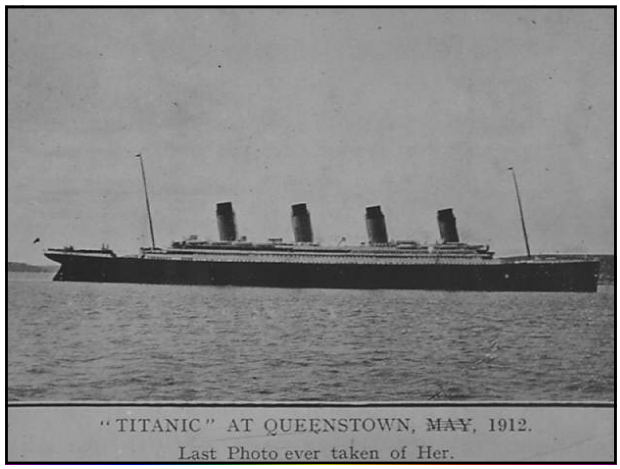
An Introduction to Pattern Classification

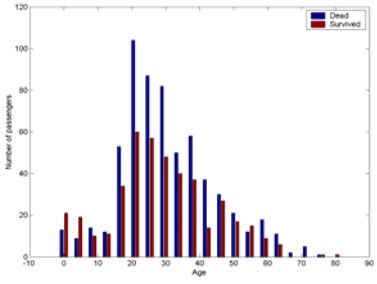
Elad Yom-Tov



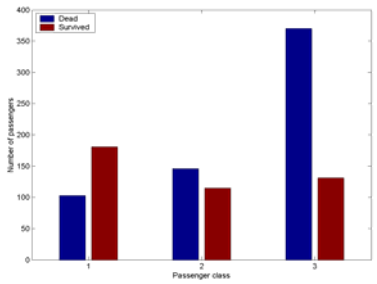
Question

Given the age, gender, passenger class and survival data for 75% of the passengers, can we predict who survived among the remaining 25% of passengers?

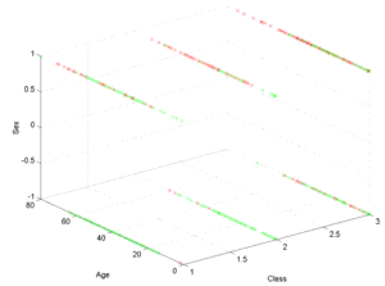
Was age a good predictor?



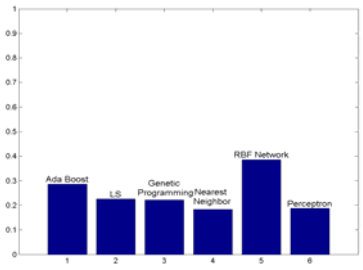
Was class a good predictor?



Maybe all three?



Maybe all three?
Pattern Classification Algorithms



What is a pattern?

"A pattern is the opposite of chaos; it is an entity, vaguely defined, that could be given a name."
(Watanabe)

A Definition of Pattern Recognition Problems

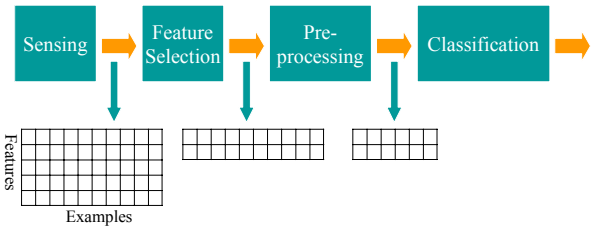
Given a feature vector X (The input vector) and a label Y, find a mapping Y=f(X).

This mapping should give a minimal error in labeling.

The problem: Build a machine to classify patterns

- Speech recognition
- Optical character recognition
- DNA sequence identification
- Finger and face identification
- etc...

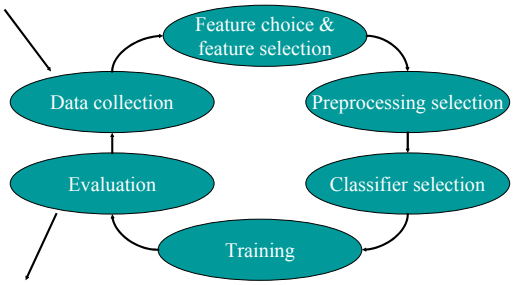
The Classification Procedure



Two main modes of learning

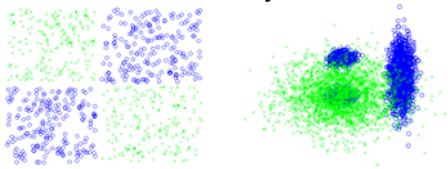
- **Supervised learning**: A "teacher" provides the category for each pattern in the training set.
- **Unsupervised learning**: The system generates groupings of the input patterns without any targets.

The Design Cycle



Issues in Data Collection

- How many examples do we need to adequately represent the data and to train and test the classifier?
- Which data should we try to collect?



Feature Choice and Feature Selection

- Depends on the problem
- Can extract many and select few
- Too many cause generalization problems and increase computational complexity
- Too few will make it impossible to separate the samples ("Ugly-duckling theory")
- Based on apriory knowledge
- Prefer independent and invariant features

Classifier and Preprocessing algorithm selection

- Model based or model-free?
- Based on apriory knowledge
- No single best algorithm (Though some come close to it…)

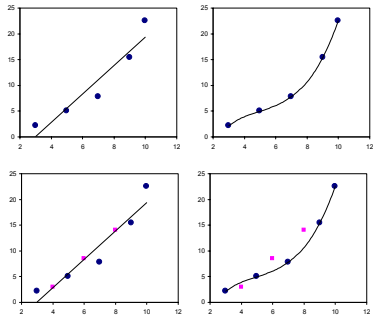
Training

- Separate the data into train and test data.
- Train using some of the data, and test on the rest.

Error estimation

- Measure the error rate of the classification procedure
- Try to obtain an unbiased estimation of the error

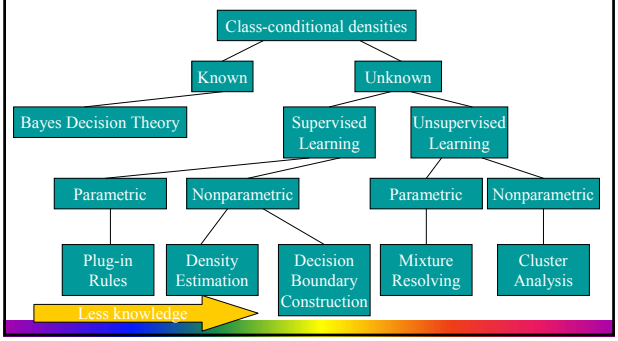
Classifier Complexity



Computational Complexity

- The tradeoff between computational ease and performance.
- How does the procedure scale as a function of the number of features and patterns?

General Approaches to Pattern Recognition



Related Fields

- Statistics
- Optimization
- Signal Processing
- ….

Dimensionality Reduction

The Curse of Dimensionality

An example by Trunk (1979)

Is more knowledge always useful?

Consider a Gaussian distribution with identity covariance matrices and equal prior probabilities. The mean vectors are:

$$m^1 = \left(+1, +\frac{1}{\sqrt{2}}, +\frac{1}{\sqrt{3}}, \cdots, +\frac{1}{\sqrt{d}}\right)$$

$$m^2 = \left(-1, -\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{3}}, \cdots, -\frac{1}{\sqrt{d}}\right)$$

There is only one parameter in this distribution:

The Curse of Dimensionality (2)

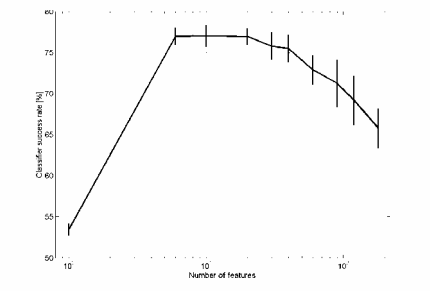
- If m is known, $\lim_{d\to\infty} P_e(d) = 0$
- If m is unknown and estimated using ML, $\lim_{d\to\infty} P_e(n,d) = \frac{1}{2}$

Why Try to Reduce the Problem Dimension?

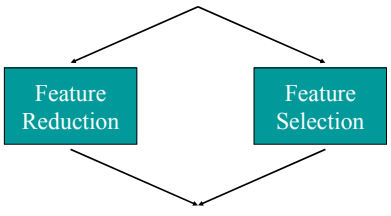
- "The curse of dimensionality"
- Improve generalization
- Decrease computational complexity
- Reduce the number of examples needed

BUT: Too few features may reduce the descrimination between classes ("Ugly duckling theorm").

An example



Two Approaches for Dimensionality Reduction



Feature Reduction

Reshape the data at a lower dimension.
Linear transformations are defined using:

$$Y = H \cdot X$$

$[Y] = d \times N,\ [X] = m \times N,\ [H] = d \times m$
$d \le m$

Methods for Feature Reduction

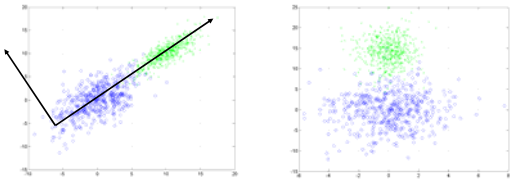
Linear methods:

- Principal component analysis (PCA, KLE)
- Independent component analysis (ICA)
- Fisher linear discriminant

Non-linear methods:

- Nonlinear component analysis (NLCA)
- Kernel PCA

Principal Component Analysis



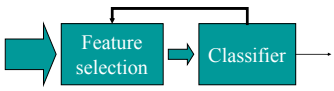
Feature Selection

Exhaustive search is computationally prohibitive, except for a small number of dimensions.
There are 2^n-1 possible combinations.

Basically it is an optimization problem, where the classification error is the function to be minimized.

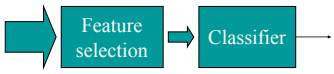
Feature Selection Methods

- Wrapper methods



- Embedded methods

- Filter methods

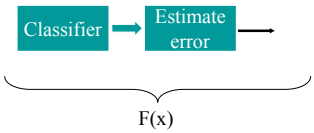


Methods for Feature Selection

- Exhaustive search
- Best individual Feature
- Sequential Forward Selection
- Sequential Backward Selection
- Optimization: Genetic algorithms / Annealing
- Information-theoretic methods

Why is feature selection an optimization task?

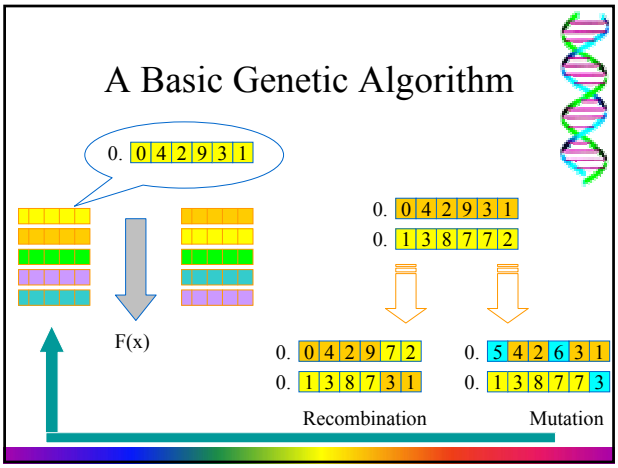
0100010100111
1011100000010
0110111111000
Feature vector



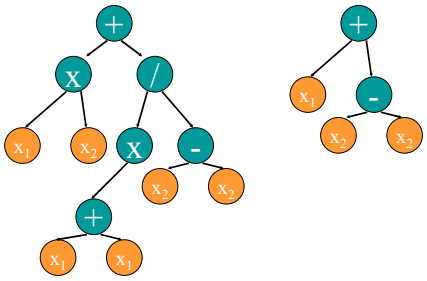
Genetic Algorithms: Basic Assumptions

- Available resources cannot support all individuals in the populations, thus there is a constant battle between individuals for resources.
- There are no two completely identical individuals.
- Specific genes that help their owners to be better adapted to the environment will multiply and spread in the population, because they help their owners survive.

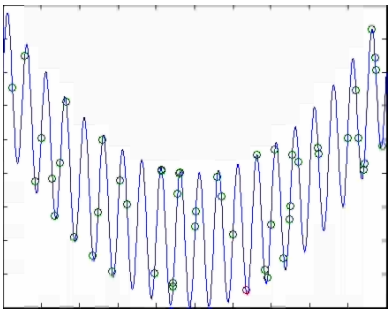
A Basic Genetic Algorithm



Genetic Programming



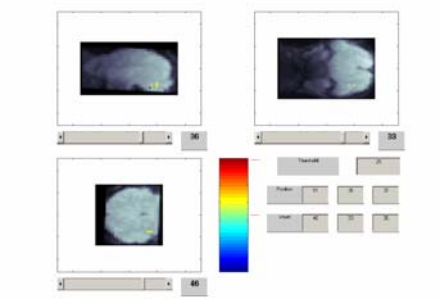
An Example



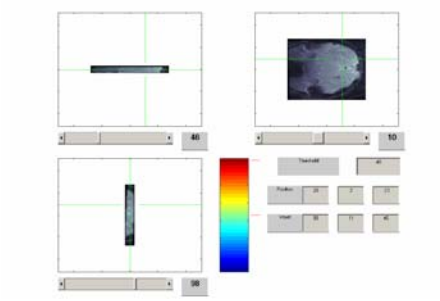
The Culling Genetic Algorithm

1. Randomly partition a set of N features into N_G groups, where N_F is the number of features to be used for classification.
2. Classify the examples using the feature groups and order them according to the classification error.
3. Discard a predetermined percentage of the groups that have the lowest score, and build the same number of new groups by randomly selecting half the features from the remaining groups and combining them.
4. Repeat steps 2-3 for a predetermined number of iterations.

An Example: Discrimination Between Men and Buildings



An Example: Discrimination Between Men and Women



Preprocessing

Approaches to Preprocessing

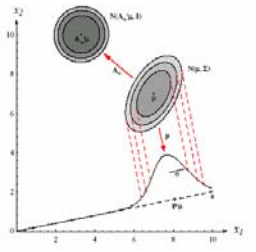
- Reshaping
- Removing superfluous data
- Clustering

Whitening Transform

$A_W = \Phi \Lambda^{-\frac{1}{2}}$

Φ – *Matrix whose columns are eigenvectors of A*

Λ – *Diagonal matrix of corresponding eigenvalues*

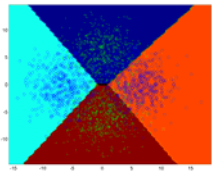


Nearest Neighbor Editing Algorithm

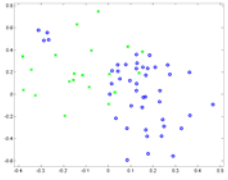
1. Construct the Voronoi diagram of the patterns.
2. For each pattern:
   a. Find the labels of its' neighbors
   b. Delete it if they are all of the same class as the test pattern.
3. Return the remaining patterns.

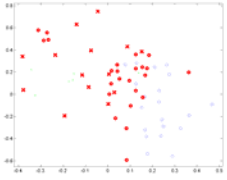
Voronoi Diagram

The partitioning of a plane with n points into n convex polygons such that each polygon contains exactly one point and every point in a given polygon is closer to its central point than to any other.

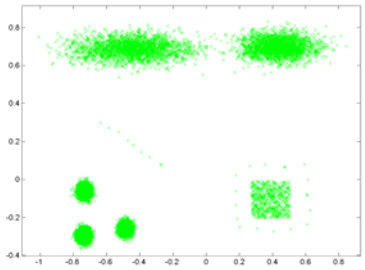


Nearest Neighbor Editing: An Example





What Are Clusters?



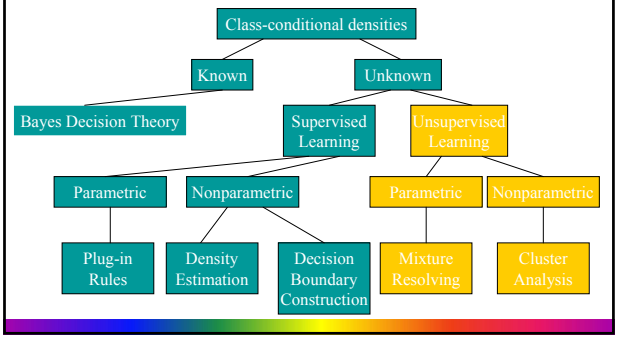
What Are Clusters? (II)

- Patterns within a cluster are more similar to each other than are patterns in other clusters.
- A cluster is a volume of high-density points separated from other clusters by a relatively low density volumes.

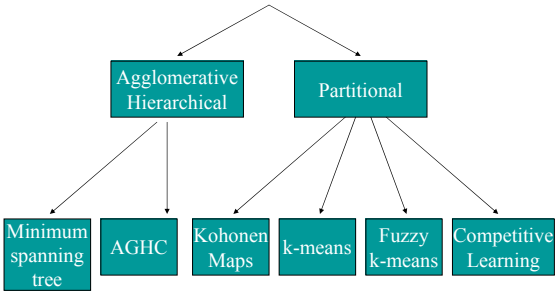
WARNING

- There is no “best” clustering method.
- A clustering algorithm will always find clusters, even if there are non.

General Approaches to Pattern Recognition



Main Clustering Techniques



The K-means Algorithm

1. Partition the data randomly into K partitions.
2. Compute the centers for each of the K partitions using the data points associated with each partition.
3. Reassign the each of the data points to one of the K centers according to a metric.
4. Repeat steps 2-3 until convergence.

The Agglomerative Hierarchical Clustering Algorithm

1. Each data point is a cluster.
2. Find the two nearest* clusters and merge them.
3. Repeat step 2 until a convergence criterion is met.

Classification

Is there an optimal decision rule?

Bayes decision rule:

Minimize the expected loss function

$$R(\omega_i \mid x) = \sum_{j=1}^{c} L(\omega_i, \omega_j) P(\omega_j \mid x)$$

L is the loss function for deciding on the wrong class.

Bayes Rule (2)

Given the 0/1 loss,

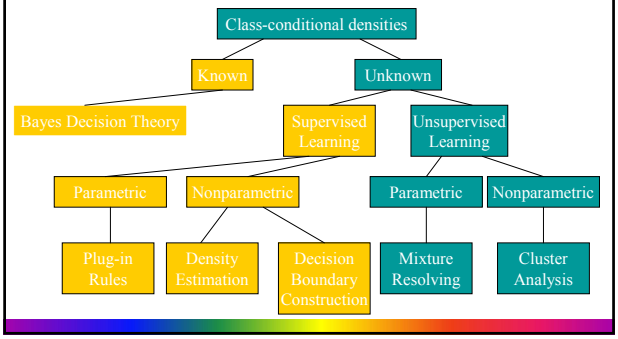
$$L(\omega_i, \omega_j) = \begin{cases} 0, & i = j \\ 1, & i \neq j \end{cases}$$

Bayes decision rule is simplified to the **maximum aposteriory (MAP) rule:**

Assign the input pattern *x* to class w_i if:

$$P(\omega_i \mid x) > P(\omega_j \mid x) \quad for\ all\ j \neq i$$

General Approaches to Pattern Recognition

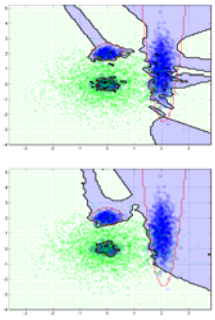


Another Division of Classification Algorithms

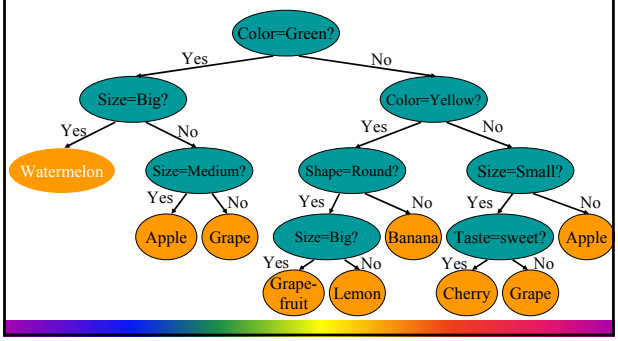
- "Heuristic"
- Plug-in rules
- Optimization
- Boosting

"Heuristic" Algorithms: Nearest Neighbors

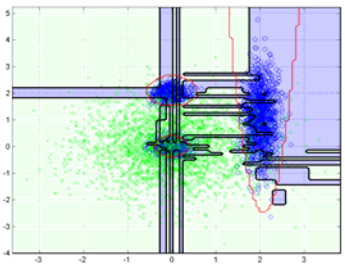
- Computationally intensive
- Requires large memory
- Simple
- Usually quite good results



"Heuristic" Algorithms: Tree-based Algorithms



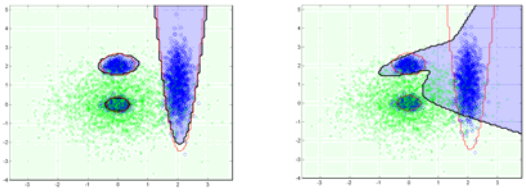
Tree-based Algorithms (2)



Plug-in Rules

- Maximum Likelihood
- Expectation-Maximization (EM)

Plug-in Rules: The Expectation-Maximization Algorithm



Optimization Algorithms

- Least-squares (LS)
- Genetic Programming
- Neural Networks
- Radial-Basis Function (RBF) Networks
- Support-Vector Machines (SVMs)
- … and more

Least-Squares

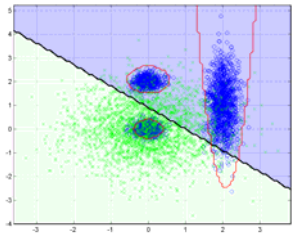
A matrix of measurements: $P^T \cdot w = T^T$

P_{DxN}, T_{1xN}, w_{Dx1}

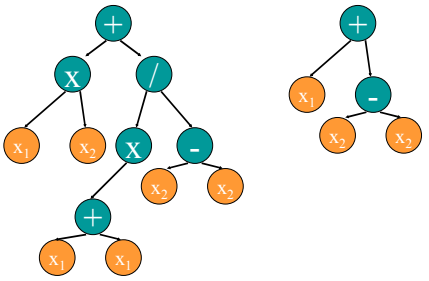
Moore-Penrose Pseudo-inverse: $w = (P \cdot P^T)^{-1} \cdot P \cdot T^T$

Predicted targets: $\hat{T} = w^T \cdot P$

Least-Squares (II)

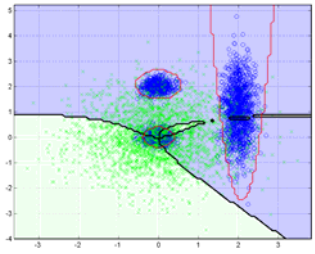


Genetic Programming

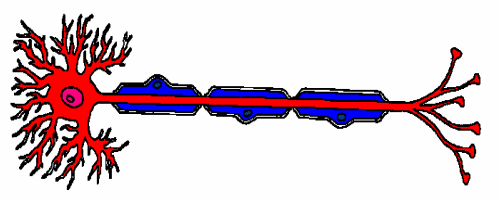


Genetic Programming

$$-\frac{x_2^5 \cdot (x_1 + x_2 - x_1/x_2)}{(x_1 x_2 + x_2/x_1 - x_1) \cdot x_1}$$



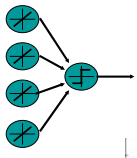
Neural Networks



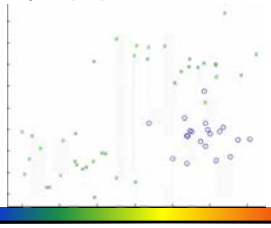
Training Methods

- Single neuron - Perceptron
- Network - Backpropagation

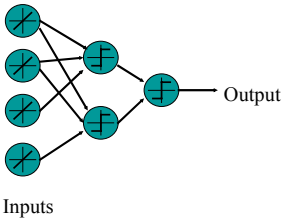
The Perceptron



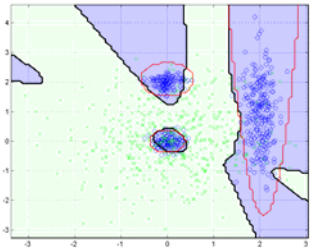
Stochastic training rule:
1. Normalize patterns so that: x=x*y
2. Choose a random training pattern.
3. If it is missclassified, a=a-y.
4. Repeat 2-4 until all training patterns are correctly classified.



A Feedforward Neural Network



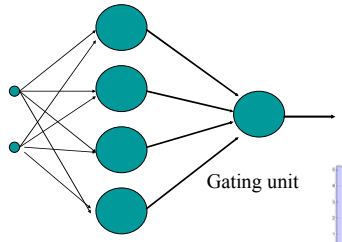
Neural networks (2)



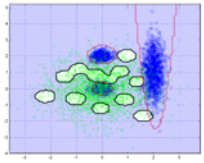
Drawbacks of Neural Networks

- Difficult to set an architecture.
- Several parameters to set.
- “Black box”

Radial-Basis Function Networks



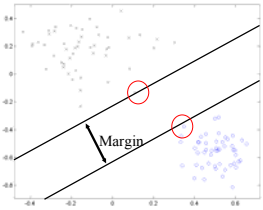
Hidden units (Radial basis)



Support-Vector Machines

Observation 1: Not all patterns have to be used for finding a separating hyperplane.

Observation 2: At a sufficiently high dimension, patterns are orthogonal to each other.



Support-Vector Machines (2)

Transform the patterns using a kernel: $y_k = \varphi(x_k)$

A separating hyperplane is then: $g(y) = a^T y$

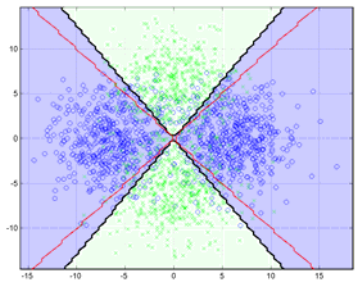
This hyperplane ensures that: $z_k g(y_k) \geq 1 \qquad k = 1,\ldots n$

$z_k = \{-1, +1\}$

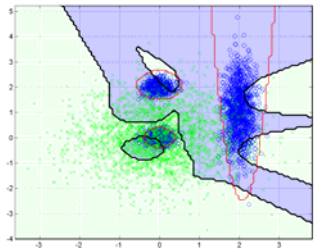
Thus the problem is to find the coefficients $\boldsymbol{a}$.

Some methods for finding the coefficients are Quadratic Programming (QP) and Perceptron.

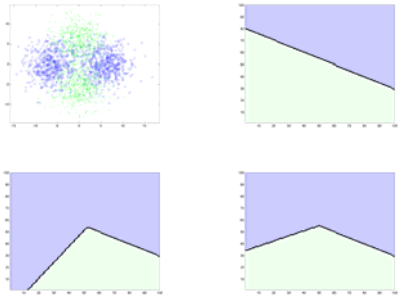
Support-Vector Machines (3)



Support-Vector Machines (4)



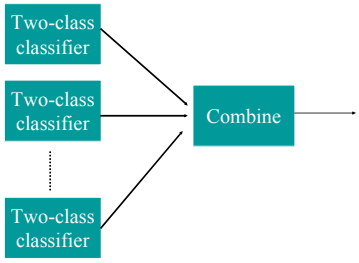
Boosting: AdaBoost



Multiple Class Problems

- Some classifiers can easily handle multi-class problem (e.g.: LS, Nearest Neighbor, etc).
- It is possible to combine several two-class classifiers into multi-class classifiers….

Combining two-class classifiers into multi-class classifiers



Multiclass: One Against All

		Classifier outputs		
Target class		Classifier 1	Classifier 2	Classifier 3
	Class 1	0	1	1
	Class 2	1	0	1
	Class 3	1	1	0
	Class 4	1	1	1

Multiclass: Coding Matrices

	Classifier outputs							
Target class		1	2	3	4	5	6	7
	1	1	0	0	1	1	0	1
	2	0	1	0	1	0	1	1
	3	0	0	1	0	1	1	1

One-Class Classifiers

I know it when I see it

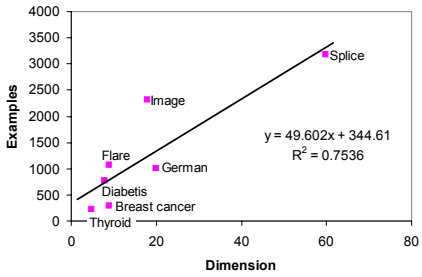
Possible similarity measures:

- Gaussian probability
- Probability of a Markov sequence

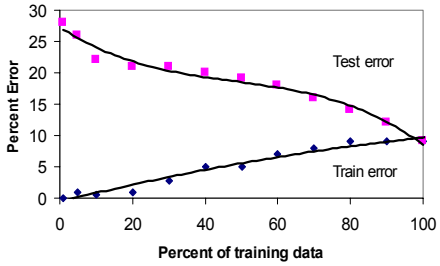
Some Rules of Thumb of Classification

- Ratio of the number of training examples to the number of features should be at least 10:1
- Ratio of the number of training examples to the number of unknown parameters should be at least 10:1 (Bernie's rule).
- Do not over-train on the train-set. Use appropriate error-estimation methods.

Ratio Training Samples to Problem Dimension



Why Not Over-Train?



Error Estimation

Error Estimation Methods

- Resubstitution: Use all the data for training and testing.
- Holdout: Use part of the data for training and the other part for testing.
- Cross-validation: Divide the data into N subsets. Train on (N-1) subsets and test on the N-th subset.

What to do in practice

1. Estimate the error using an error estimation method.
2. Build a classifier using all the data.
3. The error for classification of new patterns will be that estimated in step 1.