The Nearest-Neighbor and the k-Nearest-Neighbor algorithms

In their simplest form, these algorithms do not perform any computation during training. The computation is performed only when a test example is presented. Therefore, they are described with input that contains the training examples and one test example. The expensive step in these algorithms is the computation of nearest-neighbors. Efficient implementations exist with sophisticated data structures for efficient computation of nearest-neighbors. These are not discussed here.

Input: m training examples, given as the pairs (x_i, y_i), where x_i is an n-dimensional feature vector and y_i is its label. A test example x.

Output: y, the computed label of x.

The Nearest-Neighbor algorithm

a. Determine x_i nearest to x. It minimizes the distance to x according to a pre-defined norm.

$$\text{distance}(x_i, x) = |x_i - x| \tag{1}$$

b. Return $y = y_i$.

The most commonly used norm in (1) is the Euclidean norm:

$$|x_i - x|^2 = \sum_{j=1}^{n} (x_i(j) - x(j))^2$$

There are multiple approaches for handling the case in which there is more than one training example nearest to x.

The k-Nearest-Neighbor algorithm

a. Determine $x_{i1}, \ldots, x_{ik}$, the k training examples nearest to x according to a pre-defined norm.

b. Let $y_{i1}, \ldots, y_{ik}$ be the labels of the k nearest neighbors. Choose y as the label that appears most frequently among $y_{i1}, \ldots, y_{ik}$.

There are multiple approaches for handling the case in which no label has a clear majority in b.

The value of k

In many practical problems k-NN with $k > 1$ performs better than the simple 1-NN. The most effective method of estimating a useful value of k is the technique of cross validation.

Example

Training data:

i	x_i	y_i
1	(1,1)	A
2	(1,2)	A
3	(2,2)	B
4	(2,3)	B

To classify the test example (3,2) according to k-NN we need to compute its distances to the 4 examples. The square of the Euclidean distances is shown in the following table:

i	x_i	y_i	$\|x_i - (3,2)\|^2$
1	(1,1)	A	5
2	(1,2)	A	4
3	(2,2)	B	1
4	(2,3)	B	2

Using 1-NN the nearest example has the index $i = 3$, and the label is: $y = y_3 = B$.
Using 3-NN the 3 nearest examples are with indexes $i = 3, 4, 2$. Two have label B and one label A, so that the computed label is $y = B$.