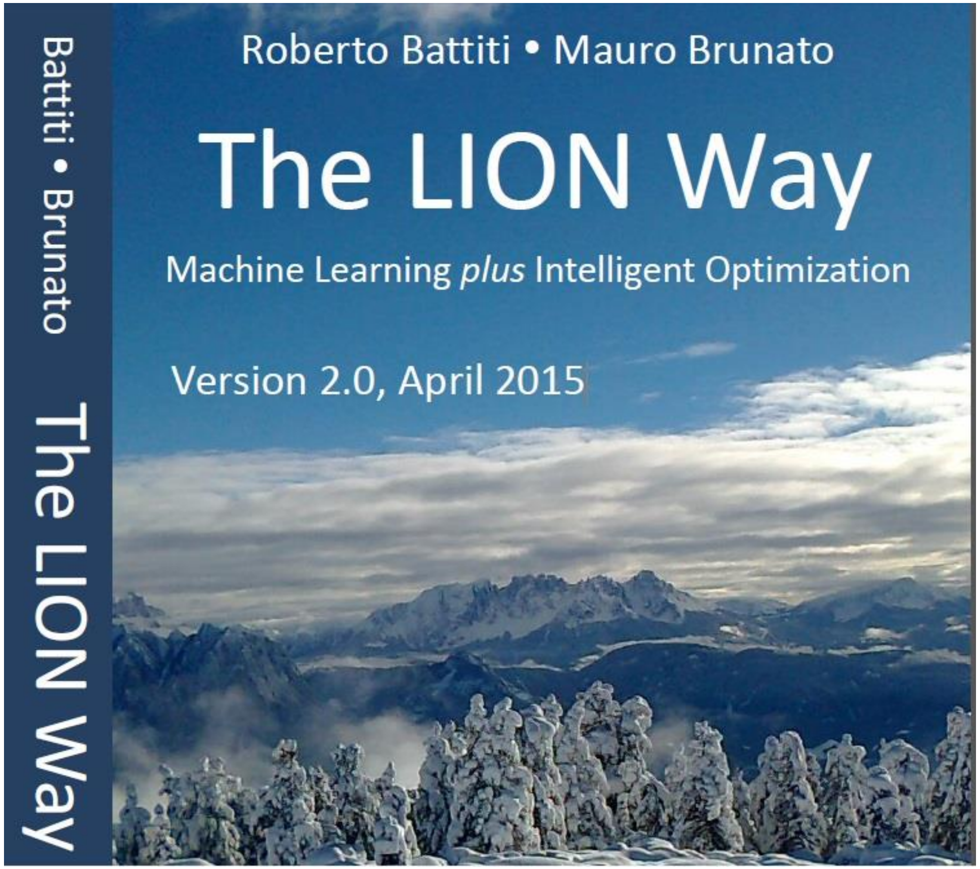
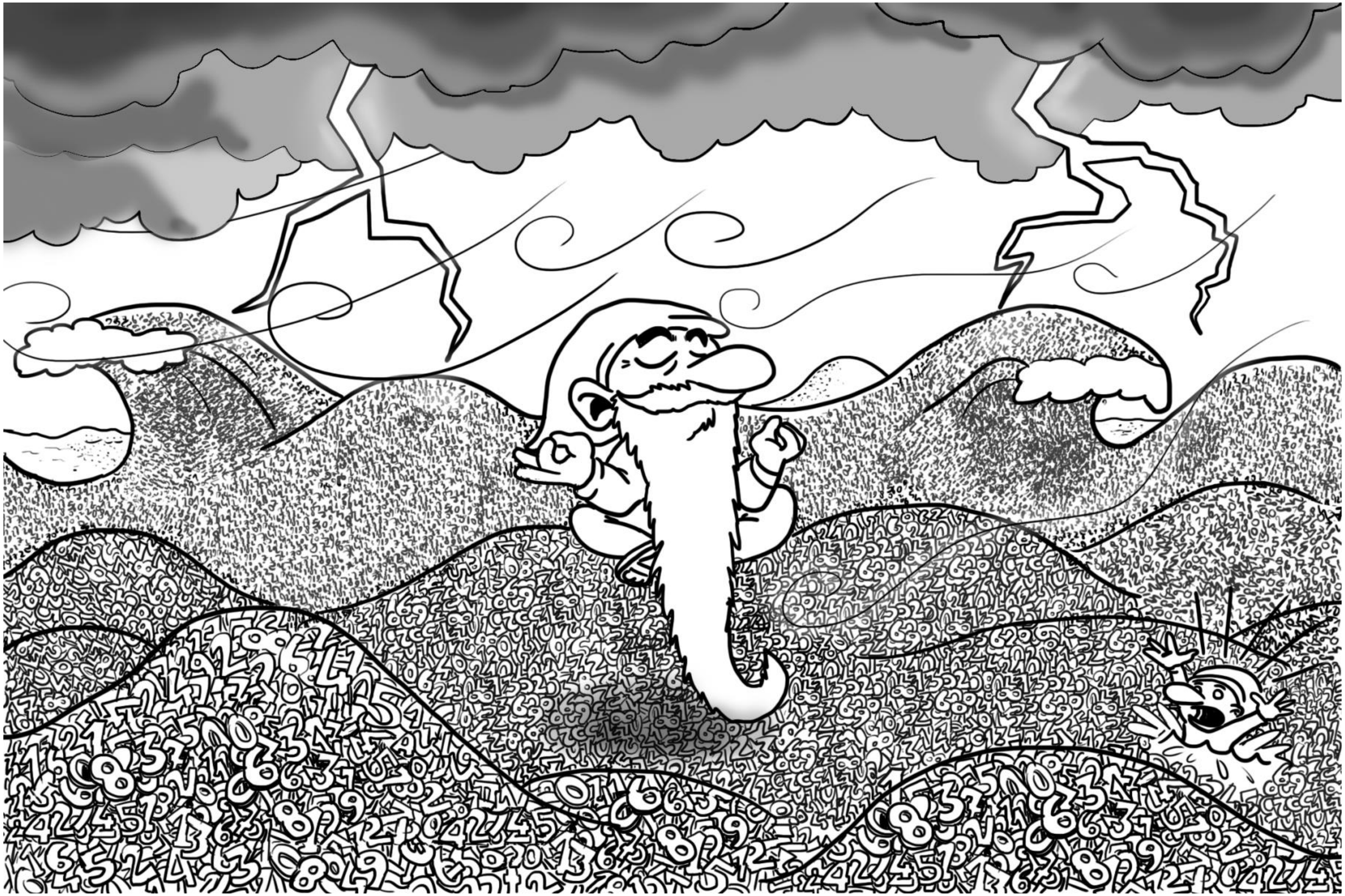

Roberto Battiti • Mauro Brunato
The LION Way
Machine Learning *plus* Intelligent Optimization
Version 2.0, April 2015
Battiti • Brunato
The LION Way

ROBERTO BATTITI, MAURO BRUNATO.
The LION Way: Machine Learning plus *Intelligent Optimization*.
LIONlab, University of Trento, Italy,
Apr 2015

http://intelligent-optimization.org/LIONbook

Chap.3 Learning requires a method

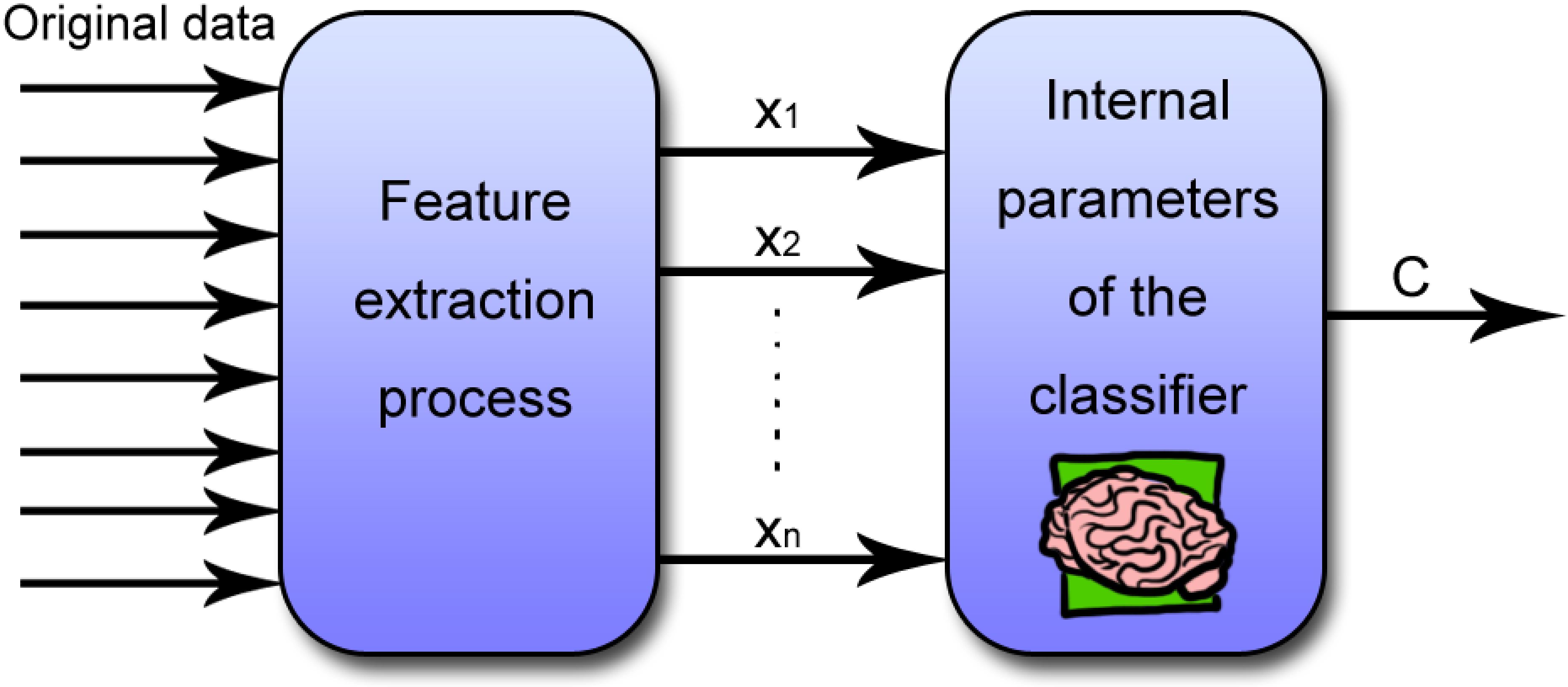
Data Mining, noun 1. Torturing the data until it confesses. . . and if you torture it long enough, you can get it to confess to anything.



What is learning?

- Unifying different cases by discovering the underlying explanatory laws.
- Learning from examples is only a means to reach the real goal: generalization, the capability of explaining new cases

Supervised learning architecture: feature extraction and classification


Original data
Feature extraction process
x_1
x_2
x_n
Internal parameters of the classifier
C

http://intelligent-optimization.org/LIONbook/

Performance estimation

- If the goal is generalization, estimating the performance has to be done with extreme care

- Feature extraction →

  (xi; **yi**), i = 1;...; L

- Classification
- Regression

Output can be probability

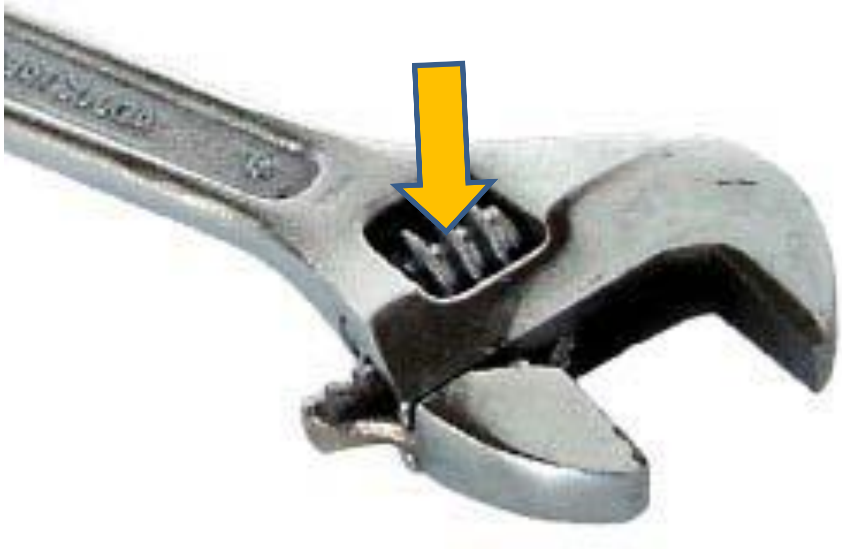
http://intelligent-optimization.org/LIONbook/

Learning from labeled examples: minimization and generalization

- A **flexible model f(x;w),** where the flexibility is given by some tunable parameters (or weights) **w**

- determination of the best parameters is fully automated, this is why the method is called *machine* learning after all

Flexible model with *internal parameters* (mental images)



http://intelligent-optimization.org/LIONbook/

Learning from labeled examples: minimization and generalization (2)

- fix the free parameters by demanding that the **learned model works correctly on the examples in the training set**.

- **power of optimization**: we start by defining an error measure to be minimized, an automated optimization process to determine optimal parameters

Learning from labeled examples: minimization and generalization (3)

- suitable **error measure** is the **sum of the errors** between the correct answer (given by the example label) and the outcome predicted

- if the function is smooth one can discover points of low altitude by being blindfolded and parachuted to a random initial point…

(gradient descent)

RMS (root mean square) error function

- Indivisual errors
- Square
- Average (Sum and divide)
- Square root is optional... (optimizing sum of squares or its square root leads to the sam eresult)

$$RMS = \sqrt{\frac{e_1^2 + e_2^2 + \cdots + e_\ell^2}{\ell}}$$

Bias-Variance dilemma

- minimization of an error function is a first critical component, but not the only one.

- We are interested in generalization → Model complexity matters!

Bias-Variance dilemma

1. Models with **too few parameters** are inaccurate because of a **large bias**: they lack flexibility.
2. Models with **too many parameters** are inaccurate because of a **large variance**: they are too sensitive to the sample details (changes in the details will produce huge variations).
3. Identifying the best model requires identifying the proper “model complexity”, i.e., the proper architecture and number of parameters.

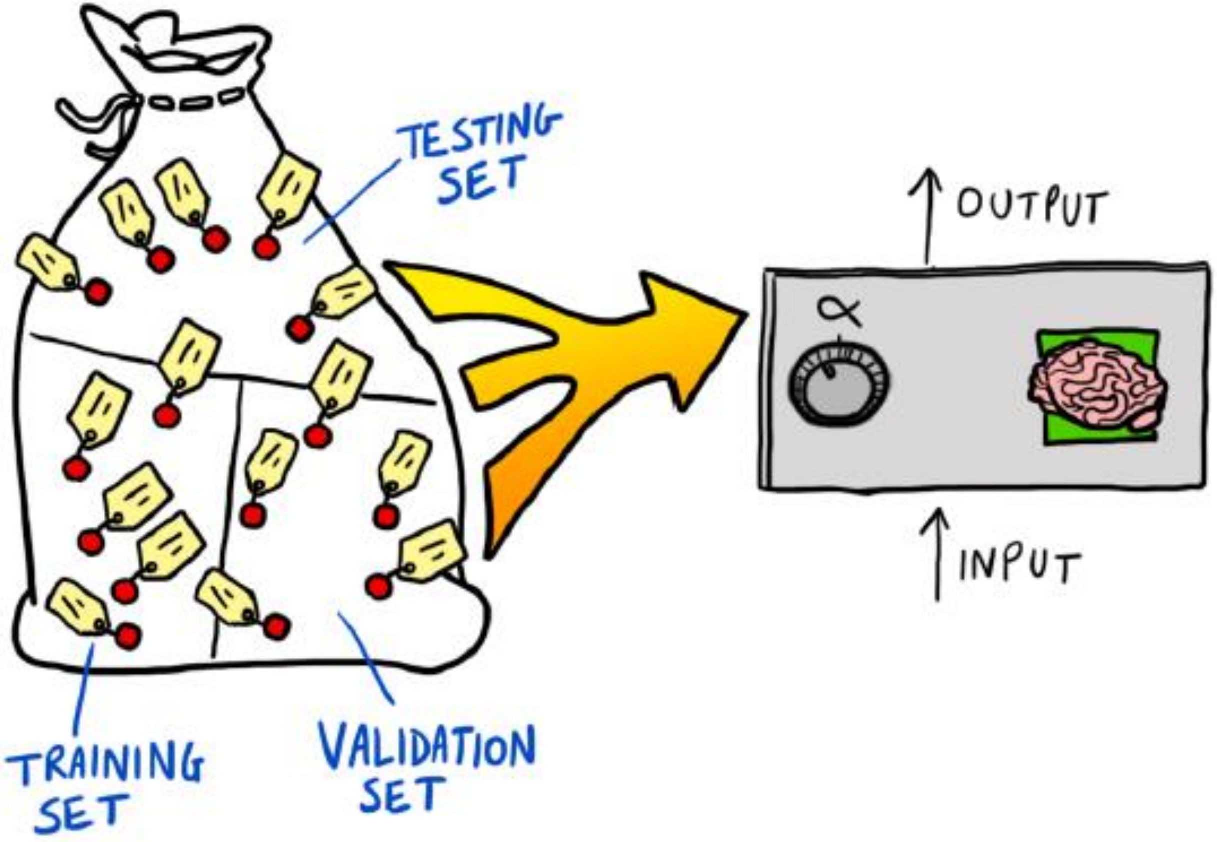
Learn, validate, test!

- careful experimental procedures to measure the effectiveness of the learning process.

- It is a terrible mistake to measure the performance of the learning systems on the same examples used for training

- **The test set is used only once** for a final measure of performance.

Learn, validate, test!

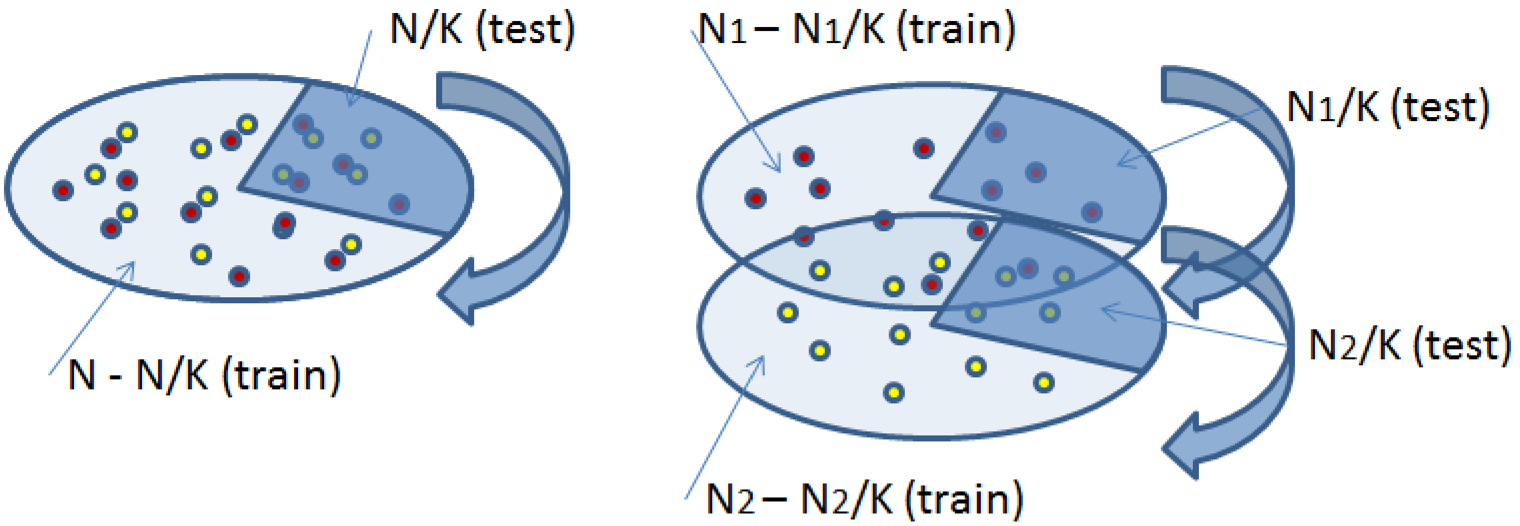


http://intelligent-optimization.org/LIONbook/

K-fold Cross-validation

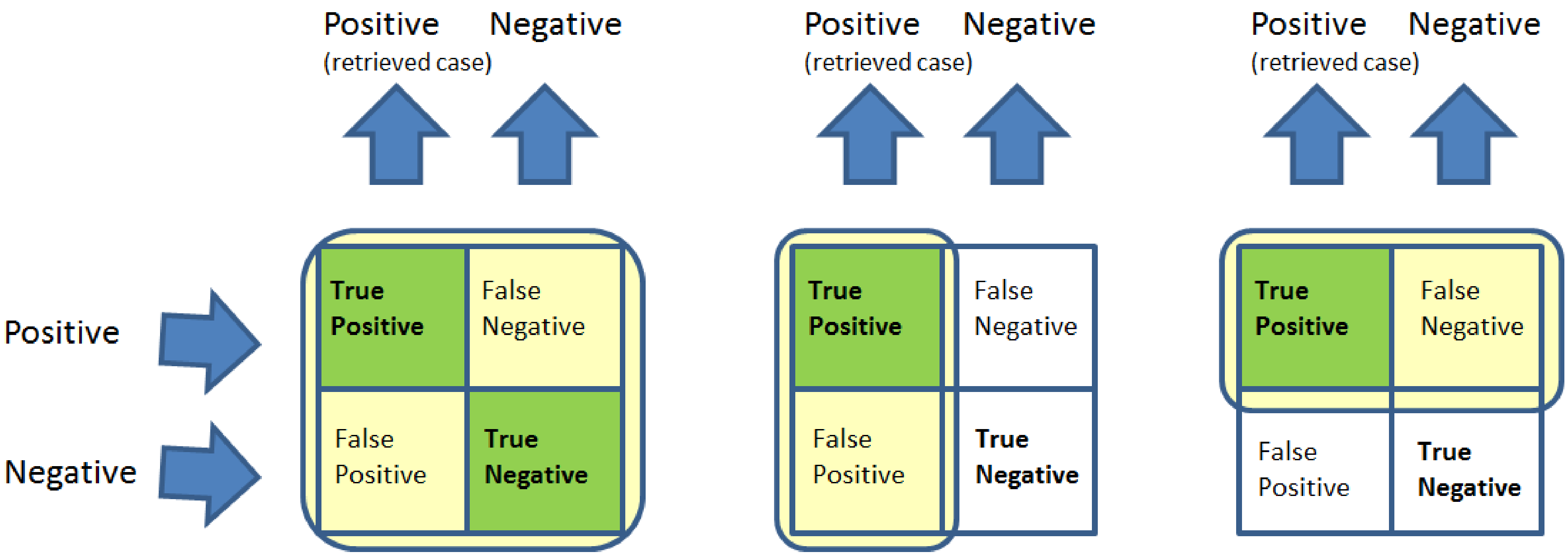
- the original sample is randomly partitioned into **K subsamples**.
- A single subsample is used as the validation data for testing, and the remaining K - 1 subsamples are used as training data.
- The cross-validation process is then repeated K times (the folds), with each of the K subsamples used exactly once as the validation data.
- Validation results are **averaged**

K-fold Cross-validation



Advanced: stratification helps in classification (subdivide *each* class)

Errors of different kinds



	Positive (retrieved case)	Negative
Positive	**True Positive**	False Negative
Negative	False Positive	**True Negative**

Accuracy

Positive (retrieved case)	Negative
True Positive	False Negative
False Positive	**True Negative**

Precision

Positive (retrieved case)	Negative
True Positive	False Negative
False Positive	**True Negative**

Recall

http://intelligent-optimization.org/LIONbook/

Errors of different kinds

- **Accuracy** is defined as the fraction of correct answers over the total
- **Precision** as the fraction of correct answers over the number of *retrieved* (positive) cases
- **Recall** is computed as the fraction of correct answers over the number of relevant (true positive) cases.

Gist

- The goal of machine learning is to use a set of training examples to realize a system which will correctly generalize to new cases, in the same context but not seen during learning.
- ML learns, i.e., determines appropriate values for the free parameters of a flexible model, by automatically minimizing a measure of the error on the example set, possibly corrected to discourage complex models, and therefore improving the chances of correct generalization.
- The output value of the system can be a class (classification), or a number (regression). In some cases having as output the probability for a class increases flexibility of usage.

Gist

- Accurate classifiers can be built without any knowledge elicitation phase, **just starting from an abundant and representative set of example data**. This is a dramatic paradigm change.
- ML is very powerful but requires a strict method (a kind of “pedagogy” of ML). For sure, **never estimate performance on the training set** — this is a mortal sin: be aware that re-using validation data will create optimistic estimates. If examples are scarce, use **cross-validation** to show off that you are an expert ML user.
- To be on the safe side, set away some test examples and use them only **once** at the end to estimate performance.
- There is no single way to measure the performance of a model, **different kinds of mistakes** can have very different costs. Accuracy, precision and recall are some possibilities, a confusion matrix is giving the complete picture for more classes.