High Performance Computing and Networking Institute
National Research Council, Italy

Classification methods for microarray gene expression data analysis

Mario Rosario Guarracino
December 18, 2007

Agenda

- Problem definition and characteristics
- Supervised learning techniques:
  - Support Vector Machines
  - Generalized Eigencalue Classification
  - Other methods
- The kernel trick
- Error estimation:
  - Holdout
  - Random sampling
  - k-fold cross validation
  - Leave one out
  - P-value
  - ROC curves
- Research directions & conclusion

Introduction

- *Supervised learning* refers to the capability of a system to learn from experiments (*training set*), for which the outcome is known (health/disease, different diseases, …).
- The trained system is able to provide an answer (*output*) for each new question (*input*).
- *Supervised* means the desired output for the training set is provided by an external teacher.
- In case of two outcomes (*binary classification),* supervised learning provides very successful models.

Microarray applications

- Breast cancer: *BRCA1* vs. *BRCA2* and sporadic mutations,
  - I. Hedenfalk *et al*, *NEJM,* 2001. ***(22 patients, 3226 genes)***
- Prostate cancer: prediction of patient outcome after prostatectomy,
  - Singh D*. et al, Cancer Cell,* 2002. ***(136 patients, 12600 genes)***
- Malignant gliomas survival: gene expression vs. histological classification,
  - C. Nutt *et al*, *Cancer Res.*, 2003. (***50 patients, 12625 genes)***
- Clinical outcome of breast cancer,
  - L. van't Veer *et al, Nature*, 2002. ***(98 patients, 24188 genes)***
- Recurrence of hepatocellaur carcinoma after curative resection,
  - N. Iizuka *et al*, *Lancet*, 2003. ***(60 patients, 7129 genes)***
- Tumor vs. normal colon tissues,
  - A. Alon *et al*, *PNAS,* 1999. ***(62 patients, 2000 genes)***
- Acute Myeloid vs. Lymphoblastic Leukemia,
  - T. Golub *et al*, *Science,* 1999. ***(72 patients, 7129 genes)***

Problem definition

- A way to represent the dataset produced through several (*m*) microarray experiments is to build a table, in which each column contains the *n* gene expression levels obtained in a single experiment.
- Typically, *m* ~ 10, whereas *n* ~ 1000.

	gene #1	*gene #2*	*…*	*gene #n*	*label*
experiment #1	1234	546	…	657	health
experiment #2	334	334	…	778	disease
…	…	…	…	…	…
experiment #m	654	123	…	982	heath

- A final column contains a classification label for the experiment.

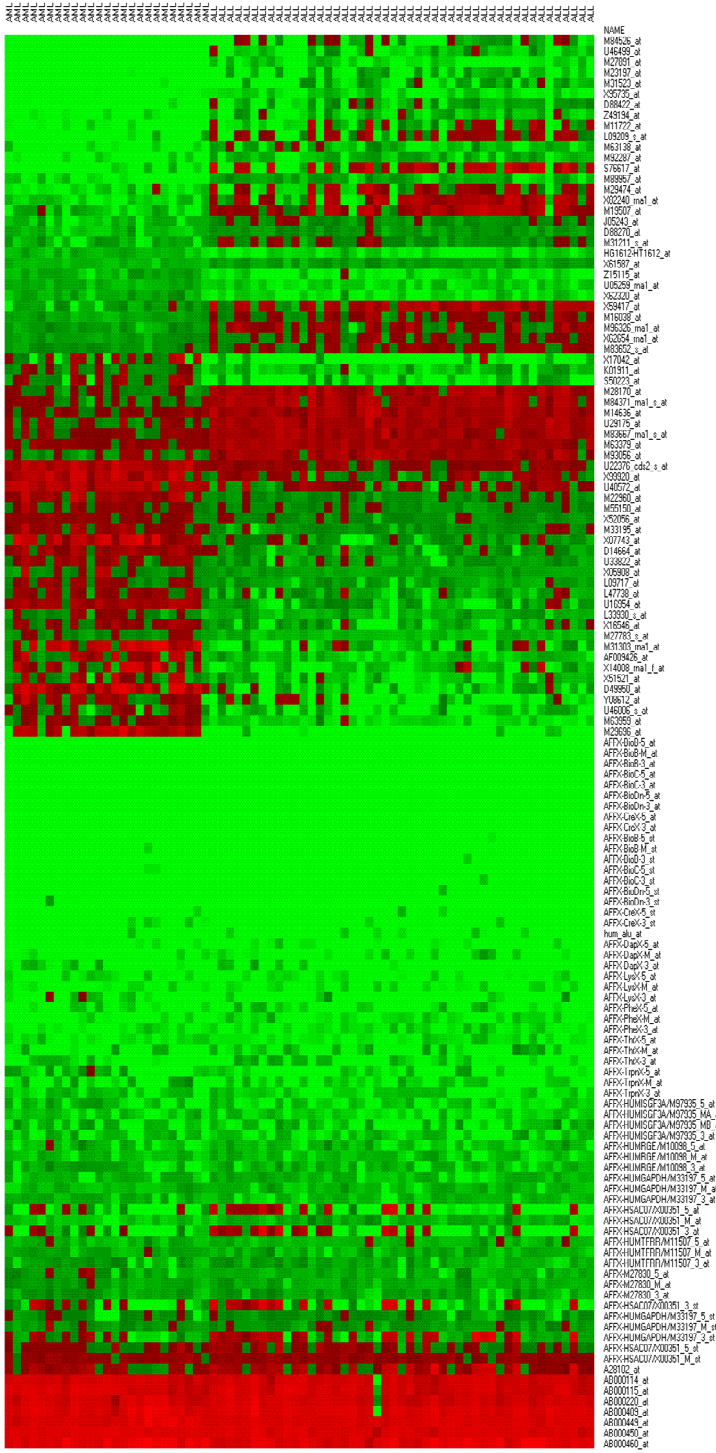
Problem definition

- If we look at the table by rows, we obtain *m* points in a space with *n* dimensions (tissue space).
- Each row represents the state of cells inside a given tissue.
- When an output class is attached to every tissue the target of the problem can be stated as:

> Find a classifier that provides the correct output for tissues not included in the original dataset

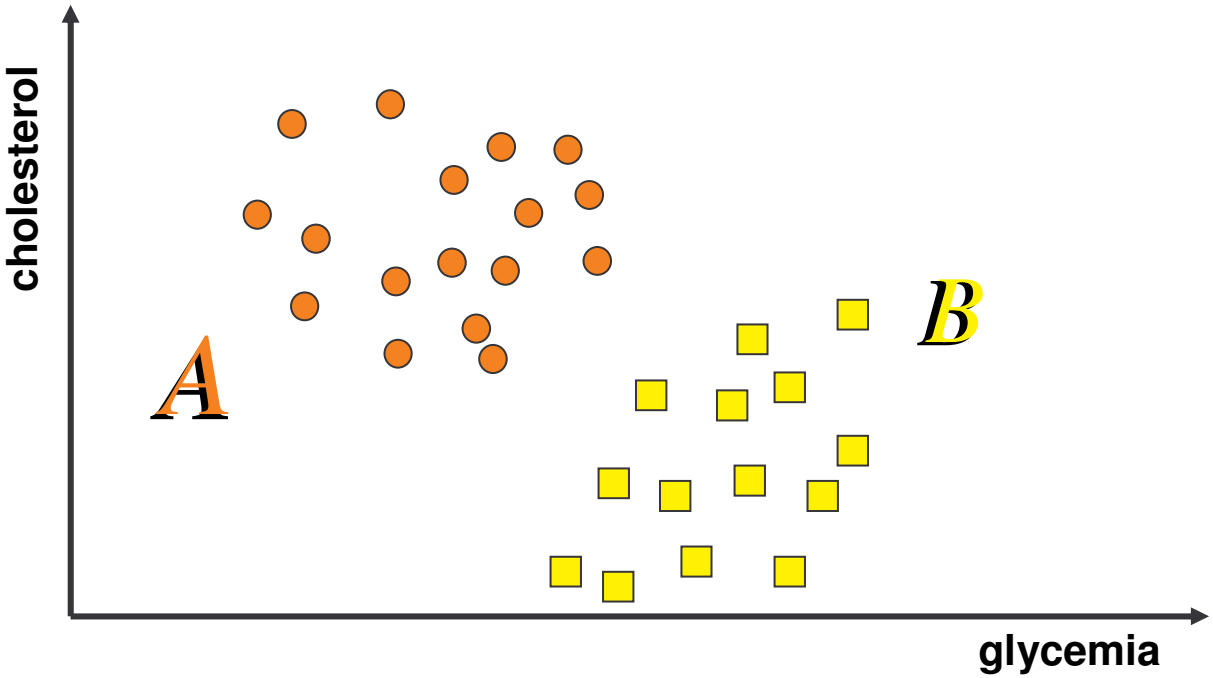
Problem characteristics

- Data produced in microarray experiments are exponentially increasing.
- Data are stored and annotated in publicly available databases, which makes it possible to build large datasets.
- Experiments contain expression values for tens of thousands of genes.
- Data can be updated, which poses problems to the training step.



A toy problem

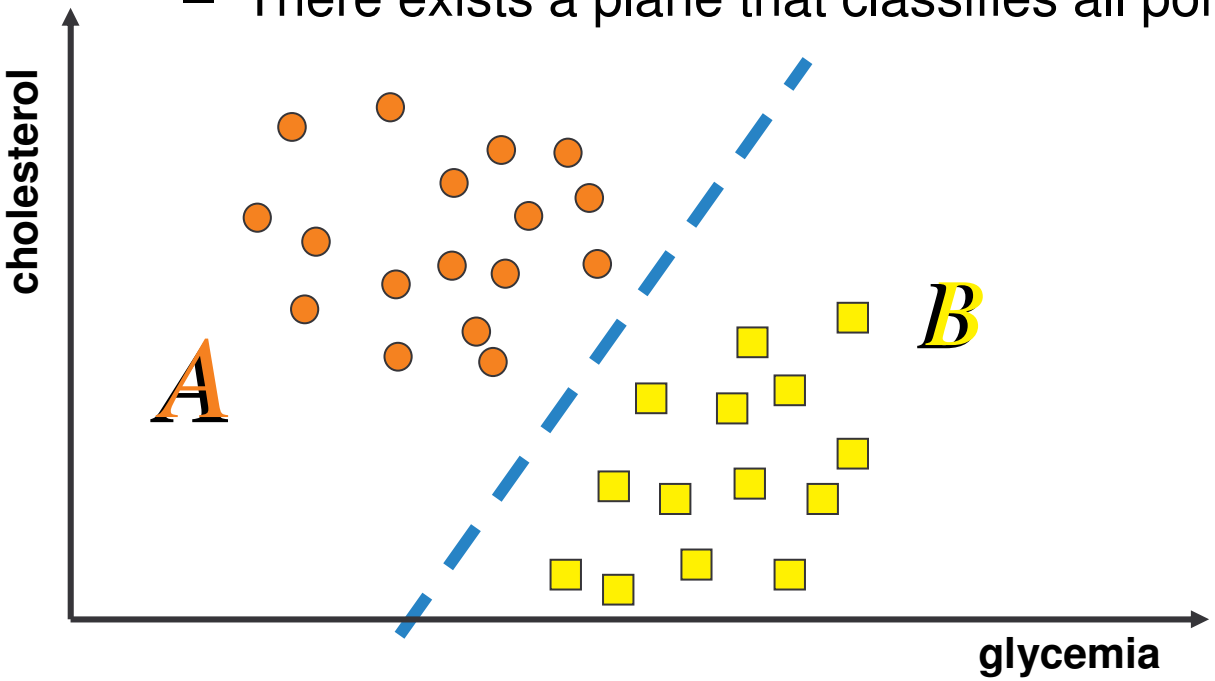
- Consider two sets of patients, for which glycemia and cholesterol have been analyzed.



- Suppose there exist an illness that affects patients in set A, but not in set B

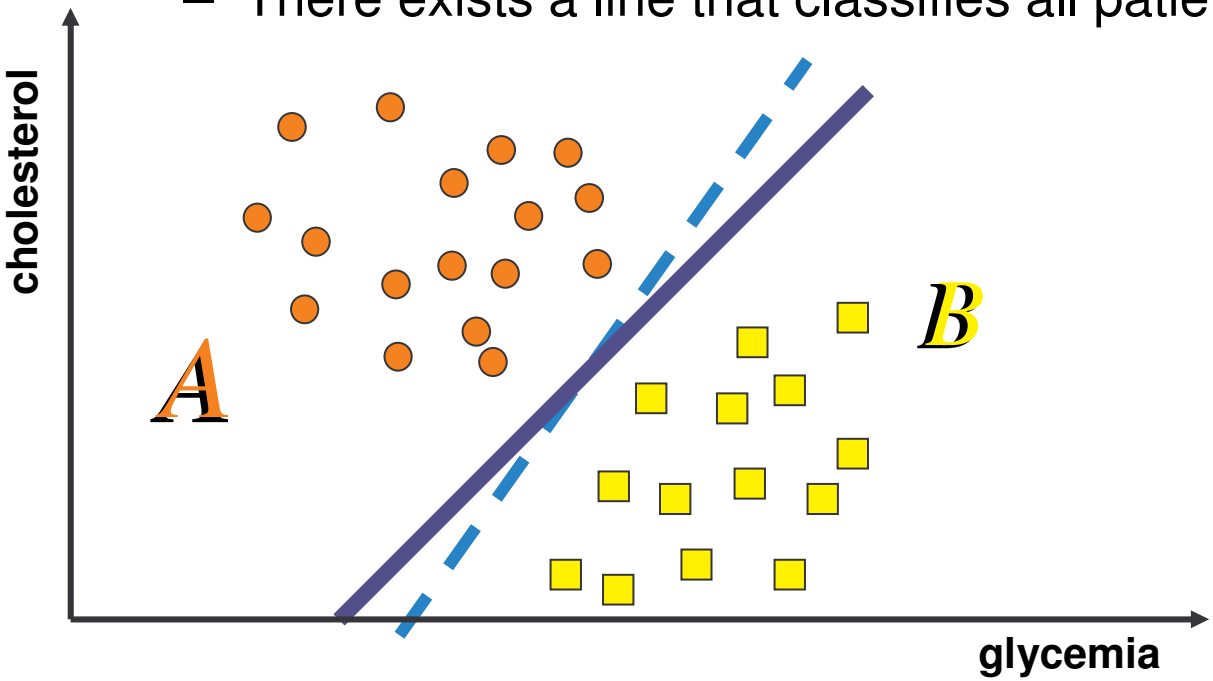
A toy problem

- We want to separate patients in A from B: for each new one, we want to predict his health state.
  - There exists a plane that classifies all points in the two sets



A toy problem

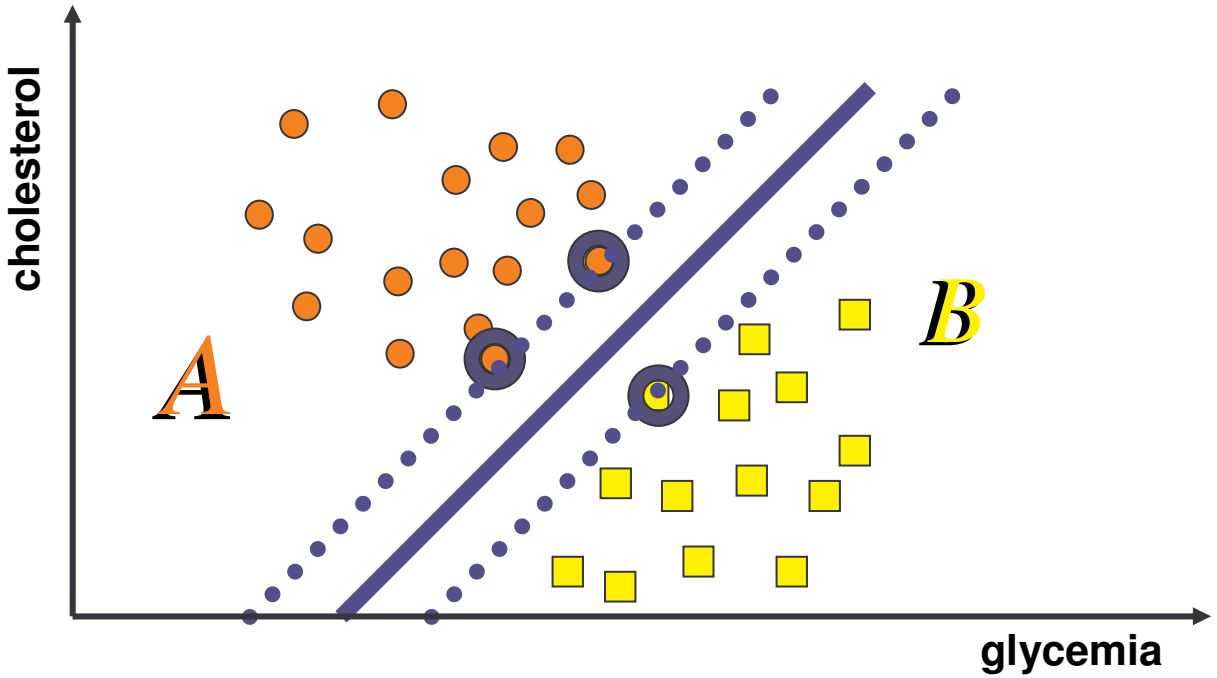
- We want to separate patients in A from B, for each new one, we want to predict his health state.
  - There exists a line that classifies all patients in the two sets



- There are infinitely many lines that correctly classify the training data.

Support vector machines

- Maximize the distance between *support planes*
  - Support planes leave all points of a class on one side



$$\min_a \frac{1}{2}\|w\|^2$$

$$s.t.$$

$$Aw + b \geq e$$

$$Bw + b < -e$$

- Support planes are pushed apart until they "bump" into a small set of data points (*support vectors*).

Support Vector Machine features

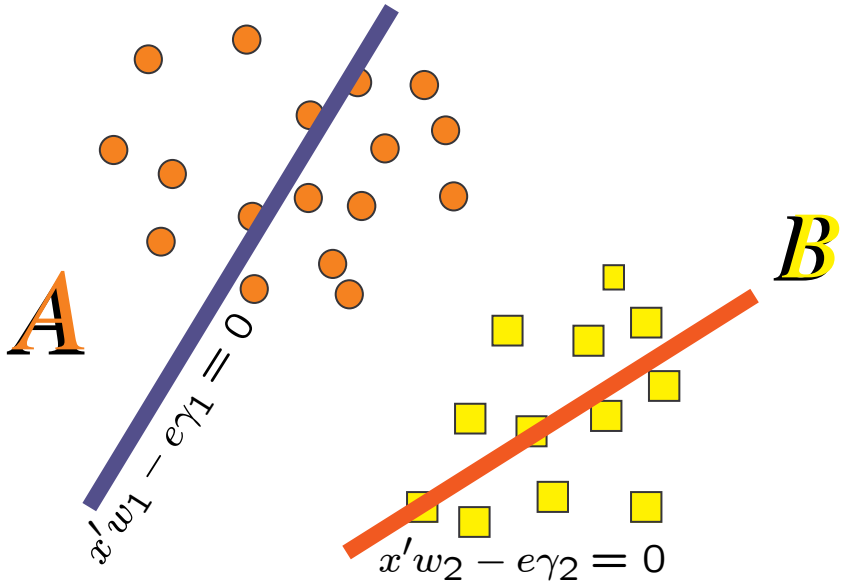
- Support Vector Machines are the state of the art for the existing classification methods for gene expression data analysis (*Brown, PNAS, 1999*).

- Their robustness is due to the strong fundamentals of statistical learning theory (Vapnik, 1995).

- Many implementations available: Matlab, R, Weka, M@chbet,...

A different approach: GEC

- The problem can be restated as: find two hyperplanes, each the closest to one set and the furthest from the other.



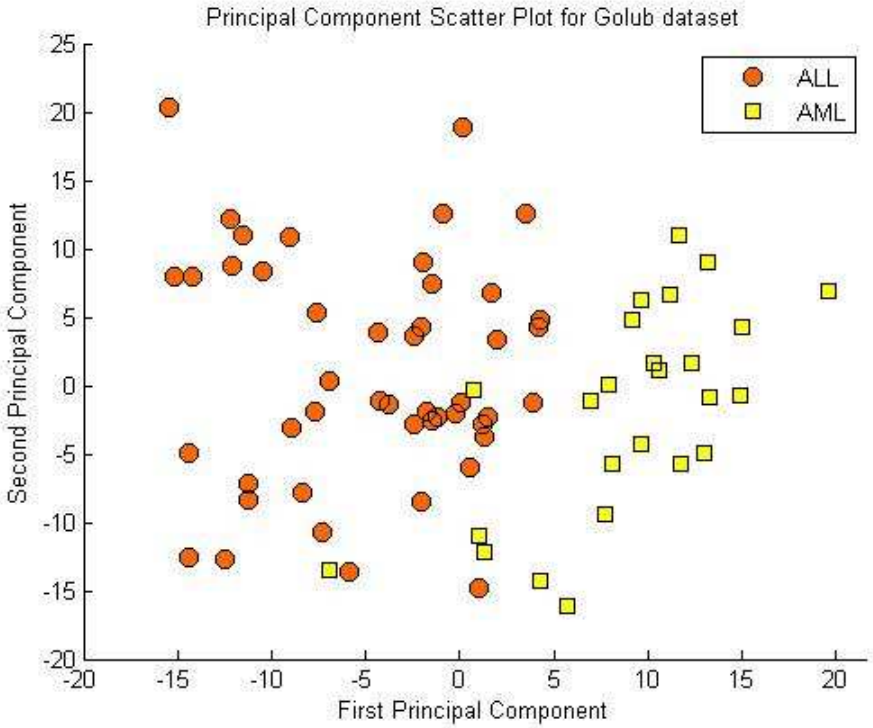
$$\min_{w_1,\gamma_1 \neq 0} \frac{\|Aw_1 - e\gamma_1\|^2}{\|Bw_1 - e\gamma_1\|^2}$$

- The binary classification problem can be solved as a generalized eigenvalue computation (GEC).

Mangasarian et al , TPAMI, 2006.

Life is not linear

- Real life data are usually nonlinearly separable…



- Data is nonlinearly transformed in another space to increase separability, and linear discrimination is found in that space.

T. Golub et. al Science, 1999.

The kernel trick

- A standard technique is to transform points into a nonlinear space, via kernel functions, like the *Gaussian kernel*:

$$K(x_i, x_j) = e^{-\frac{\|x_i - x_j\|^2}{\sigma}}$$

- Each element of the *kernel matrix* is:

$$K(A, C)_{i,j} = e^{-\frac{\|A_i - C_j\|^2}{\sigma}}$$

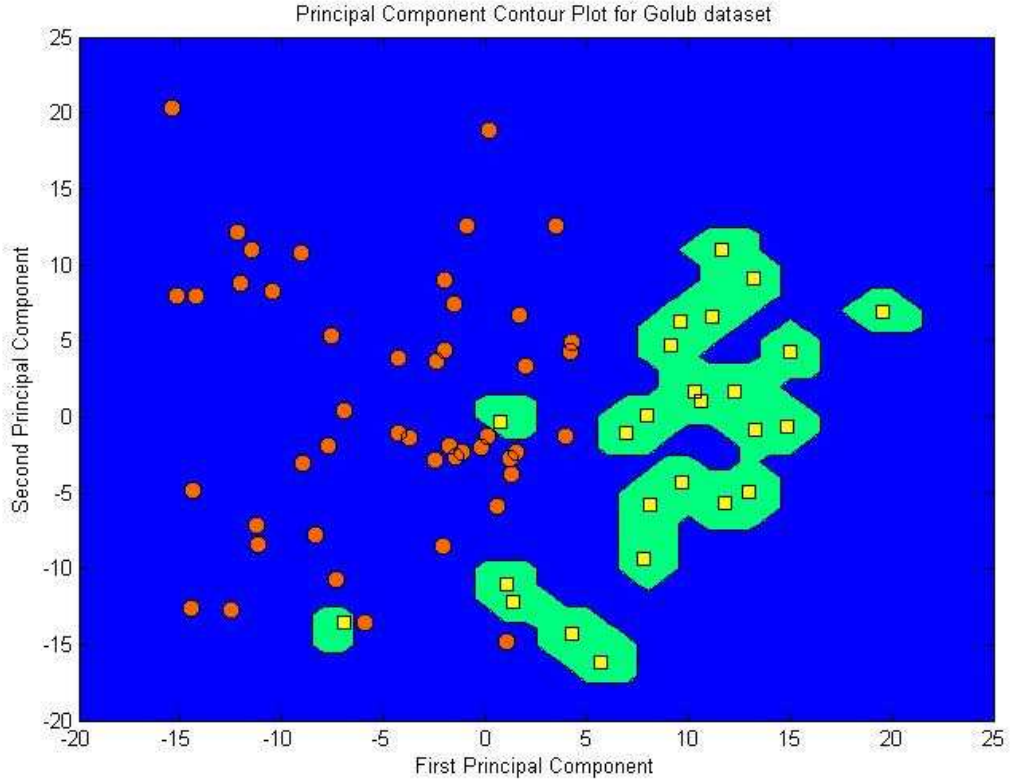
where

$$C = \begin{bmatrix} A \\ B \end{bmatrix}$$

Bennett et al. OMS, 1992.

Generalizability of the methods

- The classification surfaces can be very tangled.



- Those models are good on original data, but do not generalize well to new data (*over-fitting*).

How to solve the problem?

Incremental classification

- A possible solution is to find a small and robust subset of the training set that provides comparable accuracy results.
- A smaller set of points reduces the probability of over-fitting the problem.
- A kernel built from a smaller subset is computationally more efficient in predicting new points, compared to kernels that use the entire training set.
- As new points become available, the cost of retraining the algorithm decreases if the influence of the new points is only evaluated by the small subset.

Guarracino et al, JC, 2007.

Other supervised methods

- Linear and quadratic discriminants (*Dudoit et al.*, 2002)
- K-Nearest Neighbors (*Pomeroy et al.*, 2002)
- Neural networks (*Khan et al.*, 2001)
- Decision trees (*Dudoit et al.,* 2000)
- Ensamble methods (*Dudoit et al.*, 2002; *Diettling and Buhlmann*, 2003)

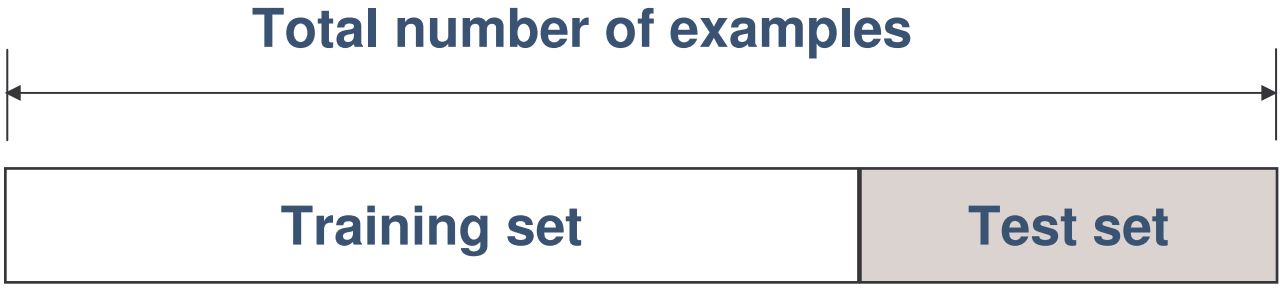
Error estimation

- **Holdout**: the labeled dataset is divided in two parts: the training set (~2/3), and the test set (~1/3).
- **Random sampling**: the holdhout is reapeted on random training sets.
- **k-fold cross validation**: data is partitioned in k distinct subsets; each time a different fold is chosen for test.
- **Leave one out**: 1-fold cross validation.
- **P-value**: significance level below which null hypothesis is rejected.
- **ROC curves**: tradeoff between positive hits and false allarms.

Holdout

- **Holdout**: the labeled dataset is divided in two parts: the training set (~2/3), and the test set (~1/3).

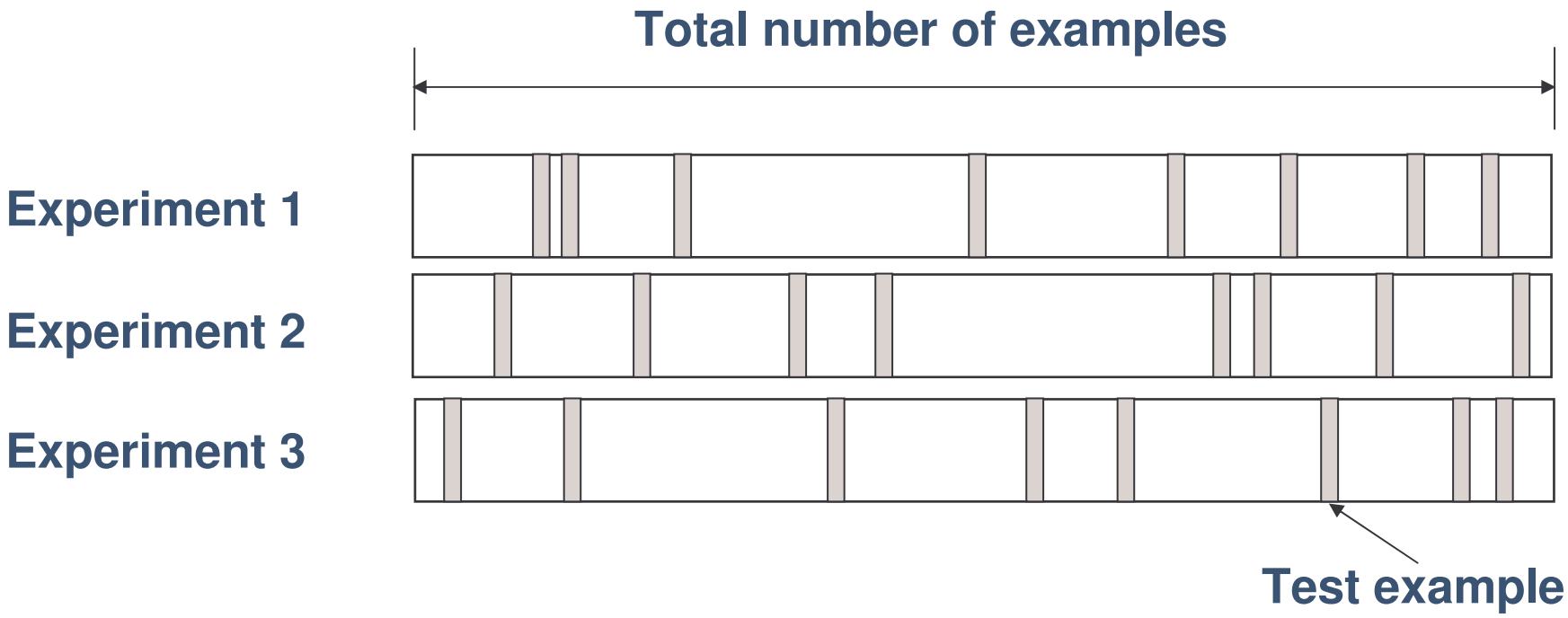
- The training set is used to train the classifier, the test set to evaluate the error rate.



- Cons: Since it is a single train and test experiment, the holdout error estimate can be misleading, in case of "unfortunate" split.

Random sampling

- **Random sampling**: the holdhout is repeated on random training sets, and error Ei is estimated on test examples.

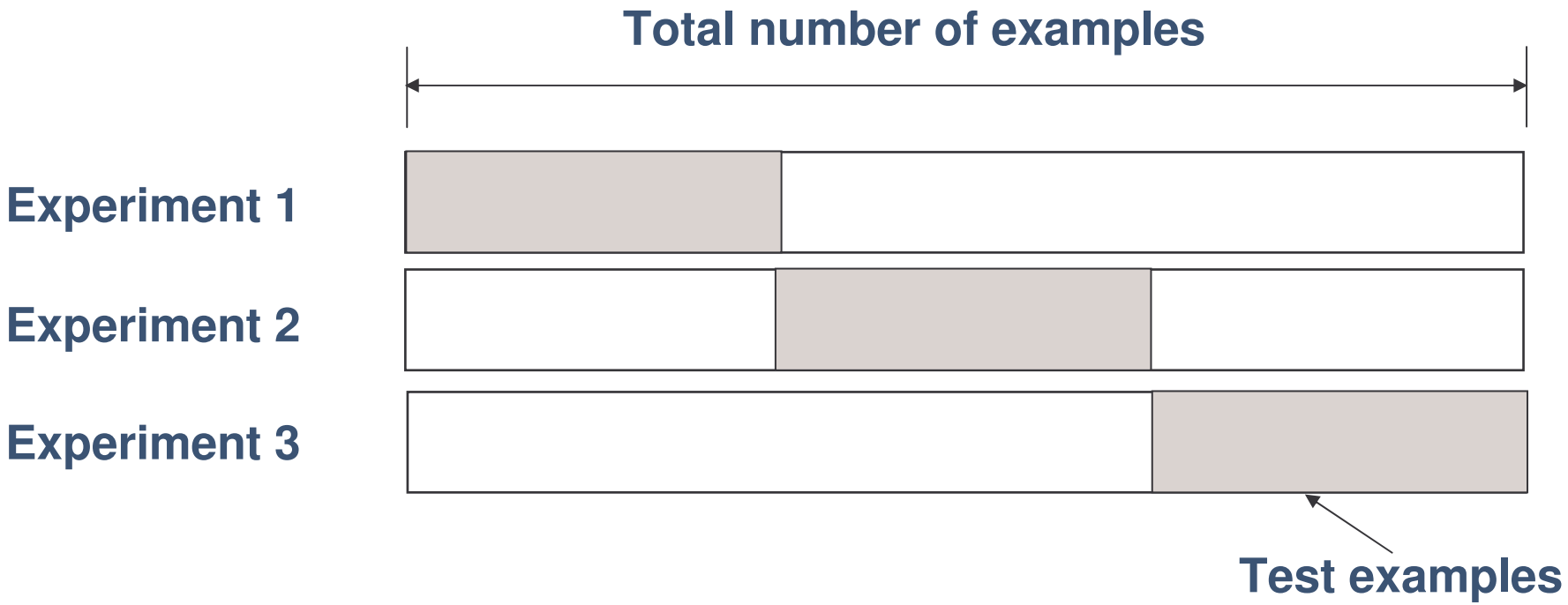


- The error is obtained as the average of estimates E_i:

$$E_i = \frac{1}{K} \sum_{i=1}^{K} E_i$$

k-fold cross validation

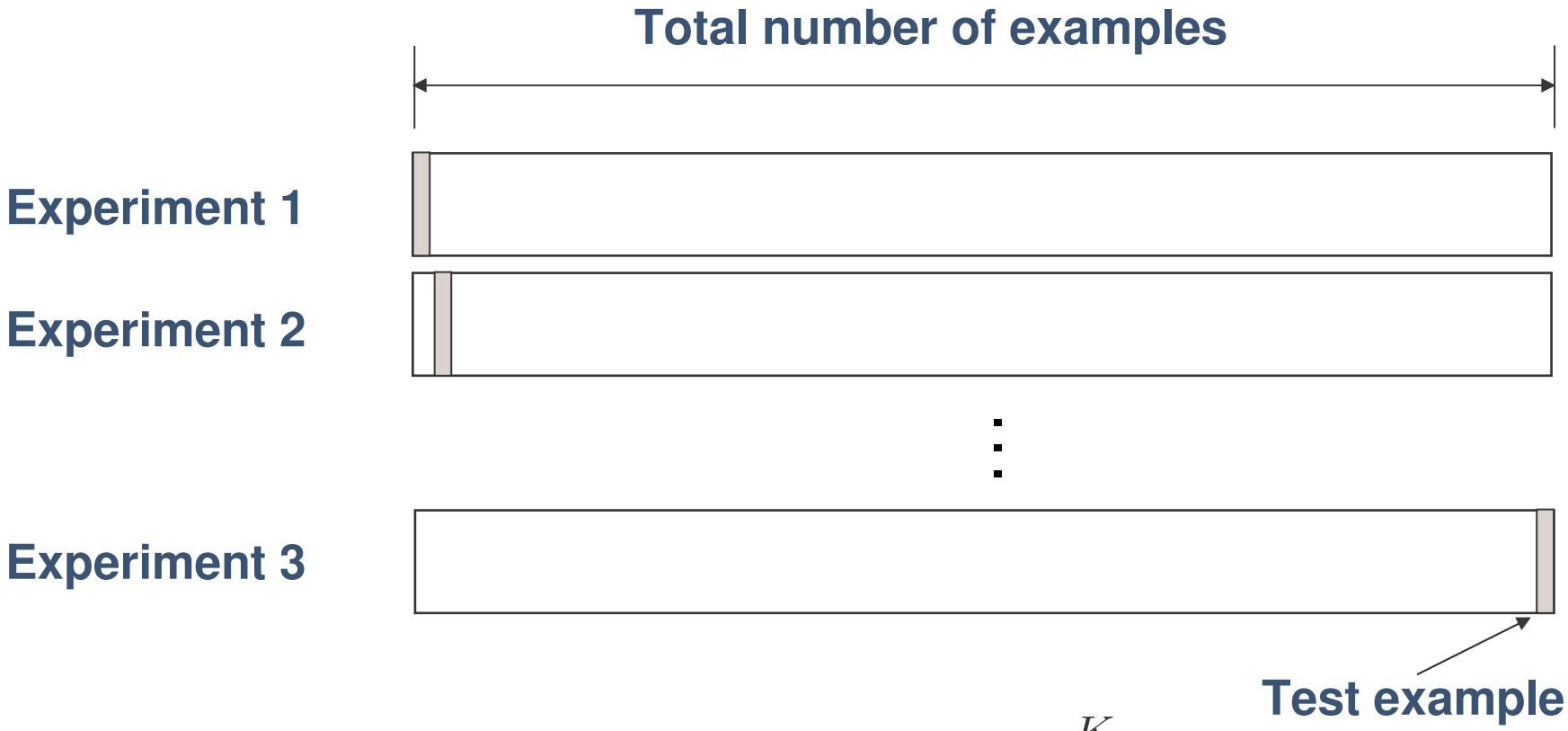
- **k-fold cross validation**: data is partitioned in k distinct subsets; each time a different fold is chosen for test.



- Pros: all examples in the dataset are used for both training and test.
- As befor the classification error is: $E_i = \frac{1}{K} \sum_{i=1}^{K} E_i$

Leave one out

- **Leave one out**: 1-fold cross validation.



- As usual the error estimate is: $E_i = \frac{1}{K} \sum_{i=1}^{K} E_i$

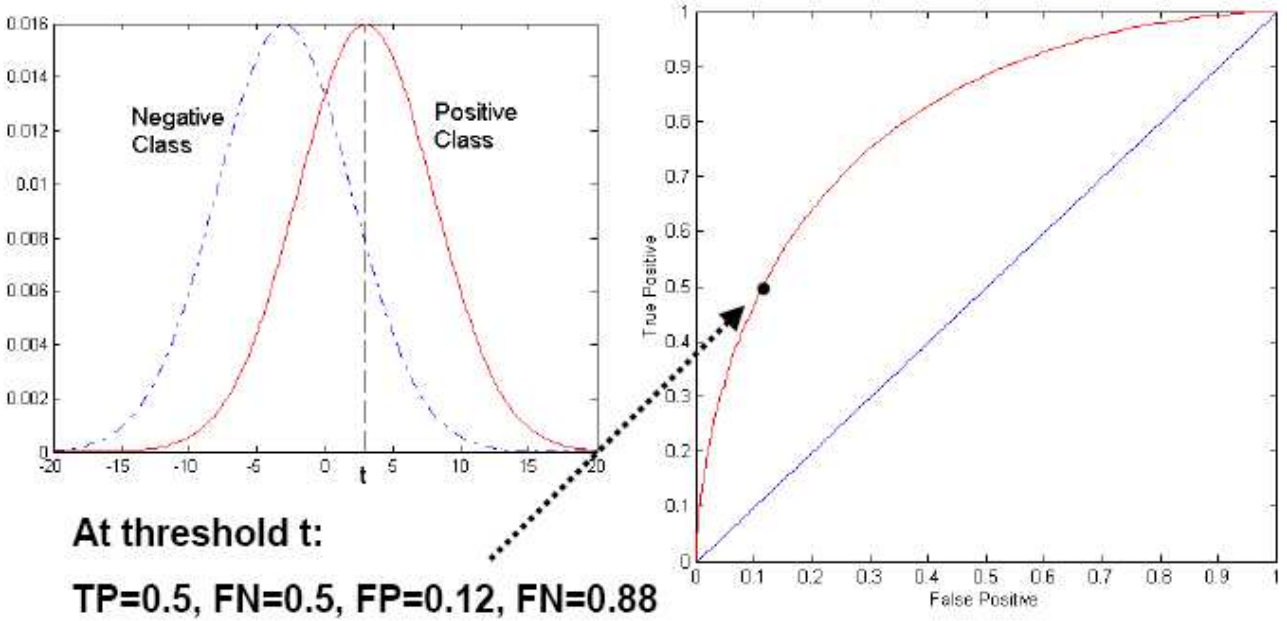
P-value

- **P-value**: significance level below which null hypothesis is rejected.
- Null hypotesis: random classifier does not perform better than a fixed classification method.
- Perform k=100 random sampling.
  - For each k: 100 random shuffles of the labels in training set and train.
  - Sum up the cases in which the random classifier outperforms the classifier on the error estimation.
- If the random classifier does better 500 times out of 10000 repetitions ($p < 0.05$), the classification method is not that good!

ROC curves

- **ROC curves**: tradeoff between positive hits and false alarms.

- 1-dimensional data set containing 2 classes (positive and negative)
- any points located at x > t is classified as positive



At threshold t:

TP=0.5, FN=0.5, FP=0.12, FN=0.88

Ongoing research

- How to derive grouping of genes responsible of classification?
- How to express conceptual a priori knowledge in more complex formulation than a {-1,1} label?
- How to integrate different data sources to obtain a system view?

Conclusions

- When a priori information is available, use it!
- Use your knowledge, don't relay only on number crunking.
- Be aware of very large and very little datasets.
- To validate your results, don't forget Vapnik's statistical learning theory!