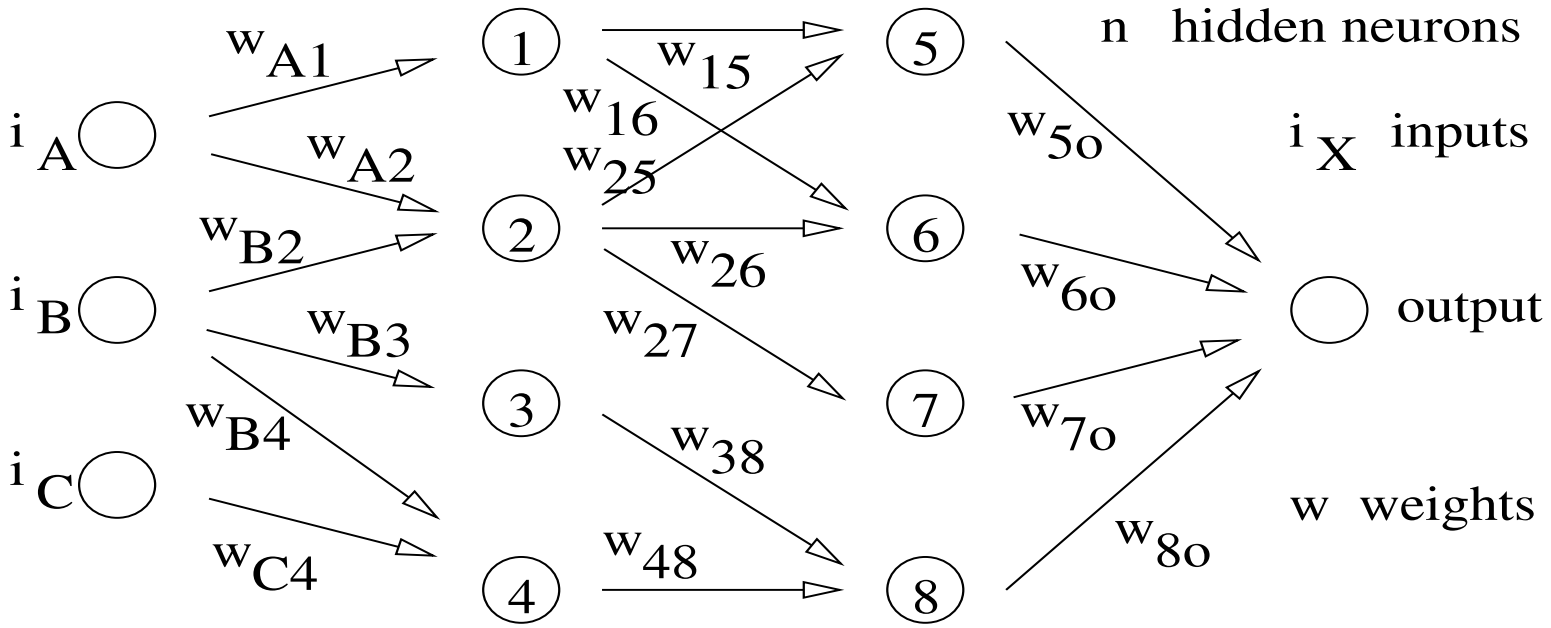
Vapnik-Chervonenkis dimension

- classification error estimation
  - more power a method has more prone it is to overfitting
  - misclassifications for a binary $f(\alpha)$ classificator
  - iid samples drawn from an unknown distribution
    - $R(\alpha)$ probability of misclassification - in real usage
    - $R^{emp}(\alpha)$ fraction of misclasified cases of a training set
  - then with probability $1 - \eta$, training set of size N

$$R(\alpha) < R^{emp}(\alpha) + \sqrt{\frac{h(1+ln(2N/h)) - ln(\eta/4)}{N}}$$

- h the VC dimension
  - size of maximal sets that $f(\alpha)$ can shatter
  - 3 for a line classifier in a 2D space

artificial neural networks



- neuron activation function
  - $f_2(e) = f_2(i_A w_{A2} + i_B w_{B2} - c_2)$
  - f_i is non-linear, usually sigmoid, with c_i given constants

ANN learning

error backpropagation

- iterative weight adjusting
  - compute errors for each training case
    $\delta = \text{desired} - \text{computed}$
  - propagate the δ backward: $\delta_5 = \delta \cdot w_{5o}$
    $\delta_1 = \delta_5 \cdot w_{15} + \delta_6 \cdot w_{16}$, ...
- adjust weights to new values
  - $w_{A1}^{new} = w_{A1} + \eta \cdot \delta_1 \cdot df_1(e)/de \cdot i_A$
  - $w_{15}^{new} = w_{15} + \eta \cdot \delta_5 \cdot df_5(e)/de \cdot f_1(e)$
  - ...
- kind of gradient descent method
  - other (sigmoid function) parameters can be adjusted as well
  - converges to a local minimum of errors

self-organizing maps

- kind of an unsupervised version of ANNs
  - the map is commonly a 2D array
  - array nodes exhibit a simple property
  - each input connected to each output
  - used to visualize multidimensional data
  - similar parts should behave similarly
- competitive learning of the network
  - nodes compete to represent particular data objects
  - each node of the array has its vector of weights
  - initially either random or two principal components
  - iterative node weights / property adjusting
    - take a random data object
    - find its best matching node according to nodes' weights
    - adjust node weights / property to be more similar to the data
    - adjust somewhat other neighboring nodes too

generative topographic map

- GTM characteristics
  - non-linear latent variable model
  - probabilistic counterpart to the SOM model
  - a generative model
    - actual data are being modeled as created by mappings from a low-dimensional space into the actual high-dimensional space
    - data visualization is gained according to Bayes' theorem
    - the latent-to-data space mappings are Gaussian distributions
    - created densities are iteratively fitted to approximate real data distribution
    - known Gaussian mixtures and radial basis functions algorithms

Nearest neighbor

- case-based reasoning classification
  - diagnosis set to the most similar determined case
  - how to measure distances between particular cases?
- *k*-NN
  - take the *k* most similar cases, each of them has a vote
  - simple but frequently works for (binary) classification
- common problems
  - which properties are significant, which are just noise
  - suitable sizes of similar cases, how to avoid outliers

Relations

the right descriptive features - the right similar cases

- search for important gene expressions and patient cases
- unsupervised methods
  - data clustering
    - for the similar data cases
  - data mining
    - for the important features
- supervised methods
  - Bayesian network inference
    - informatics and statistics
  - minimum message length
    - informatics and algebra
  - inductive logic programming
    - informatics and logic

Cliques

- graph approach to clustering
  - transform given data table to a graph - vertices for genes edges for gene pairs with similarity above a threshold
  - to find the least graph alteration to result in a clique graph

- CAST algorithm
  - iterative heuristic clique generation
  - a clique construction from available vertices
    - initiate with a vertex of maximal degree
    - while a distant vertex taken or close vertex free
      add the closest vertex into the clique
      remove the farthest vertex from the clique

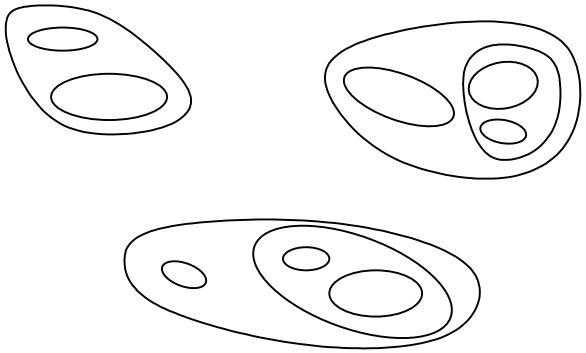
Clustering

standard clustering technics

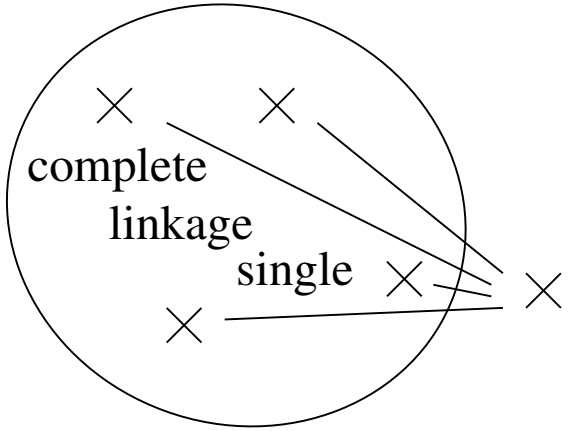
to make separated homogenous groups

- center based methods
  - k-means as the standard
  - c-means, qt-clustering
- hierarchy methods
  - agglomerative bottom-up
  - divisive top-down
- combinations
  - two-steps approach

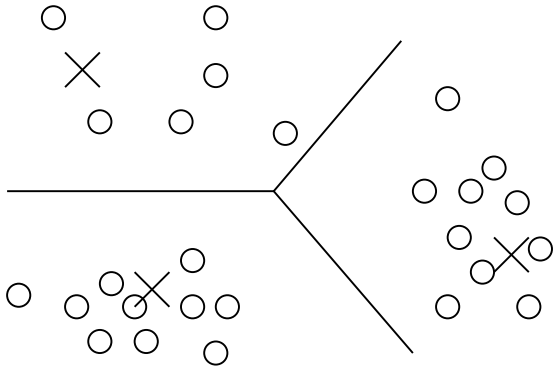
Cluster structures



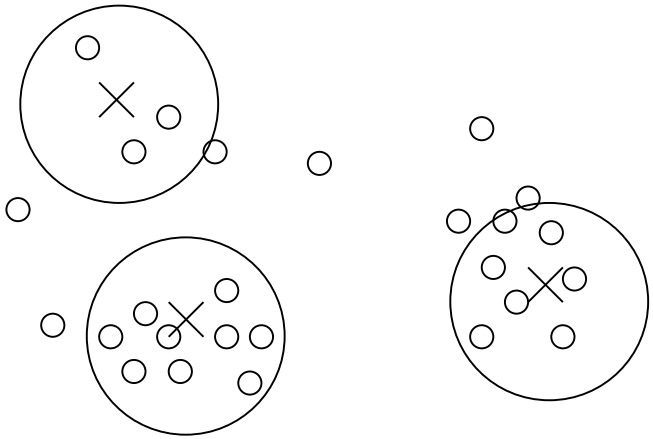
hierarchical clustering



neighbor joining



k-means clustering



qt-clustering

Distances

how to measure object-object (dis)similarity

Euclidean distance	$[\sum_i (x_i - y_1)^2]^{1/2}$
Manhattan distance	$\sum_i \lvert x_i - y_1 \rvert$
Power distance	$[\sum_i (x_i - y_1)^p]^{1/p}$
maximum distance	$\max_i \{\lvert x_i - y_i \rvert\}$
Pearson's correlation	dot product for normalized data
percentage disagreement	fraction of $x_i \neq y_i$

- metric significance
  - different powers usually do not significantly alter results
  - more different distance measuring should change cluster compositions

Hierarchical clustering

neighbor joining - common agglomerative method

hierarchical tree creation

- joining the most similar clusters
  - single linkage - nearest neighbor
    - distances according to the most similar cluster objects
  - complete linkage - furthest neighbor
    - distances according to the most distant cluster objects
  - average linkage
    - cluster distances as mean distances of respective elements
  - Wards method - information loss minimization
    - takes minimal variance increase for possible cluster pairs

The k-means

the most frequently used kind of clustering

- k-mean clustering algorithm
  - start: choose initial k centers
  - iterate for objects (e.g. genes) being clustered: compute new distances to centers, choose the nearest one
  - for each cluster compute new center
  - end when no cluster changes
  - to put less weight on similar microarrays
- pros
  - usually fast, does not compute all object-object distances
- cons
  - amount and initial positions of centers highly affect results

Center count

- do not add more clusters when it does not increase gained information sufficiently

- center selection
  - random
    - make k-means clustering several times
  - PCA
    - principal components lie in data clouds
  - data objects
    - choose distant objects, with weights
  - two steps
    - take larger amount of clusters, then do hierarchical clustering on the result centers

Cluster fit

how convenient are the gained clusters

- k-mean clustering
  - intra-cluster vs. out-of-cluster distances for objects
  - ratios of inter-cluster to intra-cluster distances
    - the Dunn's index
    - inter / intra variances
- hierarchical clustering
  - variances of each cluster
  - similarity for cluster means
  - bootstrapping for objects with suitable inner structures

Alternative clustering

- qt (quality threshold) clustering
  - choose maximal cluster diameter instead of center count
  - try to make maximal cluster around each data objects take the one with the greatest amount of objects inside
  - call it recursively on the rest of the data
  - more computation intensive - more plausible than k-means
  - can be done alike for maximal cluster sizes
- soft c-means
  - each object (gene) is in more clusters (gene families)
  - object belonging degrees, sums equal to one
  - similar to k-means, stop when small cluster changes
  - suitable for lower amounts of clusters
- spectral clustering
  - object segmentation according to similarity Laplacian matrix eigenvector of the second smallest eigenvalue

Dependency description

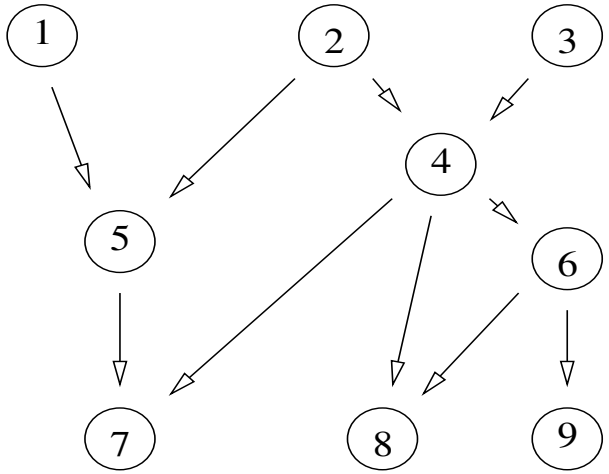
used for characterization, classification and compression

- BN
  - Bayesian networks
  - what depends on what, which variables are independent
  - then fast, suitable inference computing
- MML
  - minimal mesage length
  - shortest form of an object / feature description
  - real amount of information an object contains
- ILP
  - inductive logic programming
  - to prove most of the positive / least of the negative cases
  - accurate given objects vs. background characterization

Graphical models

- joint distributions
  - $\Pr(x_1, x_2, x_3) = \Pr(x_1 \mid x_2, x_3) \cdot \Pr(x_2 \mid x_3) \cdot \Pr(x_3)$
  - intractable for slightly larger amounts of variables
  - used to compute important probabilities themselves
  - used for conditional probabilities
    $\rightarrow$ for Bayesian inference
- conditional independence
  - A and B independent under C: $A \perp\!\!\!\perp B \mid C$
  - $A \perp\!\!\!\perp B \mid C$: $\Pr(A, B \mid C) = \Pr(A \mid C) \cdot \Pr(B \mid C)$
  - after a few rearrengements: $\Pr(A \mid B, C) = \Pr(A \mid C)$
    - Markov processes are just one example of conditional independence
- graphs and relations
  - simplifying the structure with stating just some valid dependencies, no edges $\rightarrow$ (conditional) independence
  - edges stating the only dependencies

Bayesian systems



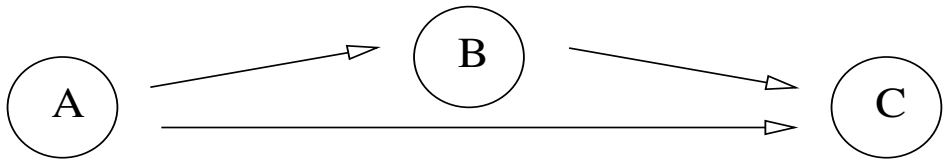
DAG of dependencies

x_5 depends on x_1 x_2

- $\Pr(x_5 \mid x_1, x_2, x_3, x_4) = \Pr(x_5 \mid x_1, x_2)$
- any node is - given all its parents - (conditionally) independent with all the nodes which are not its descendants (e.g. x_3, x_5 independent at all)

Bayesian network

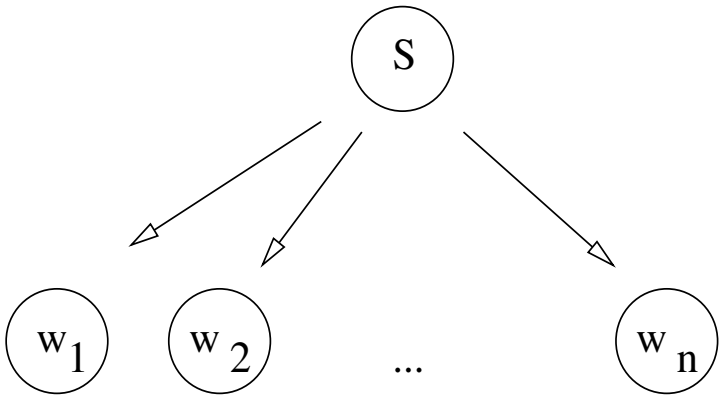
- inference
  - inference along the DAG is computed acoording to the decomposition of conditional probabilities
  - inference along the opposite directions is computed acoording to the Bayesian approach



- $\Pr(A, B, C) = \Pr(A|B, C) \cdot \Pr(B|C) \cdot \Pr(C)$
  - how to compute $\Pr(A = True|C = True)$

$$\Pr(A = True|C = True) = \Pr(C = True, A = True)/\Pr(C = True)$$
$$= \sum_B \Pr(C = True, B, A = True)/\sum_{A,B} \Pr(C = True, B, A)$$

Naive Bayes



S ... mail/spam state
w_i individual words

- assumption of independent outcomes
  - used e.g. for spam classification
  - $S = argmax_s \Pr(S = s) \prod_j \Pr(O_j = w_j | S = s)$
  - possible to compute fast online with 10^4 items

Items to remember

Nota bene:

classification methods

- learning technics
  - vectors, kernels, SVM, ANN
  - conditional independence

- clustering methods
  - hierarchical clustering
  - k-means, qt-clustering