Probabilistic Approaches

Chirayu Wongchokprasitti, PhD

University of Pittsburgh

Center for Causal Discovery

Department of Biomedical Informatics

chw20@pitt.edu

http://www.pitt.edu/~chw20

Overview

- Independence in Bayesian Networks
- Model Selection
- Regression Models
- Kernel Methods
  - Support Vector Machines
  - Principal Component Analysis
- Bayesian Network Classifiers
- Ensemble learning
  - Bagging
  - Random Forest
  - Boosting
  - Bayesian Model Averaging

Independence in Bayesian Networks

Definition

P(X|Parent(X))

Each variable is conditionally independent of its "non-descendants" given its "parents"

D-Separation

A and B are d-separated by C if all paths from a vertex of A to a vertex of B are blocked (with respect to C)

D-Separation

A and B are d-separated by C if all paths from a vertex of A to a vertex of B are blocked (with respect to C)

"Reachability"

D-Separation

A and B are d-separated by C if all paths from a vertex of A to a vertex of B are blocked (with respect to C)

"Reachability"

No active paths ➔ Independence

D-Separation

A and B are d-separated by C if all paths from a vertex of A to a vertex of B are blocked (with respect to C)

"Reachability"

No active paths ➔Independence

If A and B are d-separated by C

Then $A \perp B \mid C$

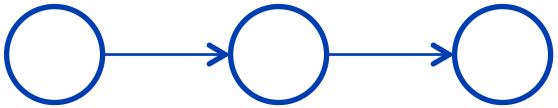
- Independence
- Model selection
- Regression models
- Kernel methods
- Bayesian classifiers
- Ensemble learning

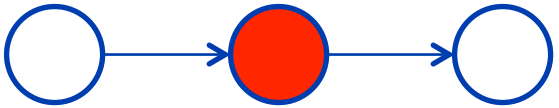
D-Separation

Active Triplets
"Variables Dependent"

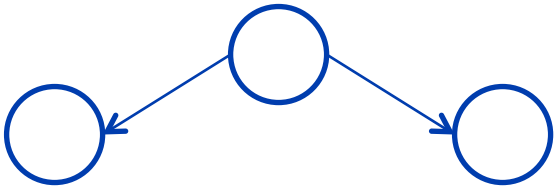
Inactive Triplets
"Variables Independent"

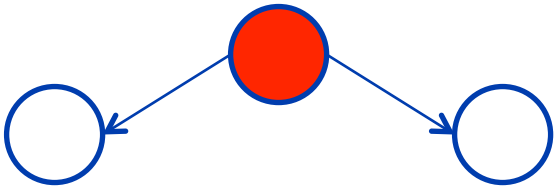
Causal Chain
"Indirect Cause"



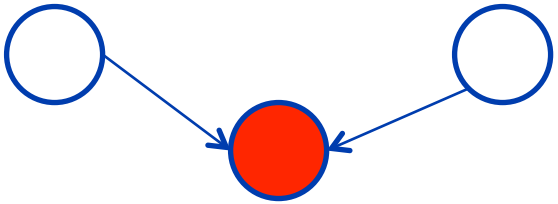


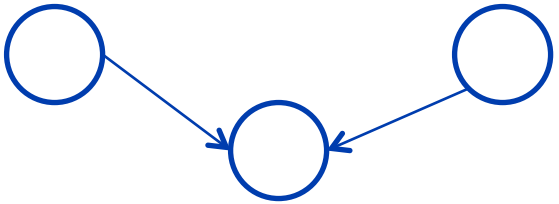
Common Cause



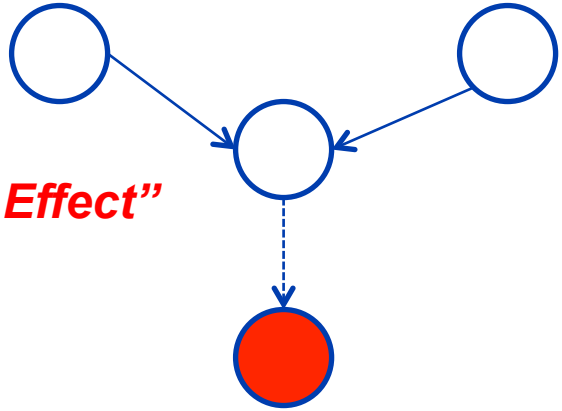


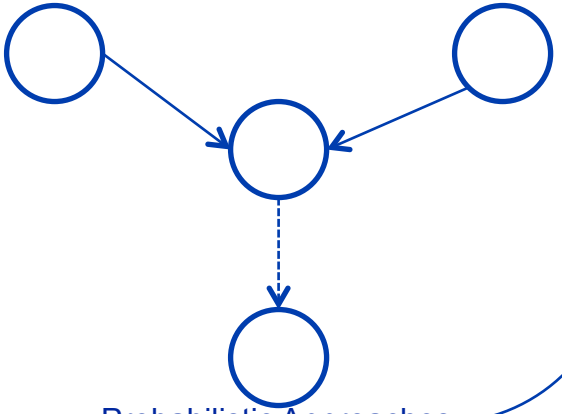
Common Effect



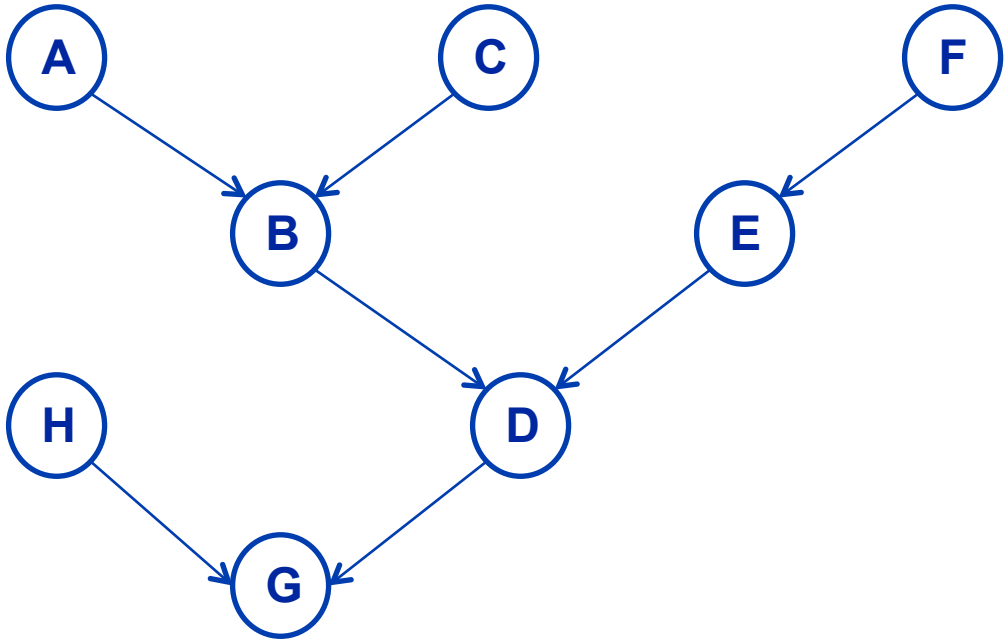


"Explain Away Effect"





D-Separation



$$A \perp F$$

$$A \perp F \mid D$$

$$A \perp F \mid G$$

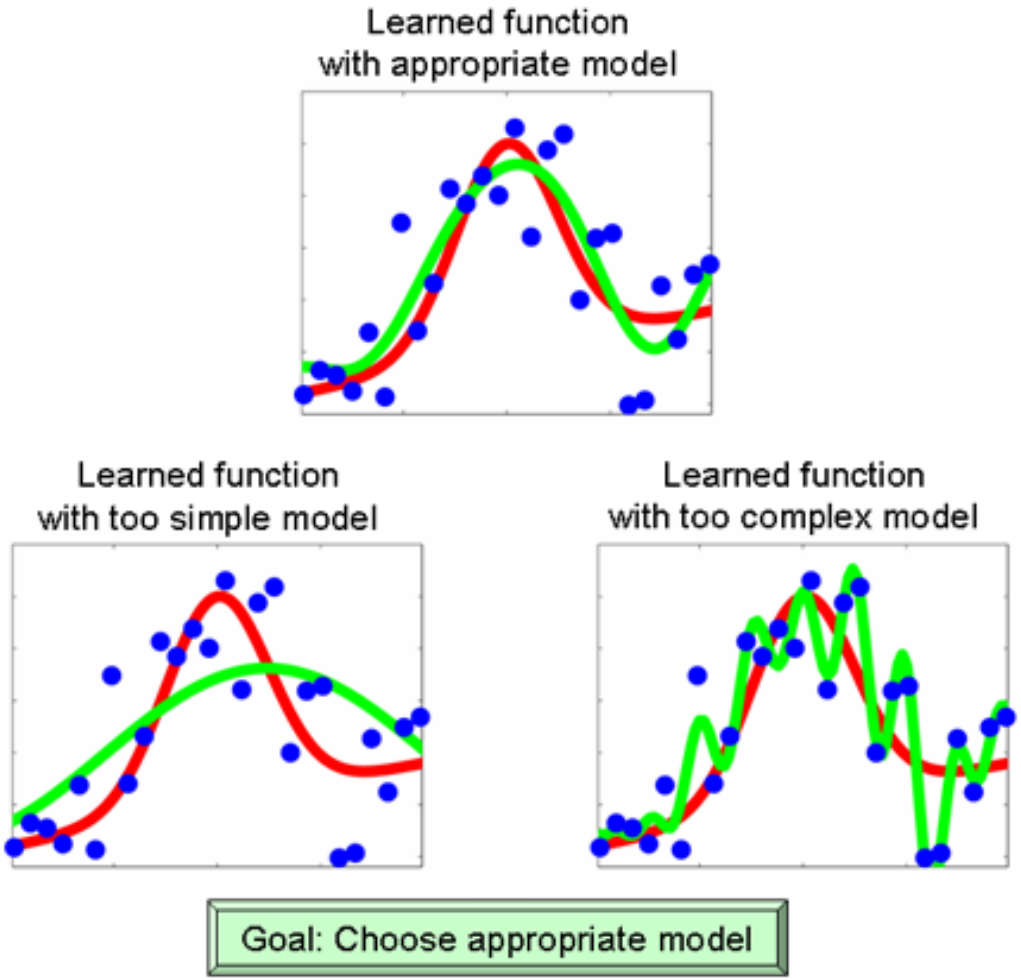
$$A \perp F \mid H$$

Model selection

Definition

Model selection is the task of selecting a statistical model from a set of candidate models given data.

Model Selection



Model selection

- Bias-variance Trade-off
- Generalization Theory (Statistical Learning Theory)
- Over-fitting Prevention
    1. Cross Validation
    2. Regularization
    3. Feature Selection

Bias-variance Trade-off

Suppose $\hat{f}(x)$ is a fit model with some training data, and let (x_0,y_0) be a test observation drawn from the population. If the true model is $Y = f(x) + \varepsilon$ with $f(x) = E(Y \mid X = x)$, then

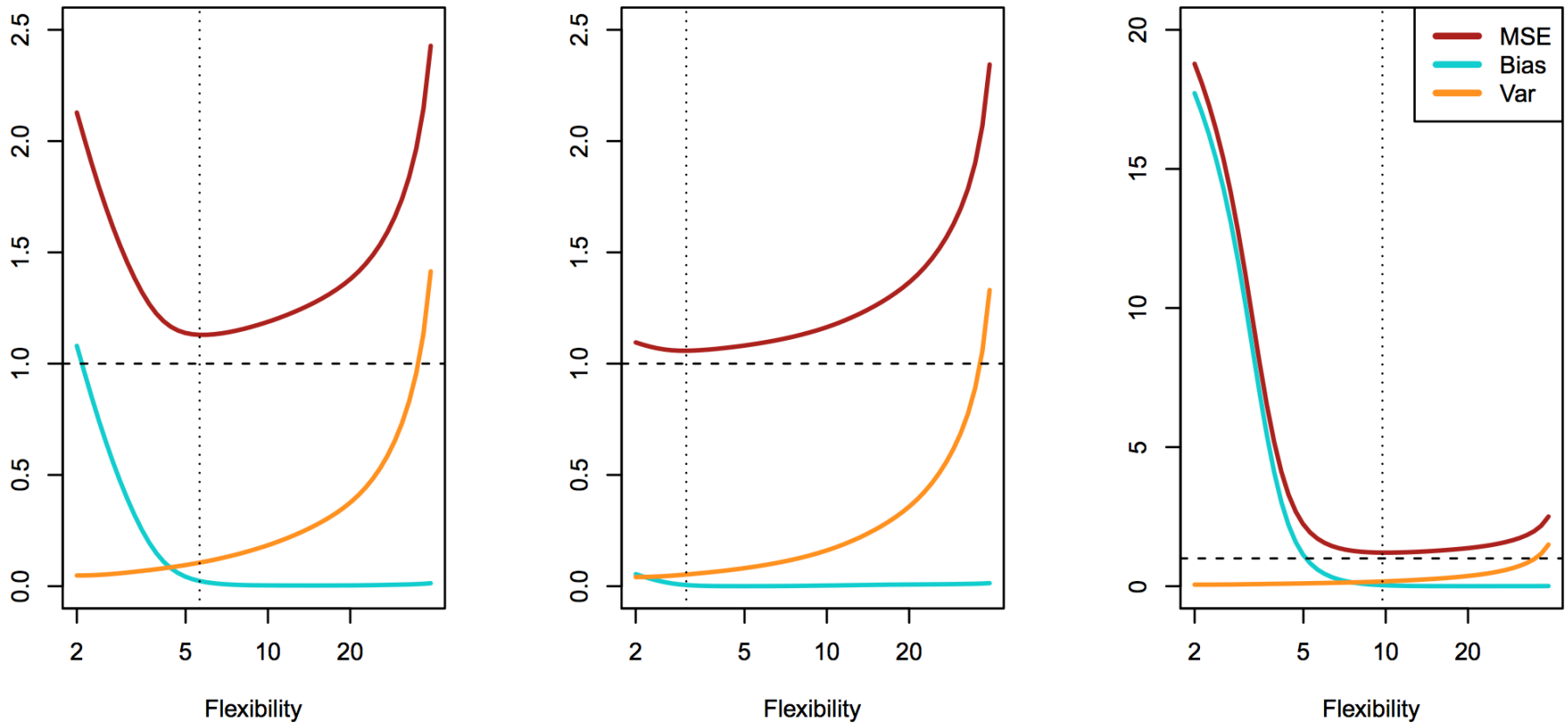
$$E\left(y_0 - \hat{f}(x_0)\right)^2 = Var(\hat{f}(x_0)) + [Bias(\hat{f}(x_0))]^2 + Var(\varepsilon)$$

The expectation averages over the variability of y_0 as well as the variability in the training data.

$$Bias(\hat{f}(x_0)) = E[\hat{f}(x_0)] - f(x_0)$$

So, the flexibility of $\hat{f}$ increases, its variance increases, and its bias decreases.

Bias-variance Trade-off (cont'd)



James et al., (2013). An introduction to statistical leanring. Ch 2. pp. 36. fig 2.12.

Model Selection: Statistical Learning Theory

- Consistency (Guarantee Generalization)
- Model Convergence Speed (A Measure for Generalization)
- Generalization Capacity Control
- A Strategy for Good Learning Algorithms

1. Cross Validation

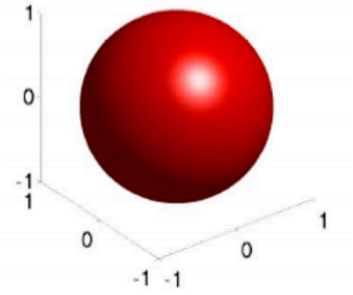
- Estimates can be used to
  - select best model
  - give an idea of the test error of the final chosen model
- Idea is to randomly divide the data into K equal-sized parts. Leave out part k. Fit the model to the other K − 1 parts (combined).Then obtain predictions for the left-out k^{th} part.
- Repeat for each part k = 1, 2, . . . K, and the results are combined.

2. Regularization

- Fit a model involving all p predictors, but the estimated coefficients are shrunken towards zero relative to the least squares estimates. This shrinkage (also known as regularization) has the effect of reducing variance and can also perform variable selection.
- There are two main regularization methods: Ridge regression (L2) and Lasso (L1).

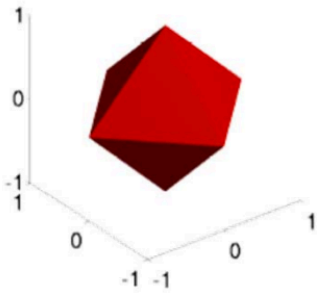
- Ridge regression (L2):

$$RSS + \lambda \sum_{j=1}^{p} \beta_j^2$$



- Lasso (L1):

$$RSS + \lambda \sum_{j=1}^{p} |\beta|_j$$



http://www.cs.cmu.edu/~epxing/Class/10701/slides/lecture10-fit.pdf

3. Feature Selection

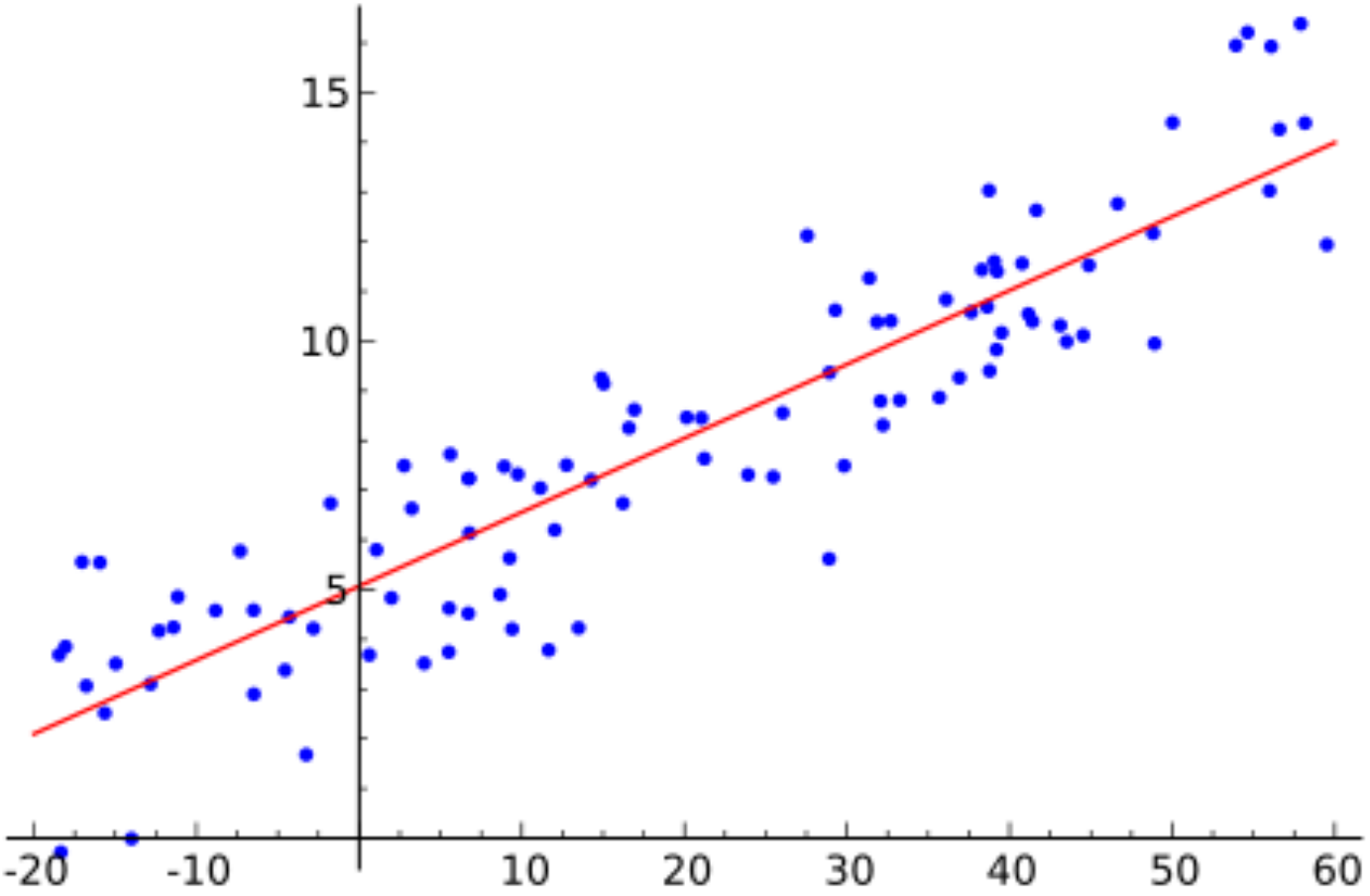
- Identify a subset of the p predictors believed to be relevant to the response. Then fit a model using least squares on the reduced set of variables.
- Such methods: Forward Selection, Backward Elimination, or Lasso (L1) regularization etc.

Regression models

Definition

- Regression analysis is a statistical technique for estimating functional relationships among variables.
- True regression functions are never linear!
- Although it may seem overly simplistic, linear regression is extremely useful both conceptually and practically.

Regression: The basic idea



Linear regression

http://en.wikipedia.org/wiki/Regression_analysis

Linear Regression

In linear regression, the dependent variable, y_i, is a linear combination of the parameters (but need not be linear in the independent variables).

For example, in simple linear regression there is one independent variable: x_i, and two parameters, β_0 and β_1:

$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i, \quad i = 1, \ldots, n.$$

In multi-variable linear regression, there are several independent variables (or functions of independent variables).

Linear Regression

The dependent variable, y_i, does not need not be linear in the independent variables.

For example, adding the term x^2 yields a parabola:

$$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \varepsilon_i, \quad i = 1, \ldots, n.$$

… but it is still linear regression!

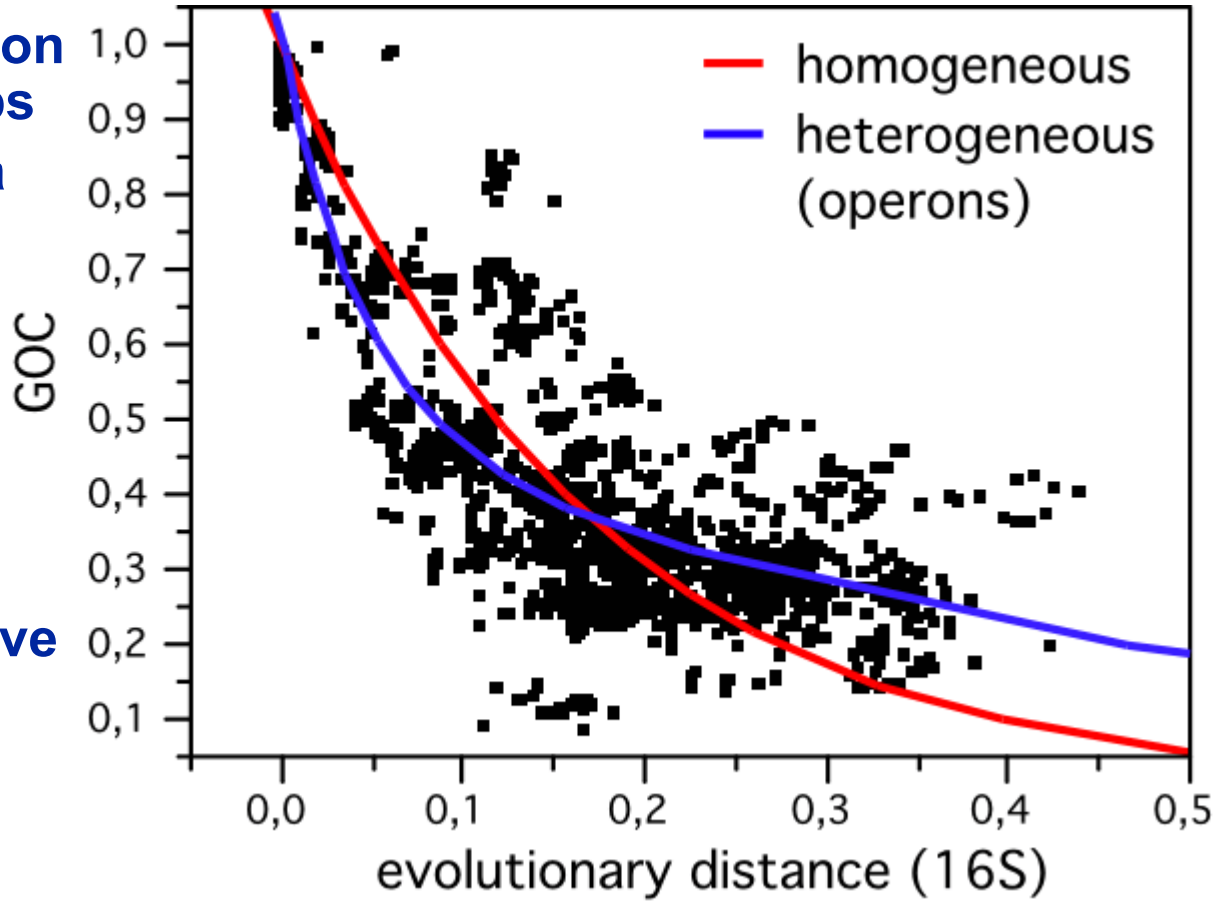
Even though the expression is quadratic in the independent variable, it is linear in the parameters β_0, β_1, and β_2

Non-linear Regression

Regression can focus on non-linear relationships

(we typically assume a functional form and fit the function to data).

- **Polynomials**
- **Step Functions**
- **Splines**
- **Local Regression**
- **Generalized Addictive Models**

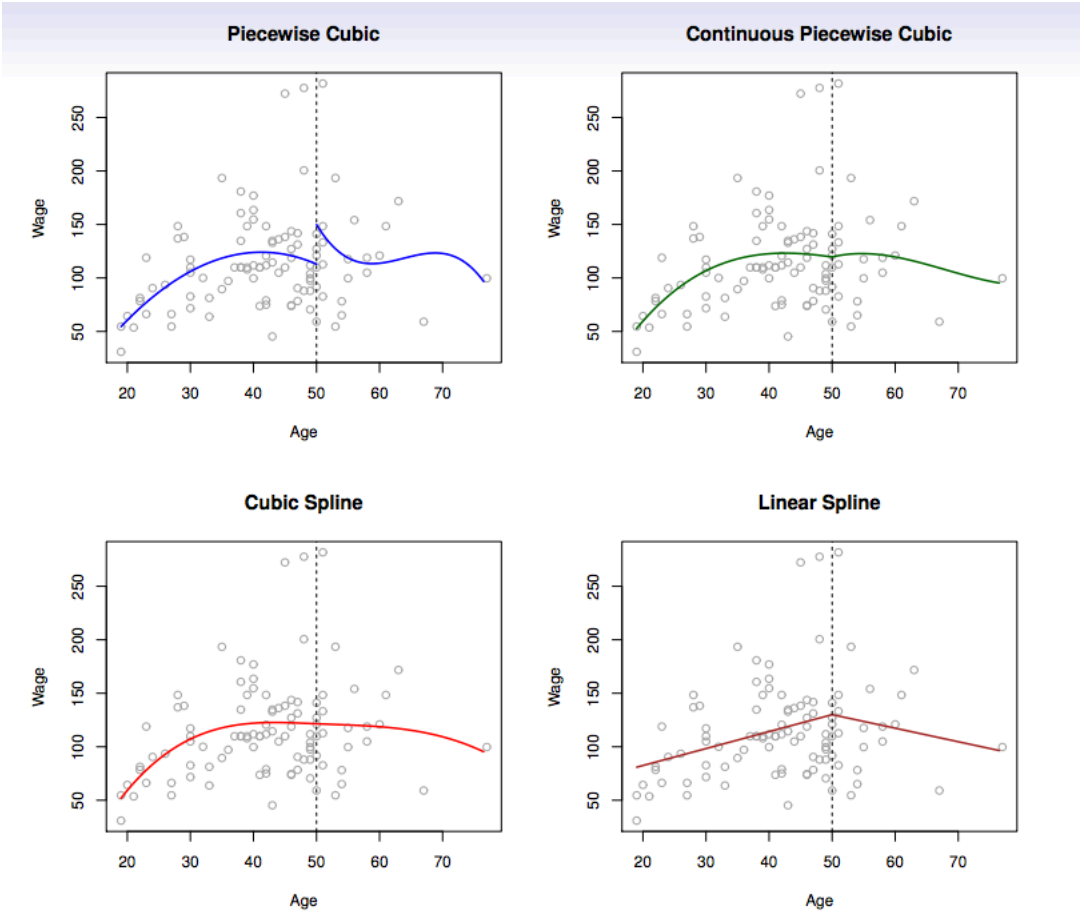


http://wwwabi.snv.jussieu.fr/~erocha/research/ordervsdisorder.html

Piecewise Polynomials

Instead of a single polynomial in X over its whole domain,

we can rather use different polynomials in regions defined by knots.



https://lagunita.stanford.edu/c4x/HumanitiesSciences/StatLearning/asset/nonlinear-handout.pdf

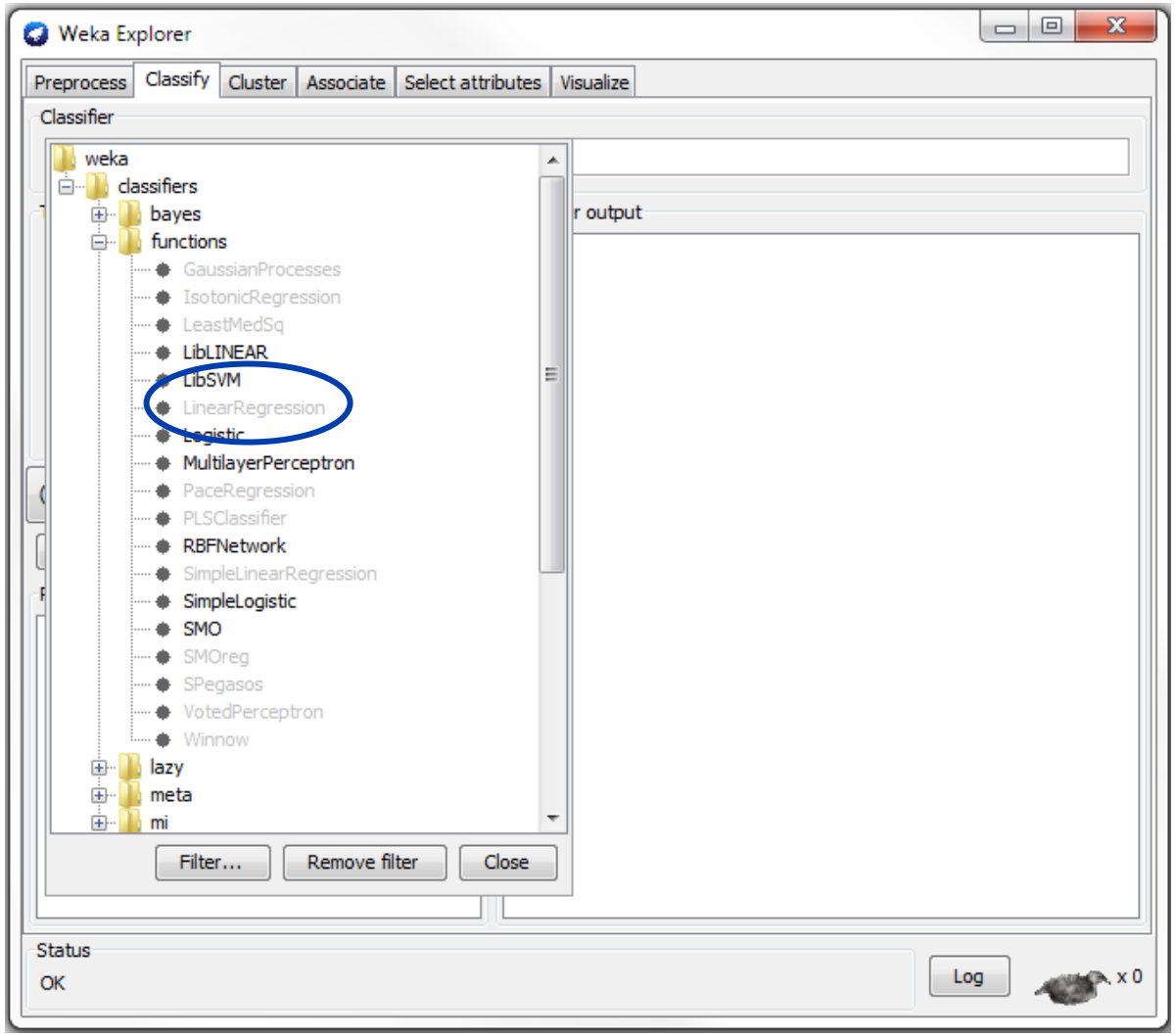
Weka

http://www.cs.waikato.ac.nz/ml/weka/

- Weka: A great piece of machine learning/data analysis software written in Java
- Runs on most platforms
- We will be using it in some parts of this course

Weka: A flightless bird (Gallirallus australis) of New Zealand, having mottled brown plumage and short legs.

Regression-based classification in Weka



Kernel methods

Definition

- Kernel methods are a class of algorithms focusing on pattern analysis in data sets.
- Kernel methods map the data into a high-dimensional feature space, where each coordinate corresponds to one feature of the data items, transforming the data into a set of points in Euclidean space.
- They search for relations in the data in the transformed space.
- Because the transformation/mapping can be very general, so can be the relations.

Efficiency of kernel methods

- Kernel methods owe their name to the use of *kernel functions*, which enable them to operate in the feature space without ever computing the coordinates of the data in that space, but rather by simply computing the inner products between the images of all pairs of data in the feature space.
- This operation is often computationally cheaper than the explicit computation of the coordinates (this approach is sometimes called the *kernel trick*).
- We will talk about two (of many) algorithms capable of operating with kernels:
  - Support Vector Machine (SVM)
  - Principal Components Analysis (PCA).

Support Vector Machines: Principles

- Support vector machines (SVMs), an instance of kernel methods, are supervised learning models (with associated learning algorithms) used for classification and regression analysis.
- An SVM constructs a hyperplane (or a set of hyperplanes) in a high-dimensional space, which separates points in the space and, hence, can be used as a basis for classification, regression, or other tasks.

Support Vector Machines: Hyperplane

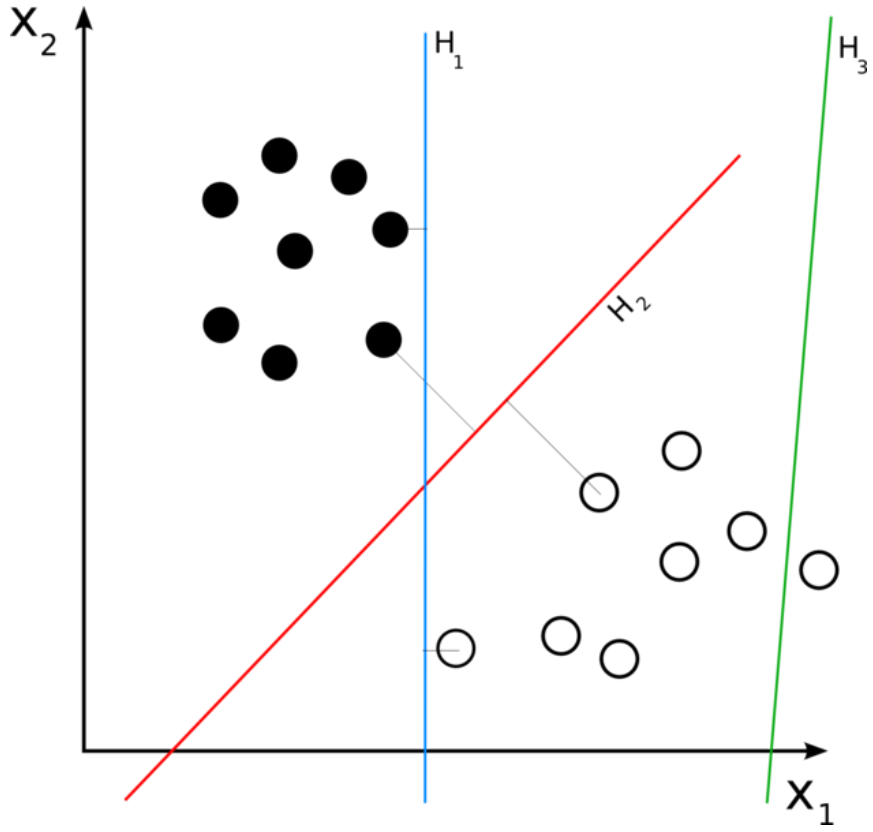
- A hyperplane in p dimensions is a flat affine subspace of dimension p − 1.
- In general the equation for a hyperplane has the form

$$\beta 0 + \beta 1X1 + \beta 2X2 + \ldots + \beta pXp = 0$$

- In p = 2 dimensions a hyperplane is a line.
- The vector β = (β1, β2, · · · , βp) is called the normal vector — it points in a direction orthogonal to the surface of a hyperplane.

Support Vector Machines: Principles

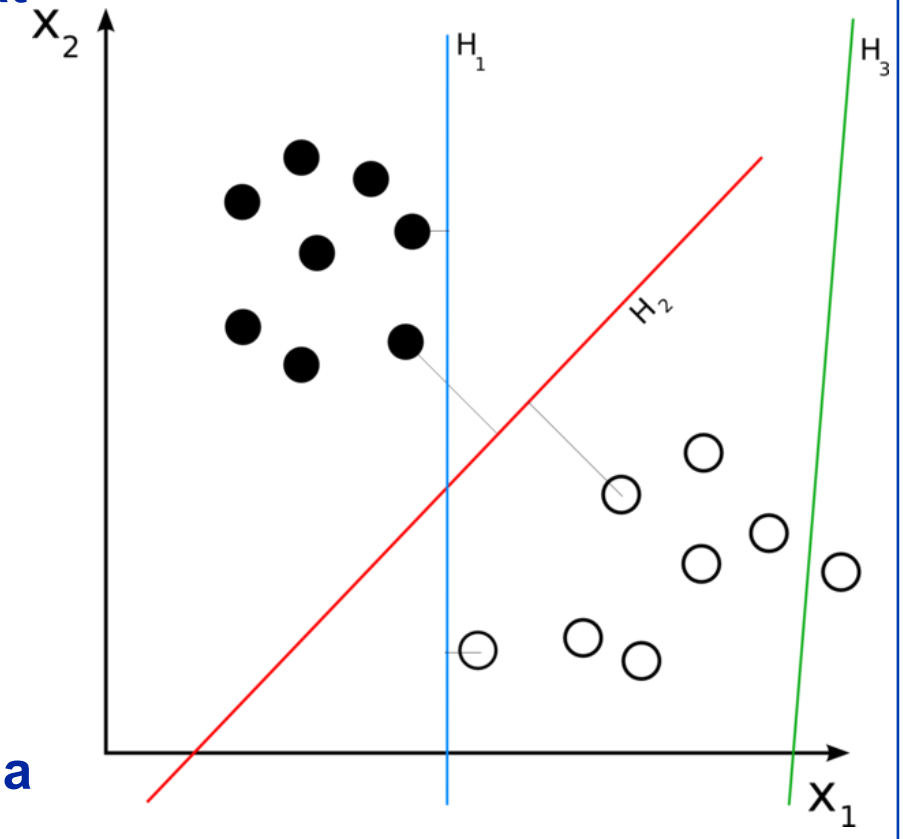
- Suppose (in the simplest case) we have a data set that consists of points belonging to one of two classes, characterized by two attributes.
- Our goal is classification, i.e., deciding to which class a new data point (that we have not yet seen) belongs.
- In the case of SVMs, each data point is viewed as a *2*-dimensional vector (a list of *2* numbers), and we want to know whether we can separate such points with a line (in general, for *p* attributes, (*p*−*1*)-dimensional hyperplane).
- This is called a linear classifier.



http://en.wikipedia.org/wiki/Support_vector_machine

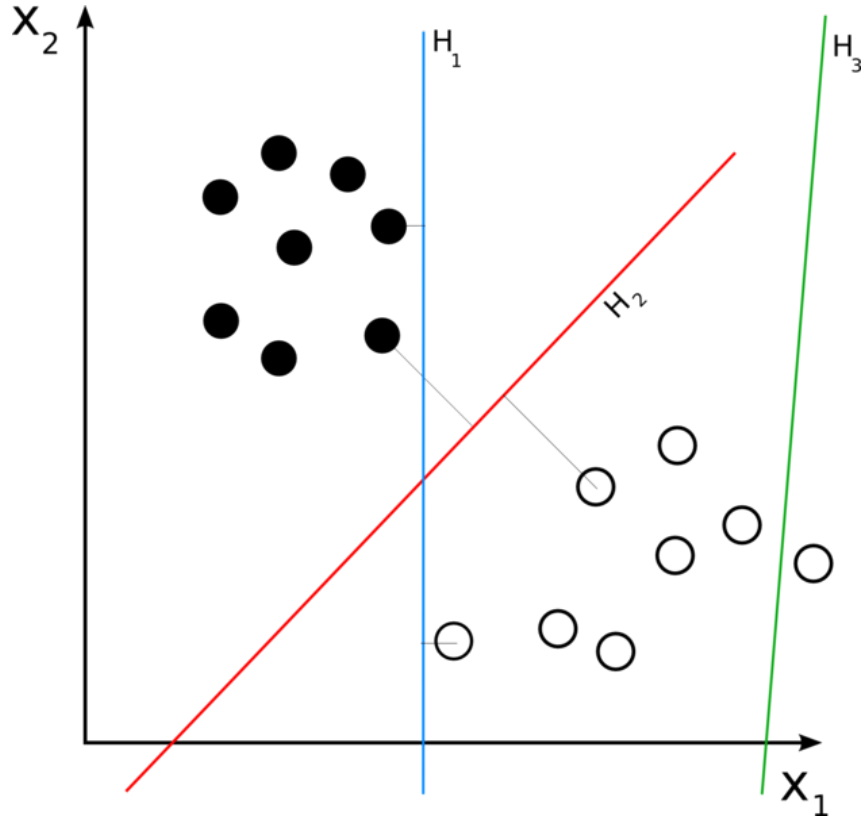
Support Vector Machines: Principles

- There are many hyperplanes that might classify the data.
- The best hyperplane is the one that represents the largest separation, or margin, between the two classes.
- We focus on finding the hyperplane that maximizes the distance from it to the nearest data point on each side.
- If such a hyperplane exists, it is known as the maximum-margin hyperplane and the linear classifier it defines is known as a maximum margin classifier; or equivalently, the perceptron of optimal stability.



http://en.wikipedia.org/wiki/Support_vector_machine

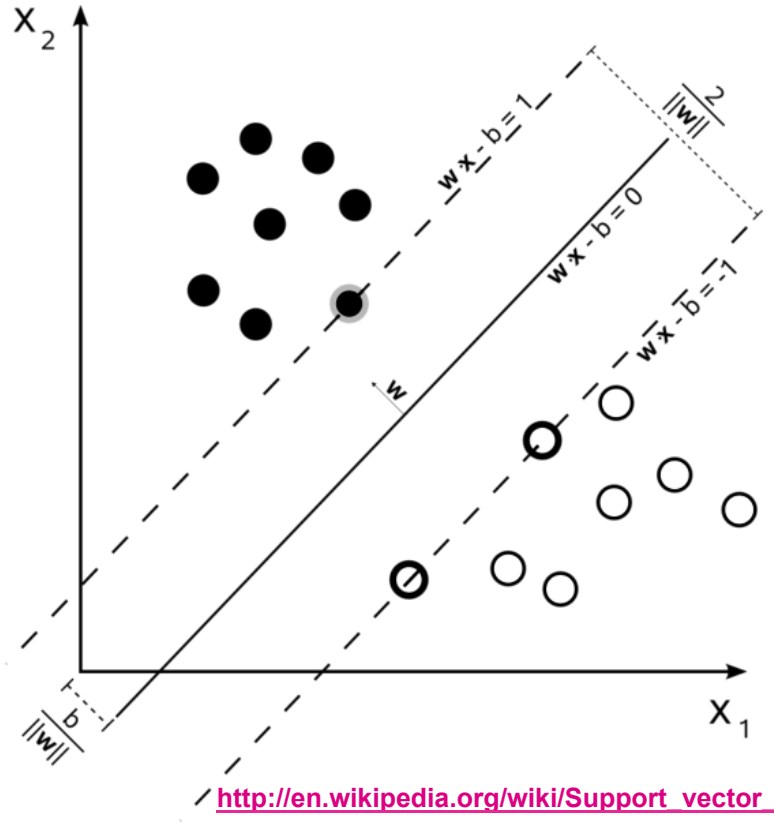
Support Vector Machines: Principles



- The green line does not separate the two classes, the blue and red lines do.
- The red line does it with a wide margin.

http://en.wikipedia.org/wiki/Support_vector_machine

Support Vector Machines: Principles

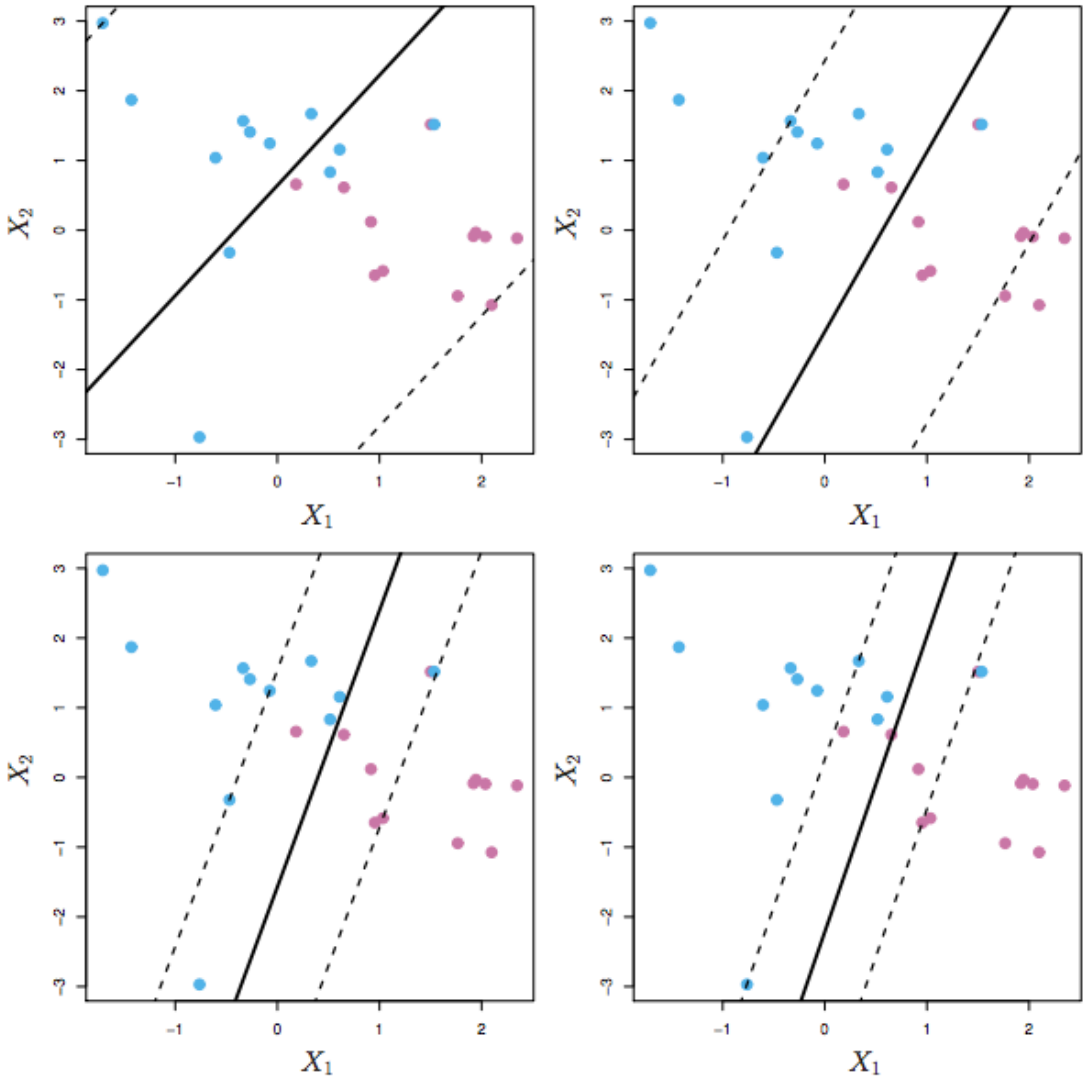


http://en.wikipedia.org/wiki/Support_vector_machine

- Maximum-margin hyperplane and margins for an SVM trained with samples from two classes.
- Samples on the margin are called the support vectors.

Support Vector Machines: Principles

Logistic Regression?

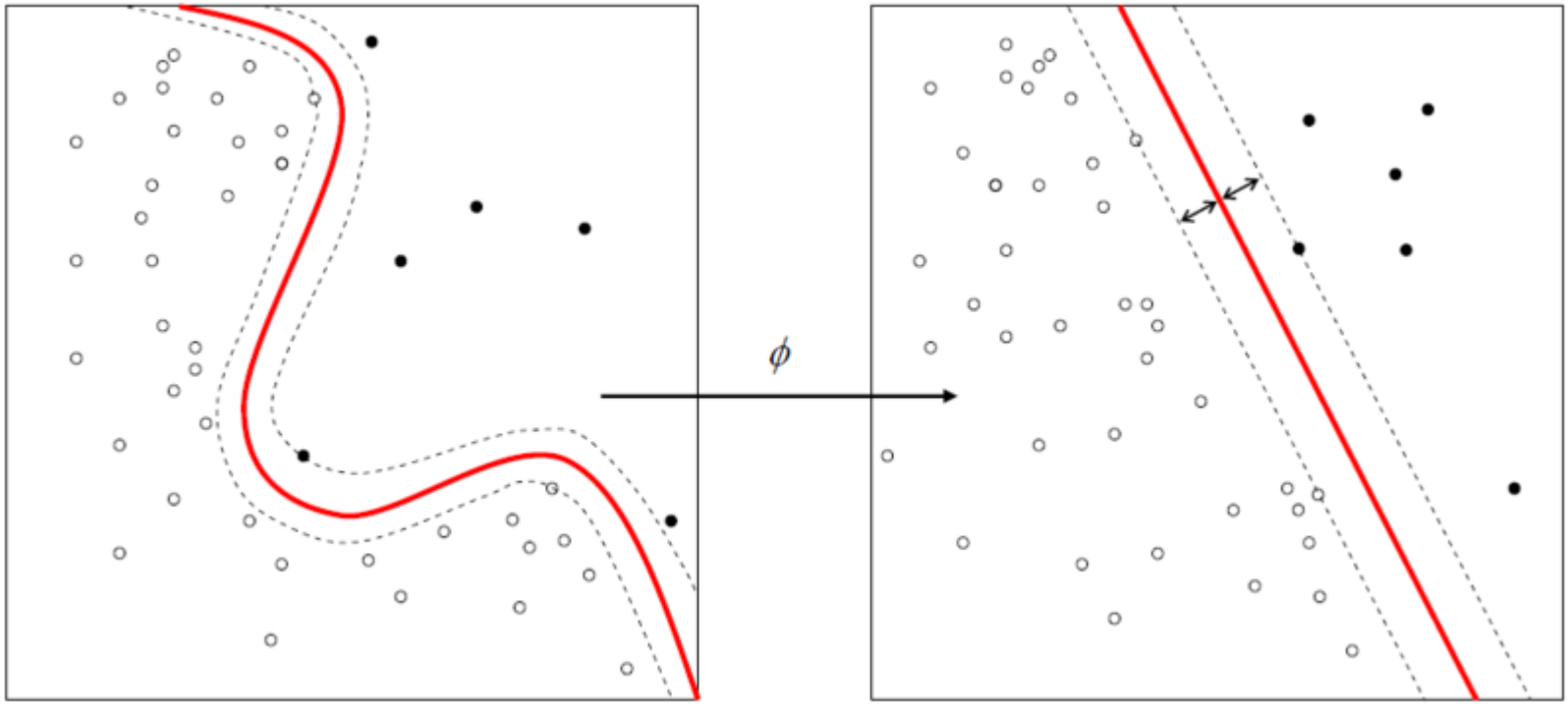


https://lagunita.stanford.edu/c4x/HumanitiesSciences/StatLearning/asset/svm-handout.pdf

Support Vector Machines: Non-linear case

- The original algorithm [Vapnik, 1963] was a linear classifier.
- However, in 1992, Boser, Guyon and Vapnik suggested a way to create nonlinear classifiers by applying the "kernel trick" [Aizerman et al., 1964] to maximum-margin hyperplanes.
- The resulting algorithm is formally similar, except that every dot product is replaced by a nonlinear kernel function.
- This allows the algorithm to fit the maximum-margin hyperplane in a transformed feature space.
- The transformation may be non-linear and the transformed space high dimensional.
- Even though the classifier is a hyperplane in the high-dimensional feature space, it is generally non-linear in the original input space.

Support Vector Machines: Non-linear case



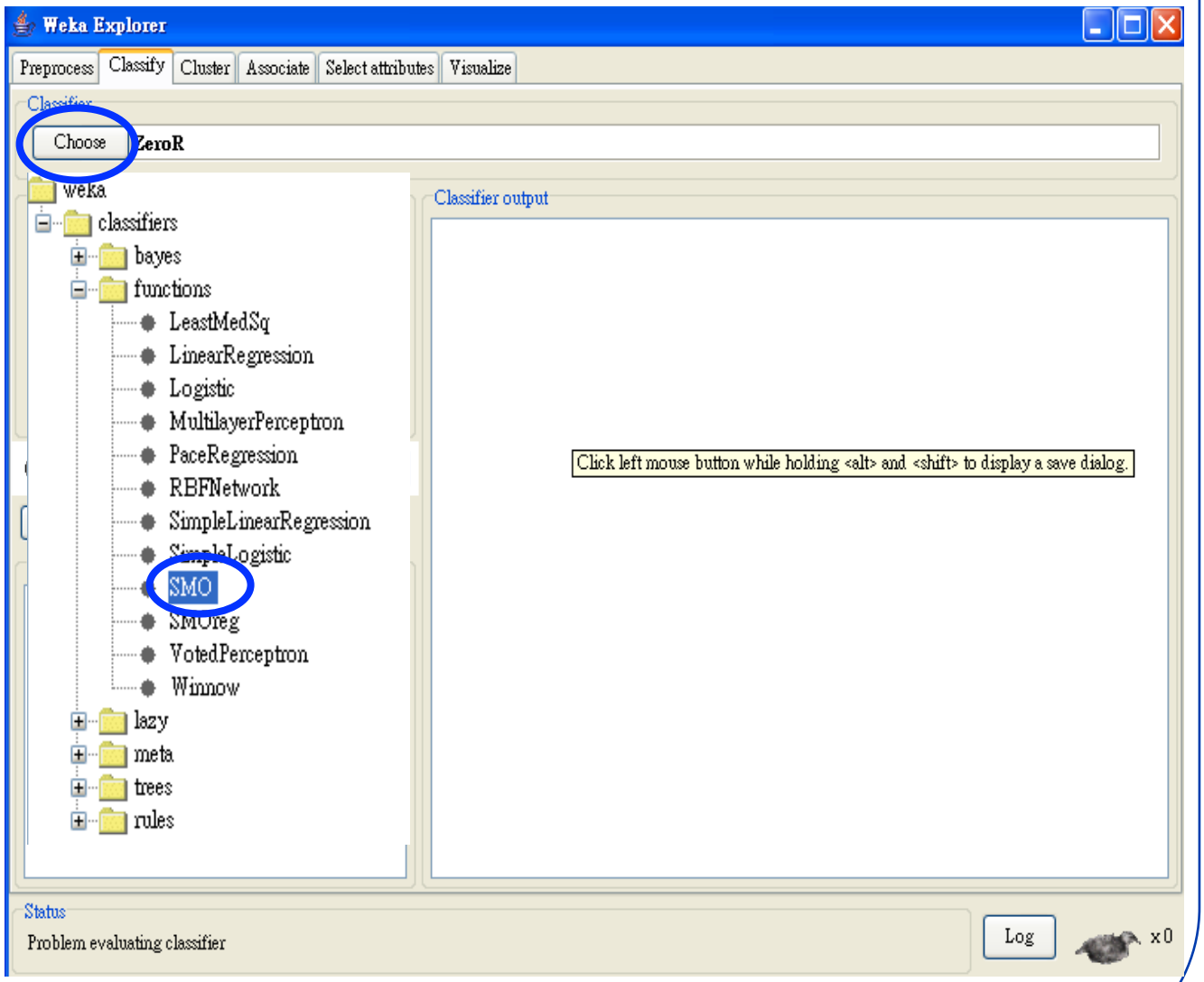
- Transformation Φ changes the multi-dimensional feature space into a space separable by a hyperplane.
- Reverse transformation changes the hyperplane into a non-linear separation function.

http://en.wikipedia.org/wiki/Support_vector_machine

Support Vector Machines in Weka

Use a data set with categorical class and numerical attributes,

e.g., the Iris data.

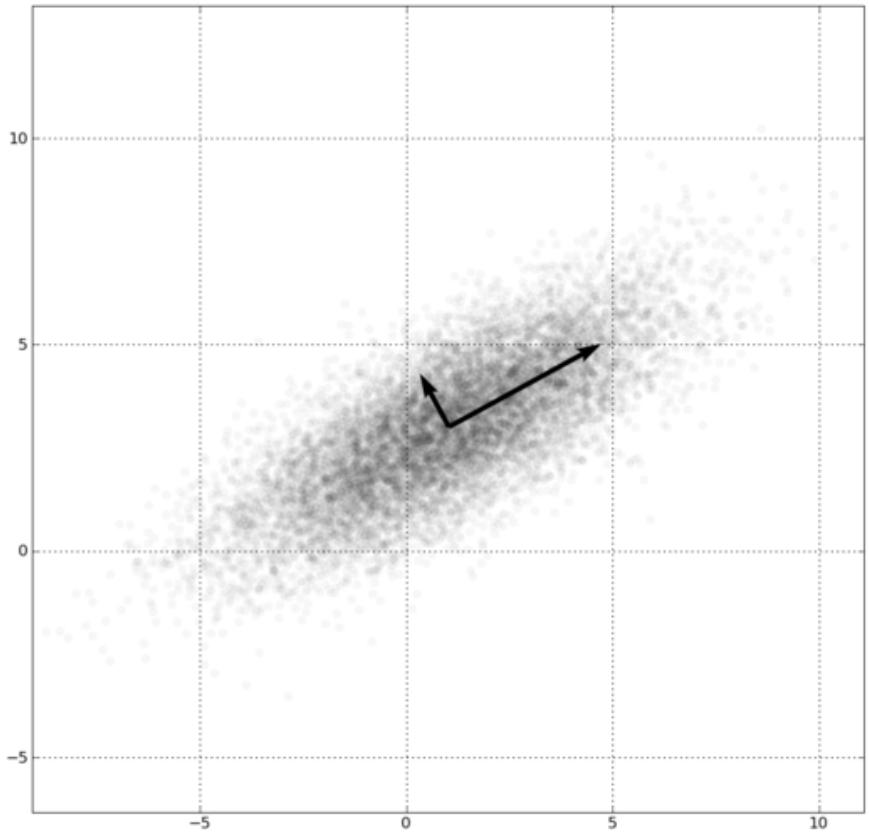


Principal Component Analysis

- PCA produces a low-dimensional representation of a dataset. It finds a sequence of linear combinations of the variables that have maximal variance, and are mutually uncorrelated.
- The number of principal components is typically less than the number of original variables.
- Principal components analysis, a tool used for data visualization or data pre-processing before supervised techniques are applied

Principal Component Analysis: Example

- PCA of a multivariate Gaussian distribution centered at (1,3) with a standard deviation of 3 in roughly the (0.878, 0.478) direction and of 1 in the orthogonal direction.
- The vectors shown are the eigenvectors of the covariance matrix scaled by the square root of the corresponding eigenvalue, and shifted so their tails are at the mean.



http://en.wikipedia.org/wiki/Principal_component_analysis

Principal Component Analysis (PCA)

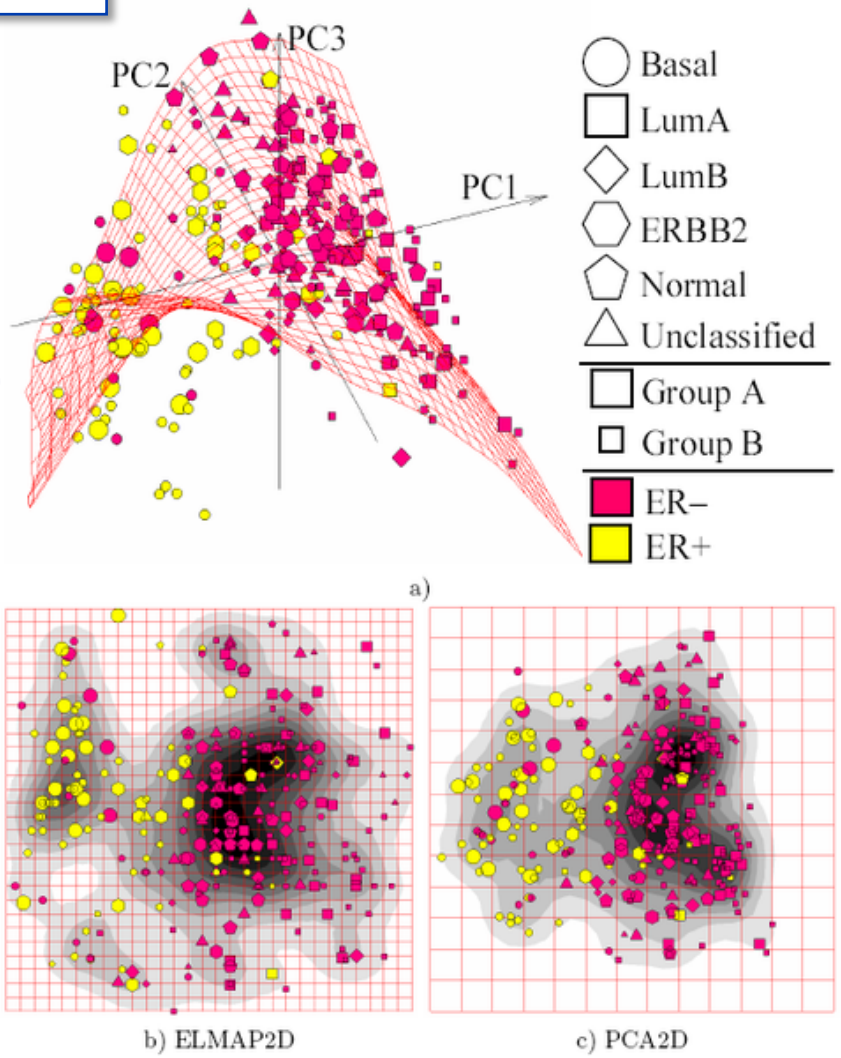
- Invented in 1901 by Karl Pearson.
- Mostly used as a tool in
  - exploratory data analysis
  - making predictive models.
- Be done by
  - eigenvalue decomposition or
  - singular value decomposition
- The results called component scores, or sometimes factor scores

Principal Component Analysis

- PCA is the simplest of the true eigenvector-based multivariate analyses.
- If a multivariate dataset is visualized as a set of coordinates in a high-dimensional data space (1 axis per variable). This is done by using only the first few principal components so that the dimensionality of the transformed data is reduced.
- PCA is closely related to factor analysis. Factor analysis typically incorporates more domain specific assumptions about the underlying structure and solves eigenvectors of a slightly different matrix.

Principal Component Analysis: Non-linear generalizations

- Linear PCA versus nonlinear Principal Manifolds for visualization of breast cancer microarray data:
  - (a) Configuration of nodes and 2D Principal Surface in the 3D PCA linear manifold. The dataset is curved and cannot be mapped adequately on a 2D principal plane;
  - (b) The distribution in the internal 2D non-linear principal surface coordinates (ELMap2D) together with an estimation of the density of points;
  - (c) The same as in (b), but for the linear 2D PCA manifold (PCA2D).
- The "basal" breast cancer subtype is visualized more adequately with ELMap2D and some features of the distribution become better resolved in comparison to PCA2D.

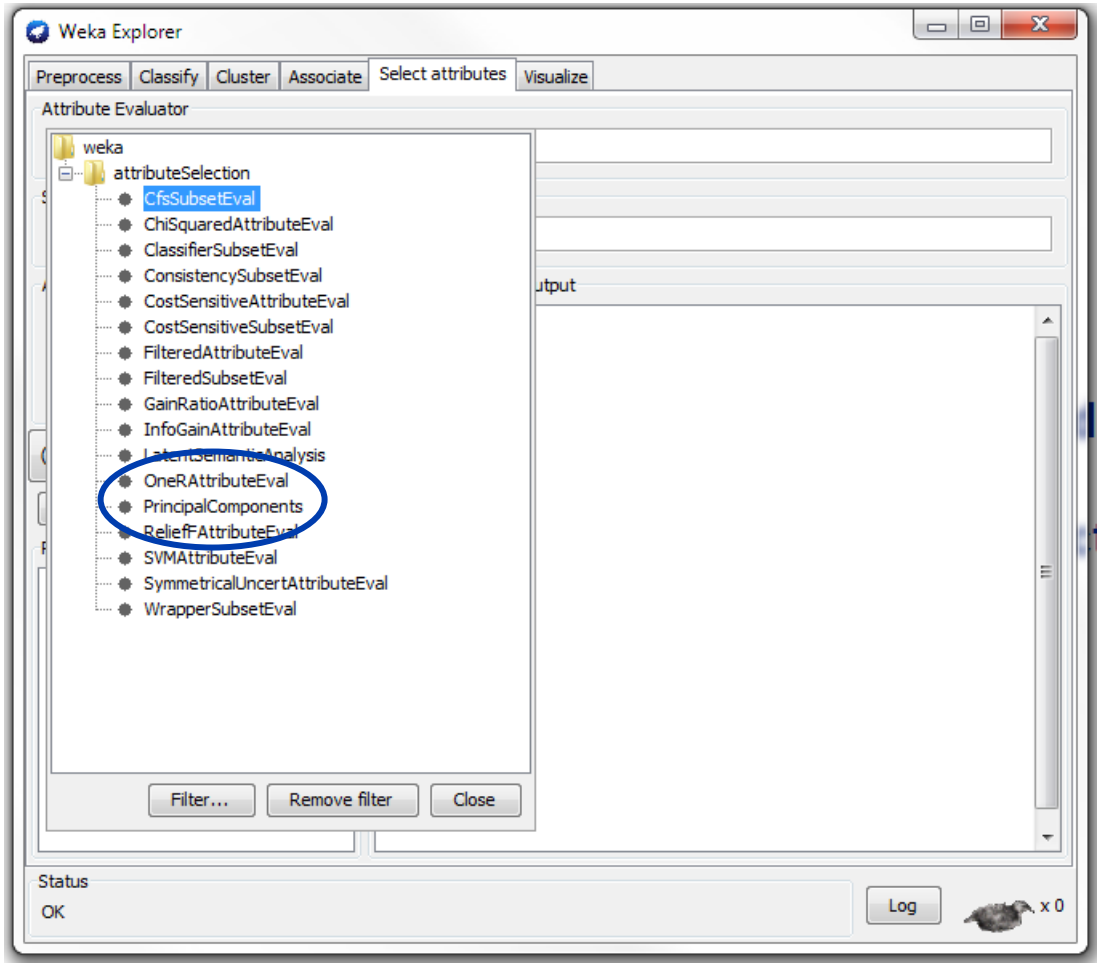


http://en.wikipedia.org/wiki/Principal_component_analysis

Principal Component Analysis in Weka

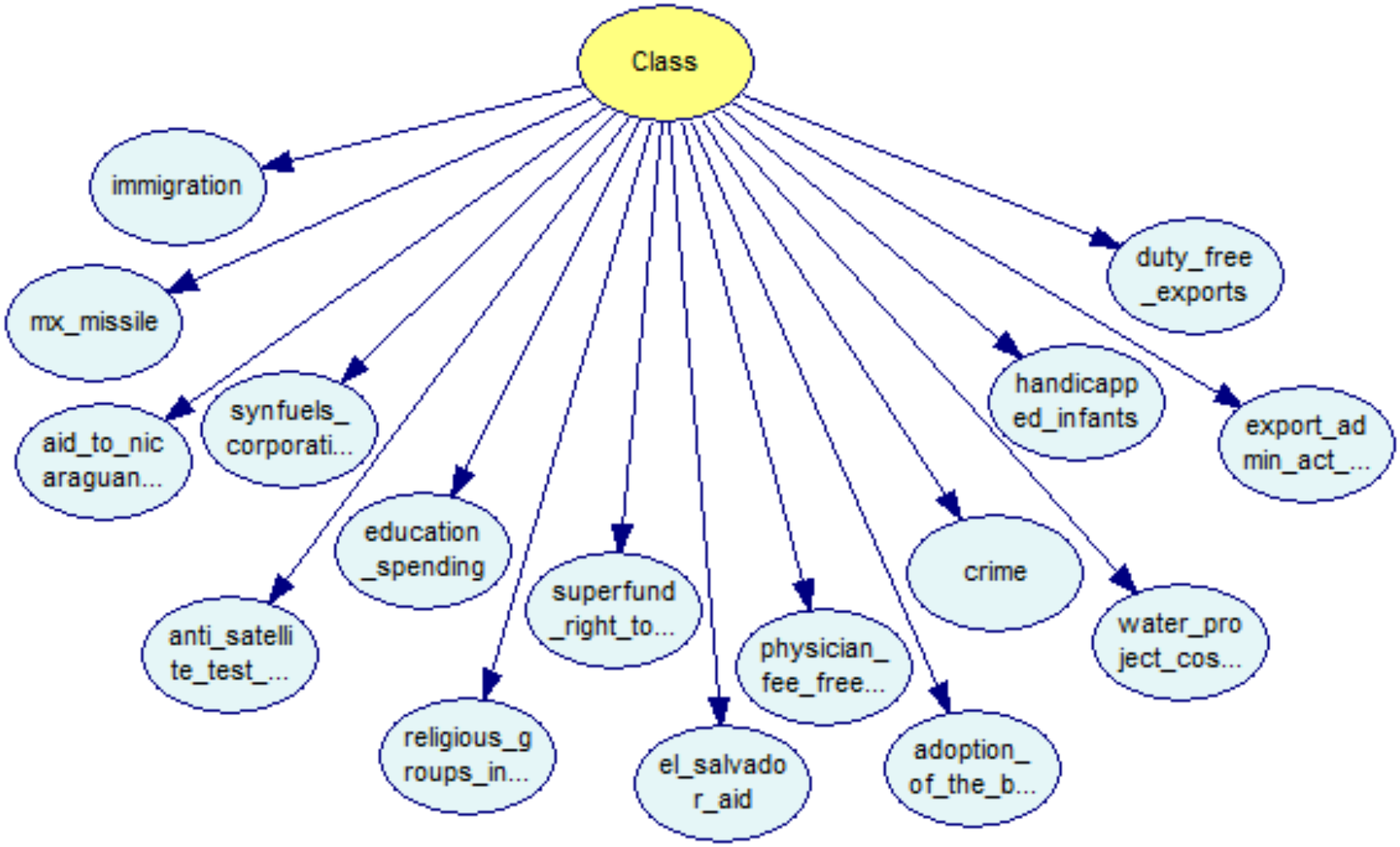
Use a data set with categorical class and numerical attributes, e.g., the Iris data.

The main application is in attribute selection.

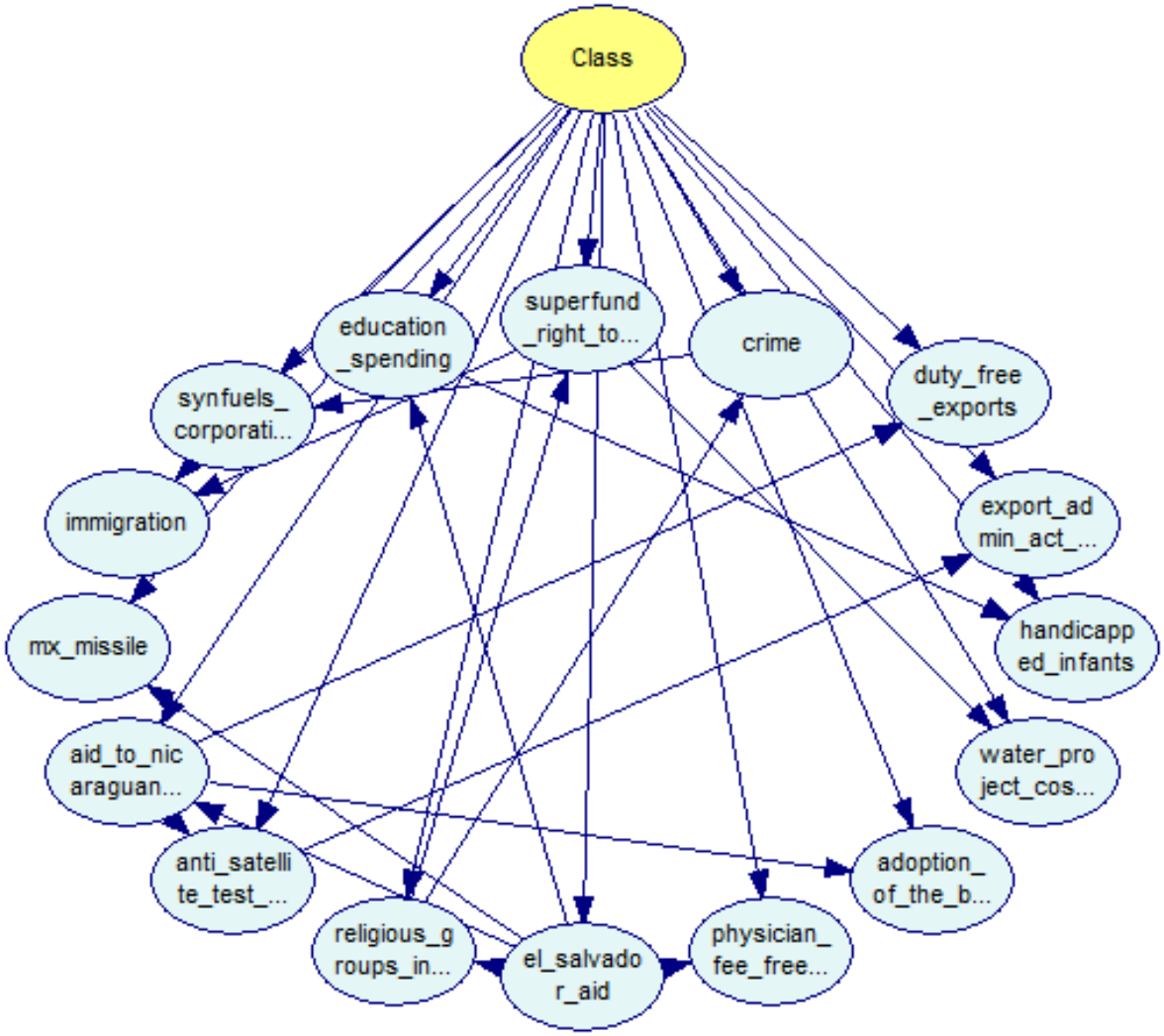


Bayesian network classifiers

Naïve Bayes models

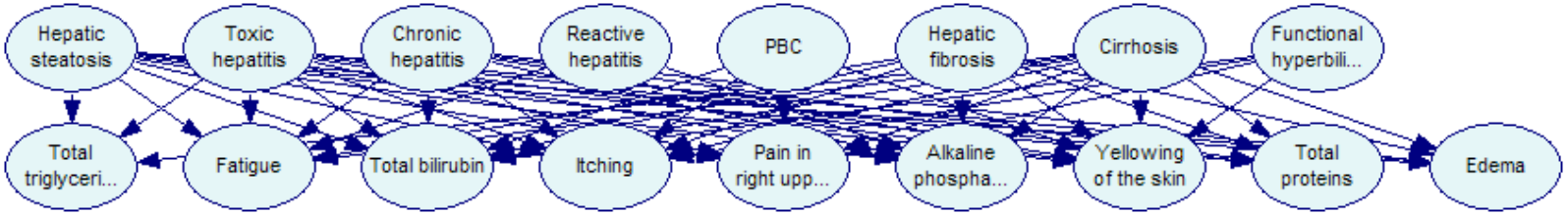


TAN (Tree Augmented Network) models

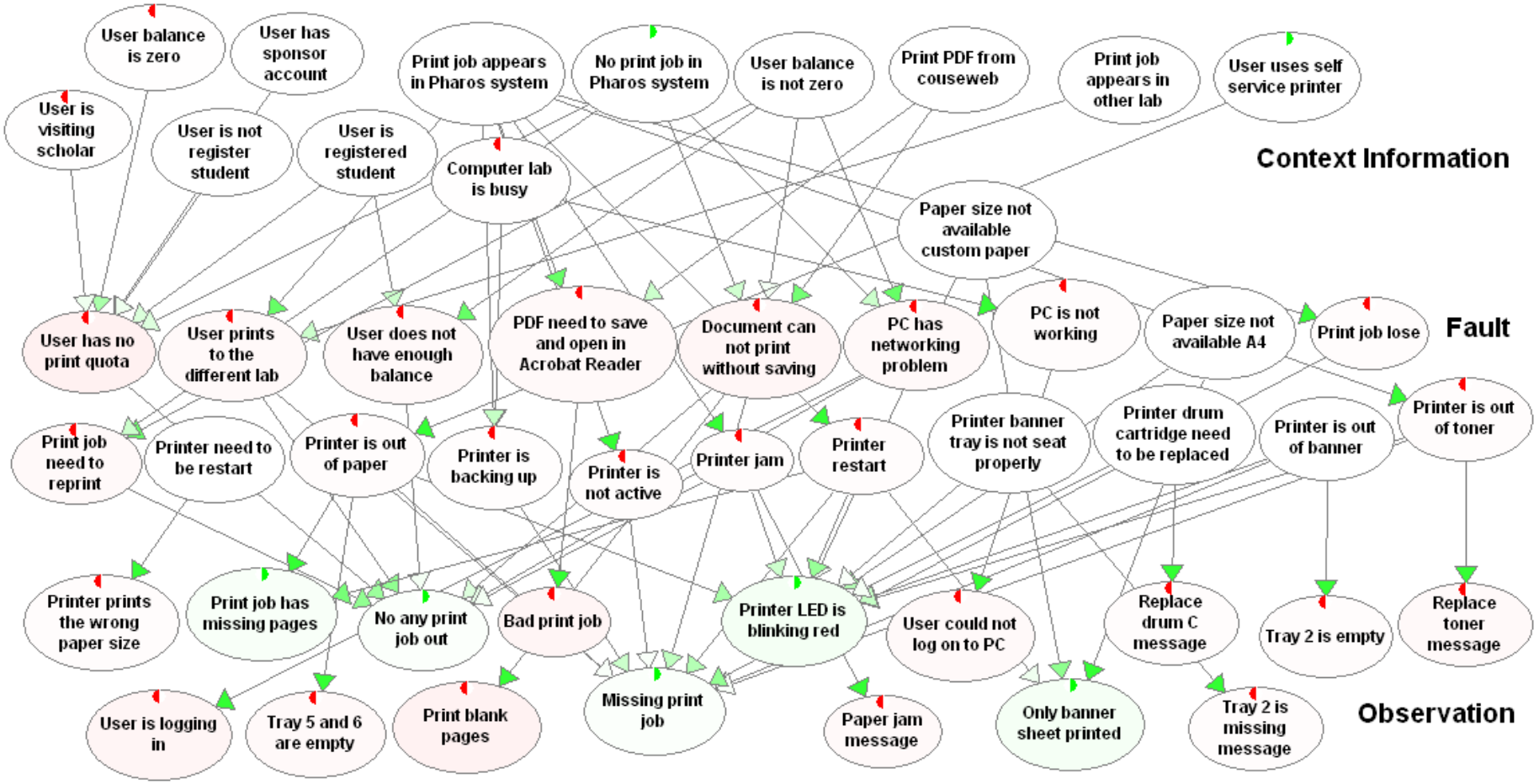


http://link.springer.com/referenceworkentry/10.1007%2F978-0-387-30164-8_850

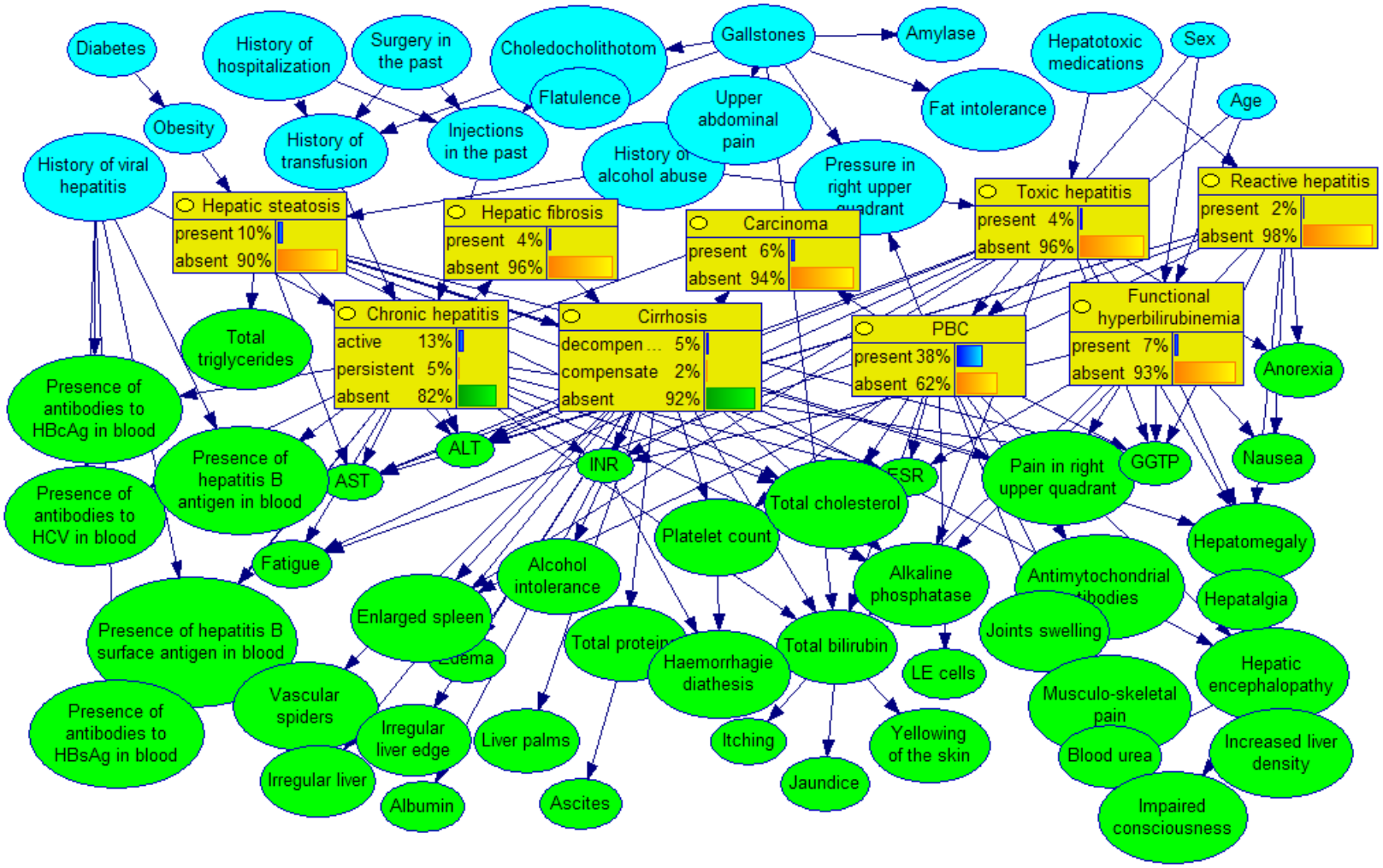
Bipartite models



BN3O Models


User balance is zero
User has sponsor account
Print job appears in Pharos system
No print job in Pharos system
User balance is not zero
Print PDF from couseweb
Print job appears in other lab
User uses self service printer
User is visiting scholar
User is not register student
User is registered student
Computer lab is busy
Context Information
Paper size not available custom paper
User has no print quota
User prints to the different lab
User does not have enough balance
PDF need to save and open in Acrobat Reader
Document can not print without saving
PC has networking problem
PC is not working
Paper size not available A4
Print job lose
Fault
Print job need to reprint
Printer need to be restart
Printer is out of paper
Printer is backing up
Printer is not active
Printer jam
Printer restart
Printer banner tray is not seat properly
Printer drum cartridge need to be replaced
Printer is out of banner
Printer is out of toner
Printer prints the wrong paper size
Print job has missing pages
No any print job out
Bad print job
Printer LED is blinking red
User could not log on to PC
Replace drum C message
Tray 2 is empty
Replace toner message
User is logging in
Tray 5 and 6 are empty
Print blank pages
Missing print job
Paper jam message
Only banner sheet printed
Tray 2 is missing message
Observation

Complete models



Is precision real or illusory?

- When getting the parameters from experts, we may well get better models when eliciting fewer parameters.
- When learning, the same may happen!

Ensemble learning

Definition

- Ensemble methods focus on improving prediction performance by using multiple models (as opposed to single models).
- The resulting performance is typically better than that of any single constituent model.

http://www.northpacificmusic.com/inner.landscape.html

Ensemble learning: Principles

- Ensembles combine multiple models with the expectation that this will form a better model.
- In other words, an ensemble is a technique for combining several weak learners in an attempt to produce a strong learner.
- The term *ensemble* is usually reserved for methods that generate multiple models using the same base learner.

Ensemble learning: Principles

- Ensembles tend to yield better results when there is a significant diversity among the models.
- It is a good idea, therefore, to promote diversity among the models in this approach.
- Interestingly, more randomness can produce a stronger ensemble than very deliberate algorithms.
- Using a variety of strong learning algorithms, however, has been shown to be more effective than using techniques that plainly promote diversity.

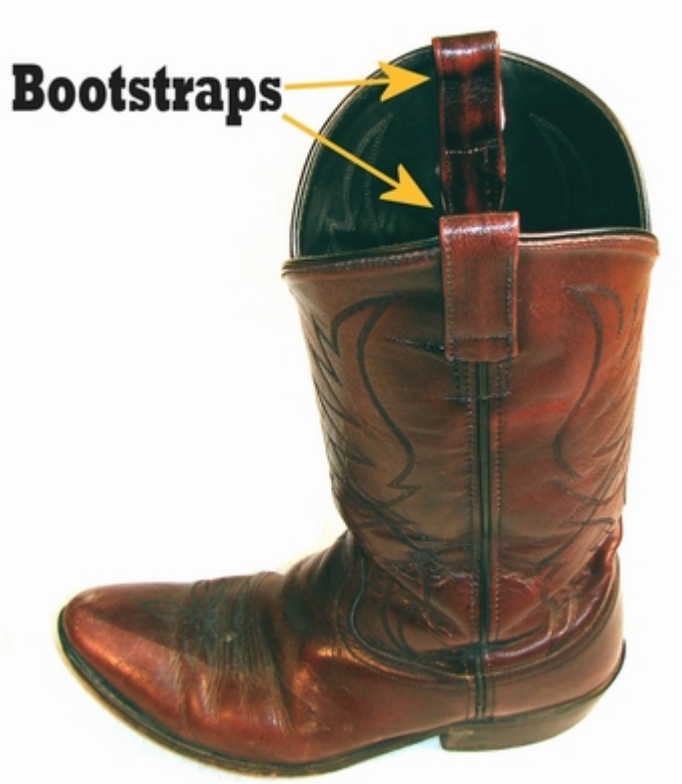
Types of ensemble learning algorithms

- Bayes optimal classifier
- Bootstrap aggregating (bagging)
- Random Forest
- Boosting
- Bayesian model averaging
- Bayesian model combination
- Bucket of models
- Stacking

Bootstrap aggregating (bagging)

- Bootstrap aggregating (typically called “bagging”) involves having each model in the ensemble vote with equal weight.
- In order to promote model variance, bagging trains each model in the ensemble using a randomly drawn subset of the training set.
- Typically applied to decision tree models but can be used with any type of model.

$$\hat{f}_{bag}(x) = \frac{1}{B}\sum_{b=1}^{B}\hat{f}^{*b}(x)$$



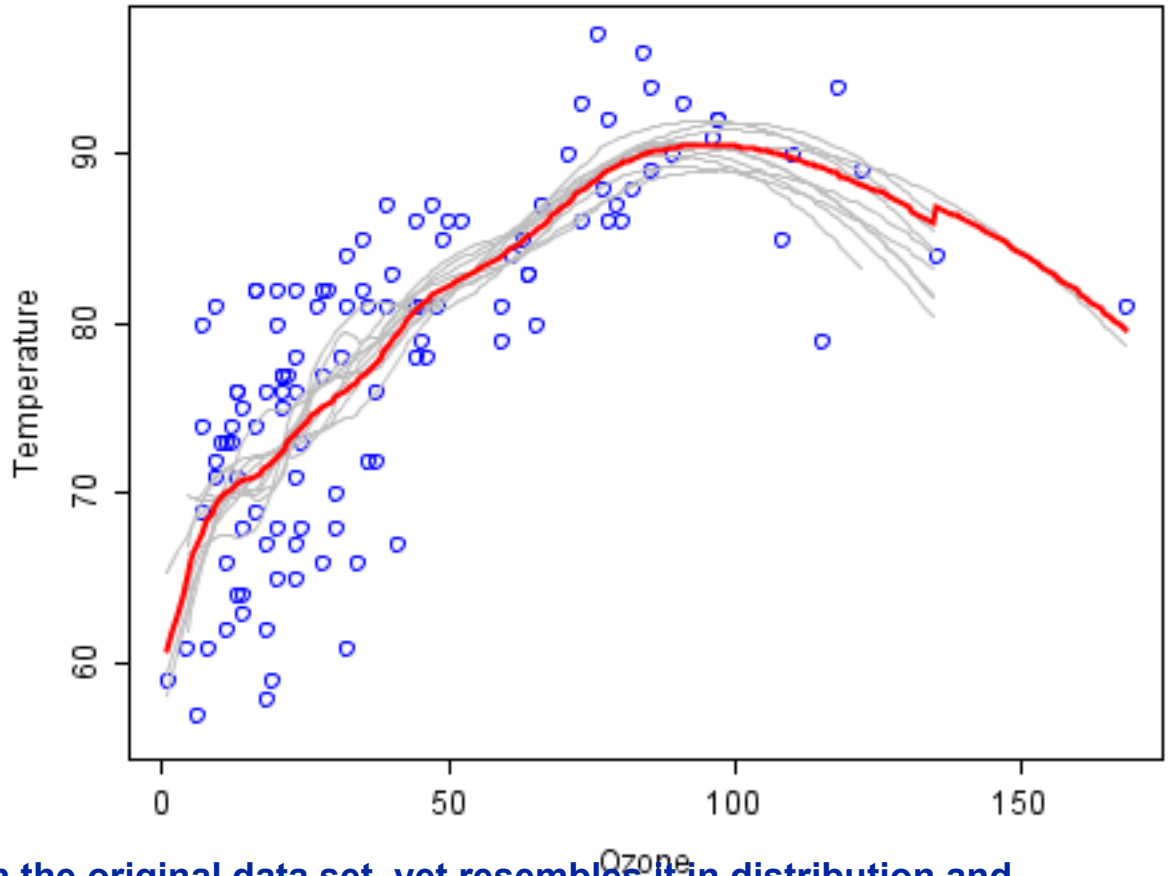
http://www.lemen.com/imageBootstrap1.html

Bootstrap aggregating (bagging)

- Given a standard training set *D* of size *n*, bagging generates *m* new training sets , each of size $n' < n$, by sampling examples from *D* uniformly and with replacement.
- If $n'=n$, then for large *n* the set can be expected to have the fraction $(1 - 1/e)$ (≈63.2%) of the unique examples of *D*, the rest being duplicates.
- This kind of sample is known as a bootstrap sample.
- The *m* models derived from the above *m* bootstrap samples are combined by averaging the output (for regression) or voting (for classification).
- Bagging leads to "improvements for unstable procedures" (Breiman, 1996), which include, for example, neural nets, classification and regression trees, and subset selection in linear regression (Breiman, 1994). On the other hand, it can mildly degrade the performance of stable methods such as K-nearest neighbor (Breiman, 1996).

Bootstrap aggregating (bagging): Example

- To illustrate the basic principles of bagging, below is an analysis on the relationship between ozone and temperature (data from Rousseeuw and Leroy, 1986).
- The relationship between temperature and ozone in this data set is apparently non-linear, based on the scatter plot. To mathematically describe this relationship, instead of building a single model from the complete data set, models from 100 bootstrap samples of the data were learned.



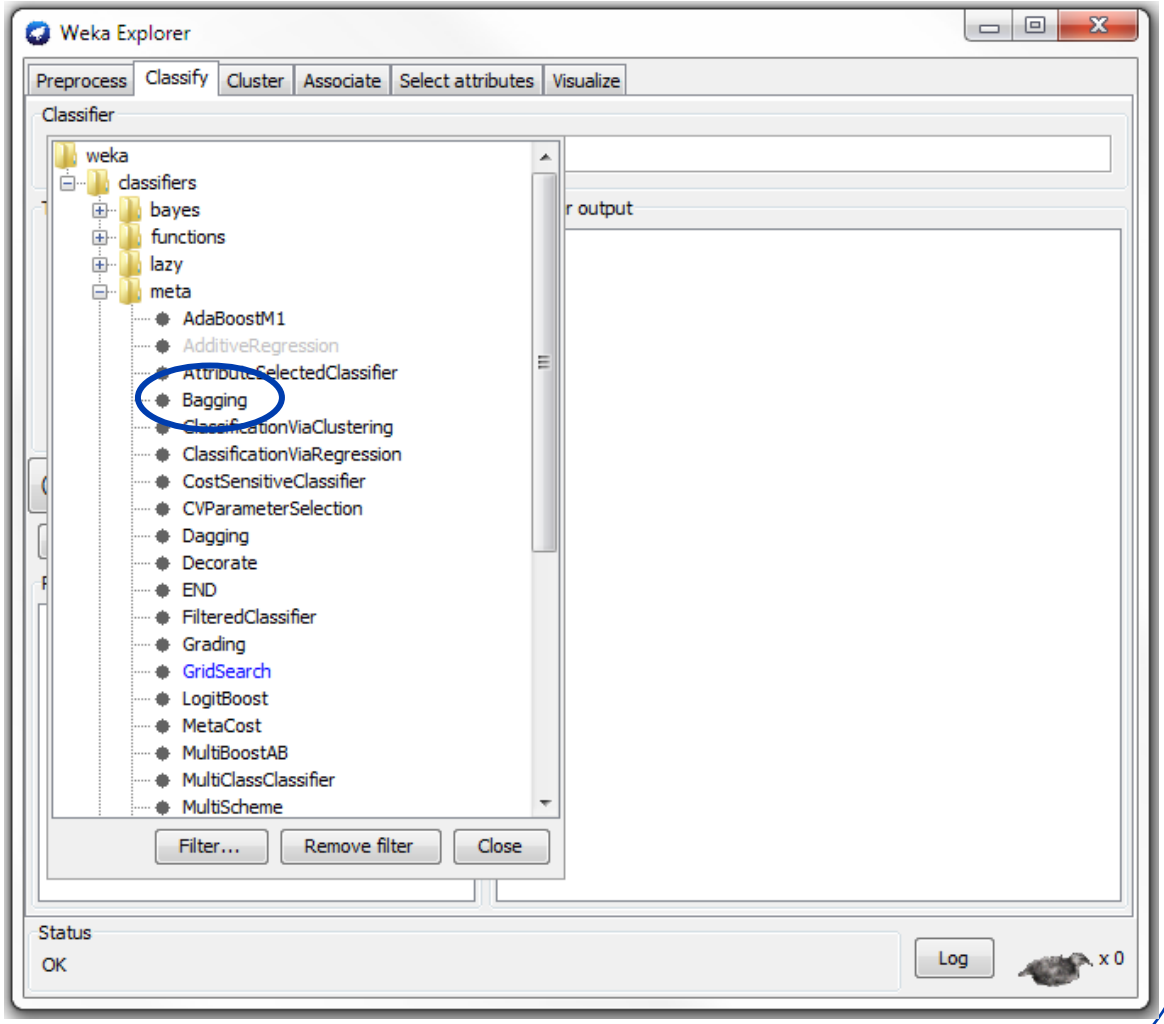
- Each sample is different from the original data set, yet resembles it in distribution and variability. Predictions from these 100 models were then made across the range of the data. The first 10 predicted smooth fits appear as grey lines in the figure below.
- But taking the average of 100 models, we arrive at one bagged predictor (red line). The mean in the average is more stable and there is less overfit.

http://en.wikipedia.org/wiki/Bootstrap_aggregating

Bagging in Weka

Use a data set with categorical class and numerical attributes, e.g., the Iris data.

The main application is in classification.



Random Forest

- Random Forests are an ensemble learning method for classification and regression
- It combines multiple individual classifiers by means of bagging
- Overcomes the problem of overfitting decision trees
- Reduce correlation between trees, by introducing randomness

$$\hat{f}_{RandomForest}(x) = \frac{1}{B}\sum_{b=1}^{B}\hat{f}^{*b}(x)$$

Random Forest (cont'd)

- As in bagging, we build a number of models on bootstrapped training samples.
- But when building these models, each time considering a random selection of m predictors is chosen as split candidates from the full set of p predictors. The split is allowed to use only one of those m predictors.
- A fresh selection of m predictors is taken at each split, and typically we choose $m \approx \sqrt{p}$ — that is, the number of predictors considered at each split is approximately equal to the square root of the total number of predictors.
- Ensemble trees (i. e. the random forest) vote on categories by majority

Boosting

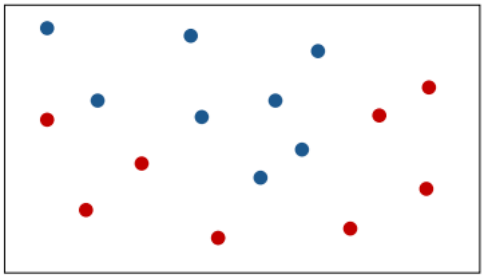
- Boosting involves incrementally building an ensemble by training each new model instance to emphasize the training instances that previous models mis-classified.
- In some cases, boosting has been shown to yield better accuracy than bagging.

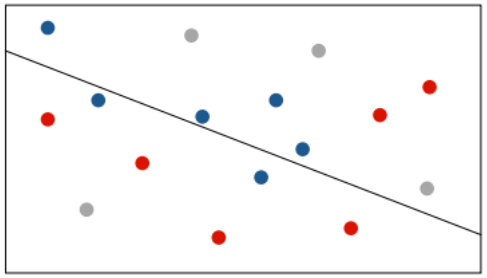
$$\hat{f}(x) = \sum_{b=1}^{B} \lambda \hat{f}^{b}(x)$$

Boosting

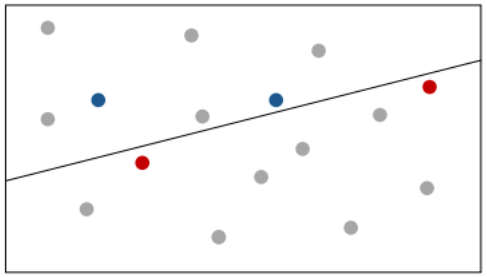
- While boosting is not algorithmically constrained, most boosting algorithms consist of iteratively learning weak classifiers with respect to a distribution and adding them to a final strong classifier.
- When they are added, they are typically weighted in some way that is usually related to the weak learners' accuracy.
- After a weak learner is added, the data is reweighted: examples that are misclassified gain weight and examples that are classified correctly lose weight (some boosting algorithms actually decrease the weight of repeatedly misclassified examples, e.g., boost by majority and BrownBoost). Thus, future weak learners focus more on the examples that previous weak learners misclassified.

Boosting

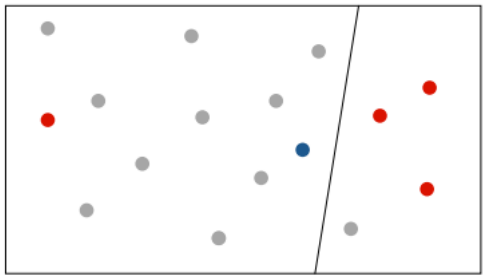




C_1

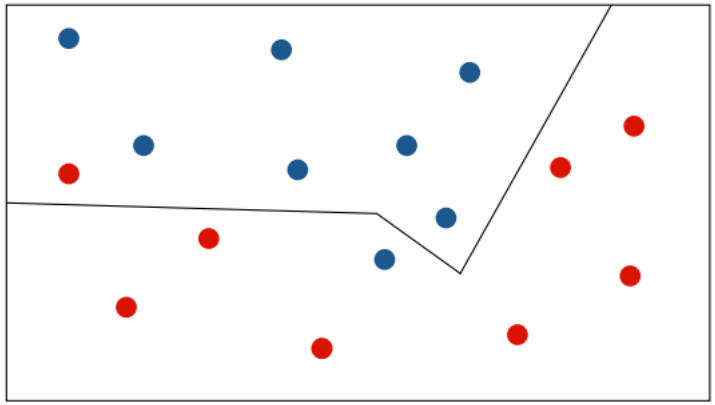


C_2



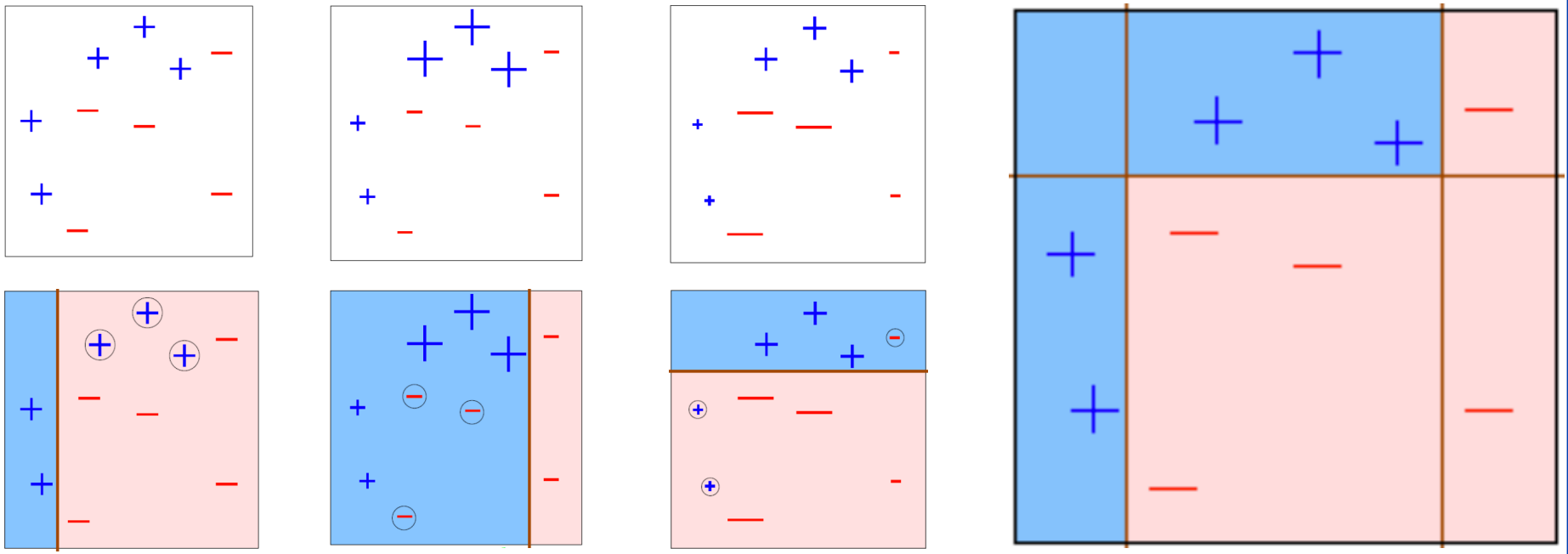
C_3

$$C_{boosting} = C_1 + C_2 + C_3$$



http://www.is.uni-freiburg.de/ressourcen/business-analytics/10_ensemblelearning.pdf

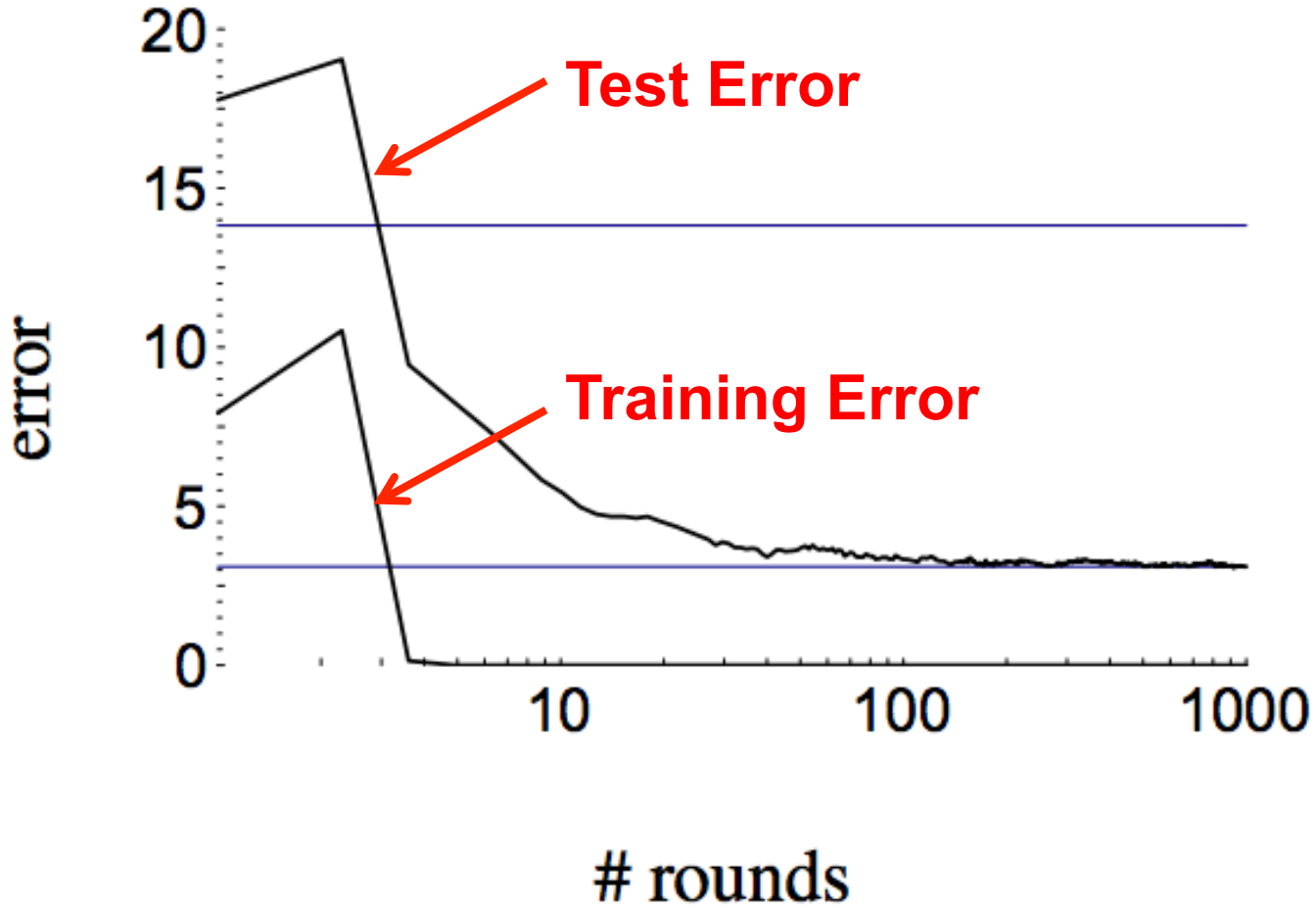
Boosting: AdaBoost



- Sequentially adds new component classifiers
- Each trained on reweighted training examples

http://www.cs.cmu.edu/~epxing/Class/10701/slides/lecture11-boosting.pdf

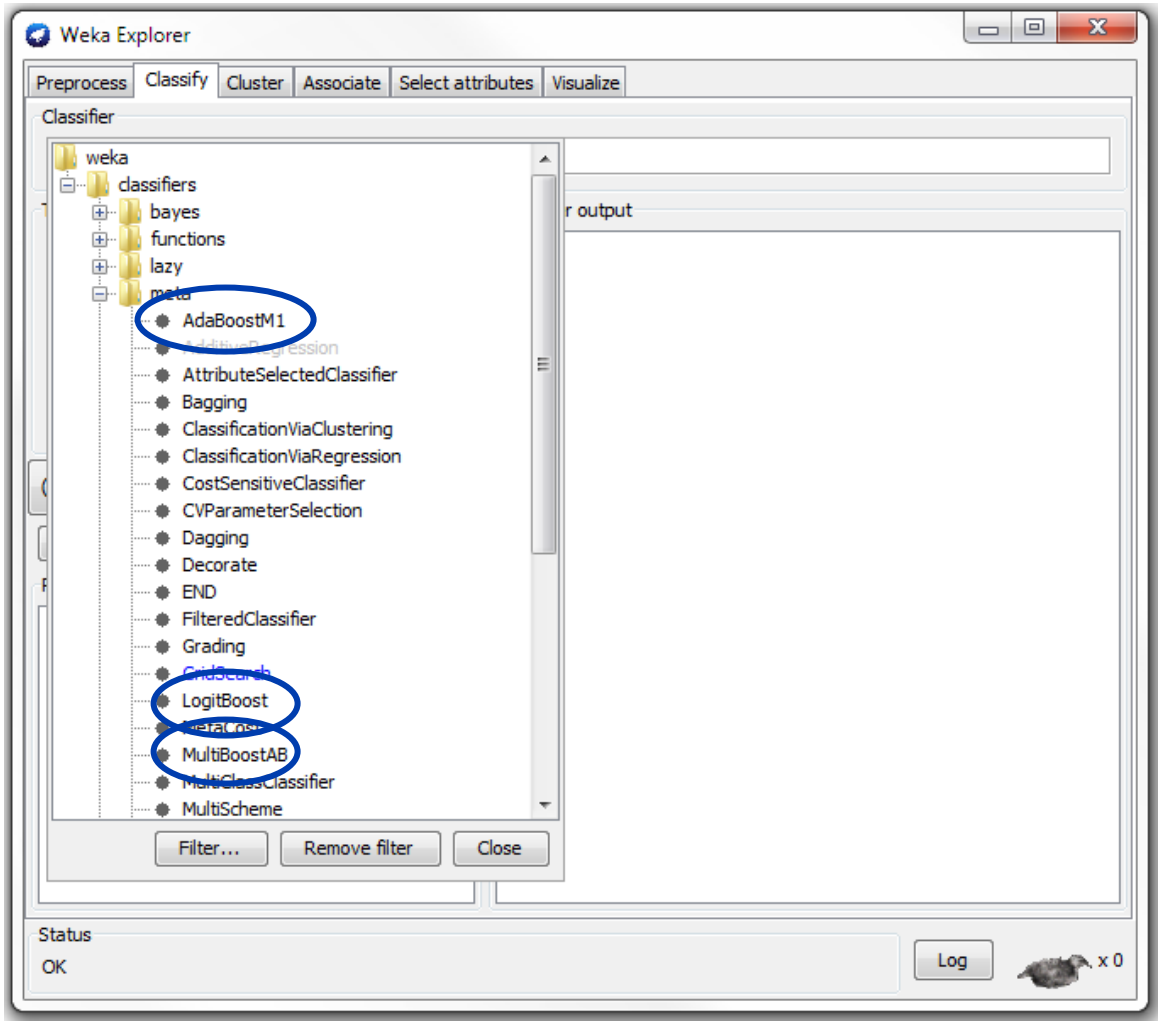
Boosting



Schapire, R. E., Freund, Y., Bartlett, P., & Lee, W. S. (1998). Boosting the margin: A new explanation for the effectiveness of voting methods. Annals of statistics, 1651-1686.

Boosting in Weka

AdaBoostM1, LogitBoost, MultiBoost AB algorithms



Bayesian Model Averaging

- Approximating the theoretically optimal classifier by sampling hypotheses from the hypothesis space and combining them using Bayes theorem.
- Unlike the Bayes optimal classifier, Bayesian model averaging can be practically implemented.

Bayesian Model Averaging

- Sampling using a Monte Carlo sampling technique such as MCMC.
- Under certain circumstances, its expected error bounded to be at most twice the expected error of the Bayes optimal classifier
- Tendency to promote over-fitting
  - Does not perform as well empirically as simpler ensemble techniques such as bagging.

Concluding Remarks

- There are many techniques based on probability theory.
- This presentation was meant to touch upon the most important/popular ones, without going into details (which you can learn in classes like Data Mining or Machine Learning).
- You can try out each of these techniques using software like Weka or GeNIe.