© Roberto Battiti and Mauro Brunato, 2015, all rights reserved.

Slides can be used and modified for classroom usage, provided that the attribution (link to book website) is kept.
Chap.2 Lazy learning: nearest neighbors

Natura non facit saltus
Nature does not make jumps
Learning from examples

• written characters ("a", "b", "c", . . .)
• labeled examples
• Memorization is different from extracting the underlying patterns and regularities

• Generalization is the key
Learning from examples

Mushroom hunting requires classifying edible and poisonous species.
Learning from examples

• In supervised learning a system is trained by a supervisor (teacher) giving labeled examples.

• Each example is an array, a vector of input parameters $x$ called features with an associated output label $y$. 
Lazy “nearest neighbor” method of machine learning.

- Lazy beginners in mushroom picking
Nearest-Neighbors Methods

• nearest-neighbors basic form of learning, also related to instance-based learning, case-based or memory-based, pattern patching

• Classify a new example according to the nearest neighbor (considering input values)
Neighbor $\rightarrow$ Neighbors

k-nearest-neighbors (KNN)

Majority or unanimity rule
Regression

- regression (the prediction of a real number, like the content of poison in a mushroom), the output can be obtained as a simple average
Weighted K-nearest neighbors (WKNN)

Closer neighbors receive a higher weight (and more influence to derive the output value)

\[ y = \frac{\sum_{j=1}^{k} \frac{y_{ij}}{d(x_{ij}, \mathbf{x}) + d_0}}{\sum_{j=1}^{k} \frac{1}{d(x_{ij}, \mathbf{x}) + d_0}}; \]
Kernels and locally weighted regression

- Kernel methods and locally-weighted regression can be seen as flexible and smooth generalizations of the nearest-neighbors idea.
- Instead of applying a brute exclusion of the distant points, all points contribute to the output but with a significance ("weight") related to their distance from the query point.
Gist

KNN (K Nearest Neighbors) is a primitive and lazy form of machine learning: just store all training examples into memory.

When a new input to be evaluated appears, search in memory for the K closest examples stored. Read their output and derive the output for the new input by majority or averaging. Laziness during training causes long response times when searching a very large memory storage.

KNN works in many real-world cases because similar inputs are usually related to similar outputs, a basic hypothesis in machine learning. It is similar to some “case-based” human reasoning processes. Although simple and brutal, it can be surprisingly effective in many cases.