5. *K*-Nearest Neighbor

Pedro Larrañaga

Intelligent Systems Group
Department of Computer Science and Artificial Intelligence
University of the Basque Country

Madrid, 25th of July, 2006
Outline

1. Introduction
2. The Basic $K$-NN
3. Extensions of the Basic $K$-NN
4. Prototype Selection
5. Summary
**Basic Ideas**

*K*-NN ≡ IBL, CBR, lazy learning

- A new instance is classified as the most frequent class of its *K* nearest neighbors
- Very simple and intuitive idea
- Easy to implement
- There is not an explicit model (transduction)
- *K*-NN ≡ instance based learning (IBL), case based reasoning (CBR), lazy learning
Outline

1. Introduction
2. The Basic $K$-NN
3. Extensions of the Basic $K$-NN
4. Prototype Selection
5. Summary
Algorithm for the basic $K$-NN

BEGIN

Input: $D = \{(x_1, c_1), \ldots, (x_N, c_N)\}$

$x = (x_1, \ldots, x_n)$ new instance to be classified

FOR each labelled instance $(x_i, c_i)$ calculate $d(x_i, x)$

Order $d(x_i, x)$ from lowest to highest, $(i = 1, \ldots, N)$

Select the $K$ nearest instances to $x$: $D_x^K$

Assign to $x$ the most frequent class in $D_x^K$

END

Figure: Pseudo-code for the basic $K$-NN classifier
Algorithm for the basic $K$-NN

Example

Figure: Example for the basic 3-NN. $m$ denotes the number of classes, $n$ the number of predictor variables, and $N$ the number of labelled cases.
Algorithm for the basic $K$-NN

The accuracy is not monotonic with respect to $K$

**Figure:** Accuracy versus number of neighbors
K-NN with rejection

Requiring for some guarantees

- **Demanding for some guarantees** before an instance is classified
- In case that the guarantees are **not verified** the instance remains **unclassified**
- Usual guaranty: **threshold** for the most frequent class in the neighbor
$K$-NN with average distance
**K-NN with weighted neighbors**

*Figure: K-NN with weighted neighbors*
**K-NN with weighted neighbors**

<table>
<thead>
<tr>
<th>$d(x_i, x)$</th>
<th>$w_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_1$</td>
<td>2</td>
</tr>
<tr>
<td>$x_2$</td>
<td>2</td>
</tr>
<tr>
<td>$x_3$</td>
<td>2</td>
</tr>
<tr>
<td>$x_4$</td>
<td>2</td>
</tr>
<tr>
<td>$x_5$</td>
<td>0.7</td>
</tr>
<tr>
<td>$x_6$</td>
<td>0.8</td>
</tr>
</tbody>
</table>

**Figure:** Weight to be assigned to each of the 6 selected instances
**K-NN with weighted variables**

<table>
<thead>
<tr>
<th>$X_1$</th>
<th>$X_2$</th>
<th>$C$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**Figure:** Variable $X_1$ is not relevant for $C$
**K-NN with weighted variables**

\[
d(x_l, x) = \sum_{i=1}^{n} w_i d_i(x_l, x_i) \quad \text{with} \quad w_i = MI(X_i, C)
\]

\[
MI(X_1, C) = p(x_1, c)(0, 0) \log \frac{p(x_1, c)(0, 0)}{p_x(0) \cdot p_c(0)} + p(x_1, c)(0, 1) \log \frac{p(x_1, c)(0, 1)}{p_x(0) \cdot p_c(1)} + \\
p(x_1, c)(1, 0) \log \frac{p(x_1, c)(1, 0)}{p_x(1) \cdot p_c(0)} + p(x_1, c)(1, 1) \log \frac{p(x_1, c)(1, 1)}{p_x(1) \cdot p_c(1)}
\]

\[
= \frac{3}{12} \log \frac{3}{12} \cdot \frac{3}{12} + \frac{3}{12} \log \frac{3}{12} \cdot \frac{3}{12} + \frac{3}{12} \log \frac{3}{12} \cdot \frac{3}{12} + \frac{3}{12} \log \frac{3}{12} \cdot \frac{3}{12} = 0
\]

\[
MI(X_2, C) = p(x_2, c)(0, 0) \log \frac{p(x_2, c)(0, 0)}{p_x(0) \cdot p_c(0)} + p(x_2, c)(0, 1) \log \frac{p(x_2, c)(0, 1)}{p_x(0) \cdot p_c(1)} + \\
p(x_2, c)(1, 0) \log \frac{p(x_2, c)(1, 0)}{p_x(1) \cdot p_c(0)} + p(x_2, c)(1, 1) \log \frac{p(x_2, c)(1, 1)}{p_x(1) \cdot p_c(1)}
\]

\[
= \frac{1}{12} \log \frac{1}{12} \cdot \frac{1}{12} + \frac{5}{12} \log \frac{5}{12} \cdot \frac{5}{12} + \frac{5}{12} \log \frac{5}{12} \cdot \frac{5}{12} + \frac{1}{12} \log \frac{1}{12} \cdot \frac{1}{12}
\]
Eliminating rare instances

- The class of each labelled instance, \((x_l, c^{(l)})\), is compared with the label assigned by a K-NN obtained with all instances except itself.

- If both labels coincide the instance is maintained in the file. Otherwise it is eliminated.
Hart condensation

Maintaining rare instances

- For each labelled instance, and following the storage ordering, consider a $K$-NN with only the previous instances to the one to be considered.
- If the true class and the class predicted by the $K$-NN are the same the instance is not selected.
- Otherwise (the true class and the predicted one are different) the instance is selected.
- The method depends on the storage ordering.
Outline

1. Introduction
2. The Basic $K$-NN
3. Extensions of the Basic $K$-NN
4. Prototype Selection
5. Summary
**K-nearest neighbor**

- **Intuitive** and easy to understand
- There is not an explicit model: *transduction* instead of *induction*
- **Variants** of the basic algorithm
- **Storage problems**: prototype selection
5. K-Nearest Neighbor

Pedro Larrañaga

Intelligent Systems Group
Department of Computer Science and Artificial Intelligence
University of the Basque Country

Madrid, 25th of July, 2006