

Nearest neighbor methods

Lecture 10

David Sontag
New York University

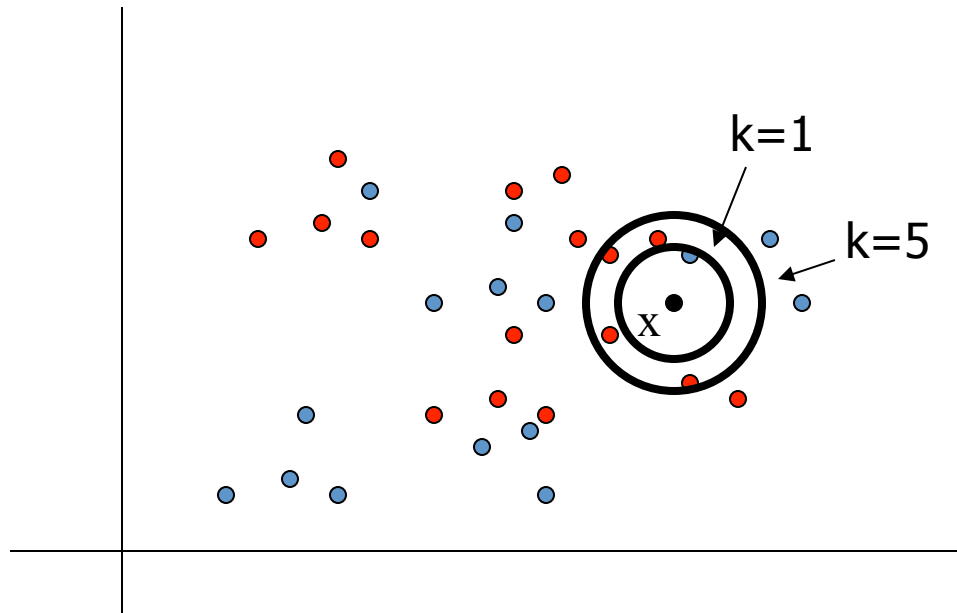
Slides adapted from Vibhav Gogate, Carlos Guestrin,
Mehryar Mohri, & Luke Zettlemoyer

Nearest Neighbor Algorithm

- Learning Algorithm:
 - Store training examples
- Prediction Algorithm:
 - To classify a new example \mathbf{x} by finding the training example (\mathbf{x}^i, y^i) that is *nearest* to \mathbf{x}
 - Guess the class $y = y^i$

K-Nearest Neighbor Methods

- To classify a new input vector x , examine the k -closest training data points to x and assign the object to the most frequently occurring class

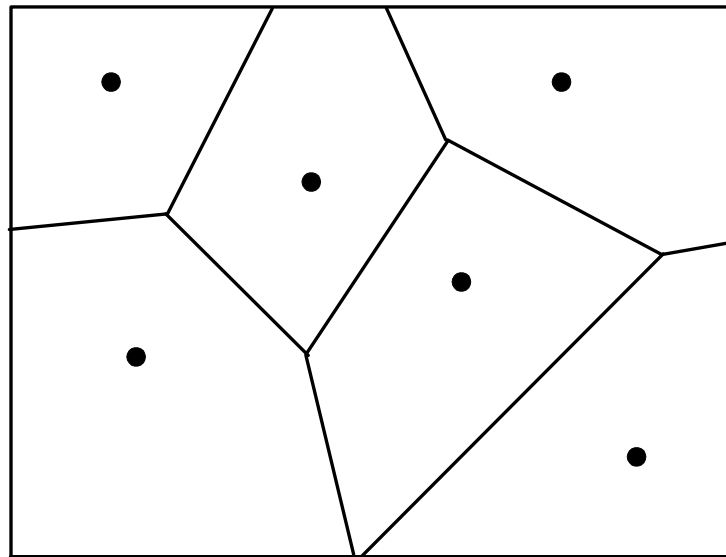


common values for k : 3, 5

Decision Boundaries

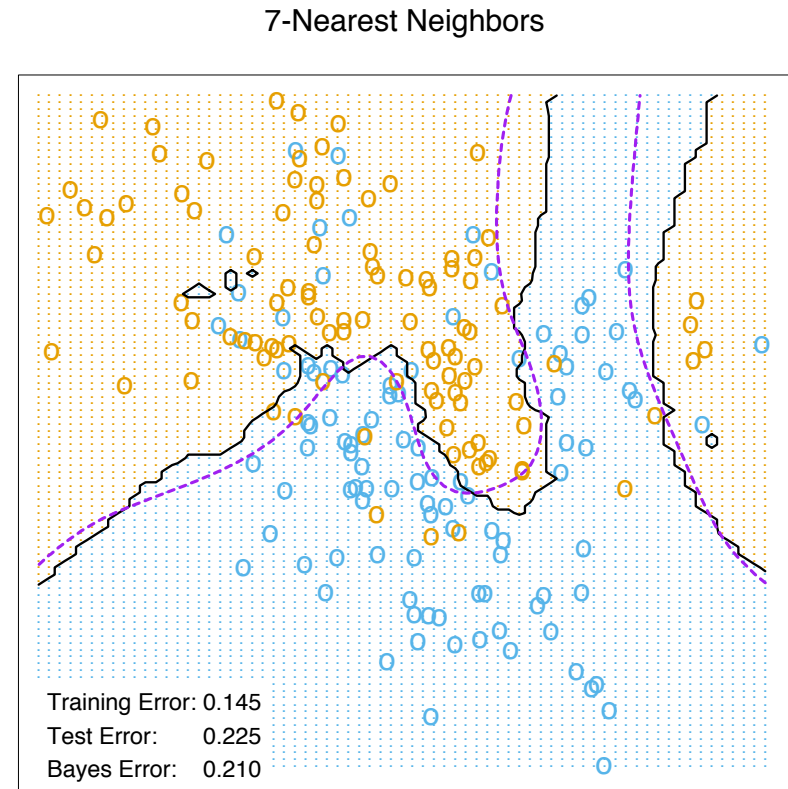
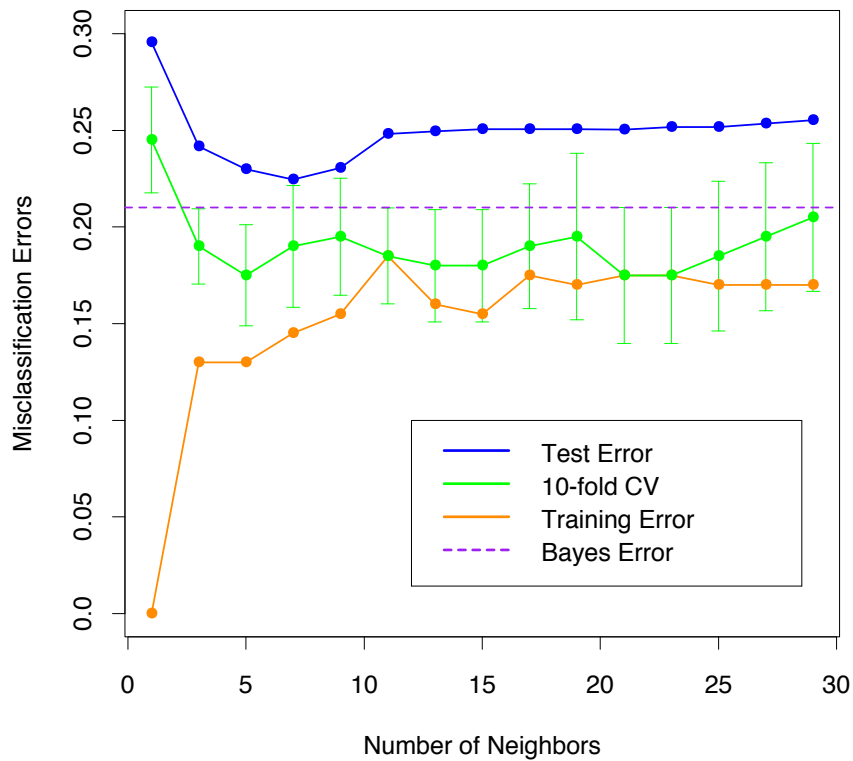
- The nearest neighbor algorithm does not explicitly compute decision boundaries. However, the decision boundaries form a subset of the Voronoi diagram for the training data.

1-NN Decision Surface



- The more examples that are stored, the more complex the decision boundaries can become

Example results for k-NN



[Figures from Hastie and Tibshirani, Chapter 13]

Nearest Neighbor

When to Consider

- Instance map to points in R^n
- Less than 20 attributes per instance
- Lots of training data

Advantages

- Training is very fast
- Learn complex target functions
- Do not lose information

Disadvantages

- Slow at query time
- Easily fooled by irrelevant attributes

Issues

- Distance measure
 - Most common: Euclidean
- Choosing k
 - Increasing k reduces variance, increases bias
- For high-dimensional space, problem that the nearest neighbor may not be very close at all!
- Memory-based technique. Must make a pass through the data for each classification. This can be prohibitive for large data sets.

Distance

- Notation: object with p measurements

$$\mathbf{x}^i = (x_1^i, x_2^i, \dots, x_p^i)$$

- Most common distance metric is *Euclidean* distance:

$$d_E(\mathbf{x}^i, \mathbf{x}^j) = \left(\sum_{k=1}^p (x_k^i - x_k^j)^2 \right)^{\frac{1}{2}}$$

- ED makes sense when different measurements are commensurate; each is variable measured in the same units.
- If the measurements are different, say length and weight, it is not clear.

Standardization

When variables are not commensurate, we can standardize them by dividing by the sample standard deviation. This makes them all equally important.

The estimate for the standard deviation of x_k :

$$\hat{\sigma}_k = \left(\frac{1}{n} \sum_{i=1}^n (x_k^i - \bar{x}_k)^2 \right)^{\frac{1}{2}}$$

where \bar{x}_k is the sample mean:

$$\bar{x}_k = \frac{1}{n} \sum_{i=1}^n x_k^i$$

Weighted Euclidean distance

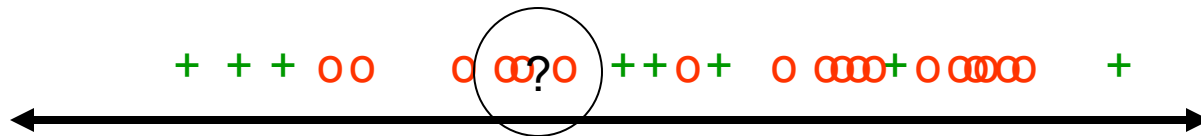
Finally, if we have some idea of the relative importance of each variable, we can weight them:

$$d_{\text{WE}}(i, j) = \left(\sum_{k=1}^p w_k (x_k^i - x_k^j)^2 \right)^{\frac{1}{2}}$$

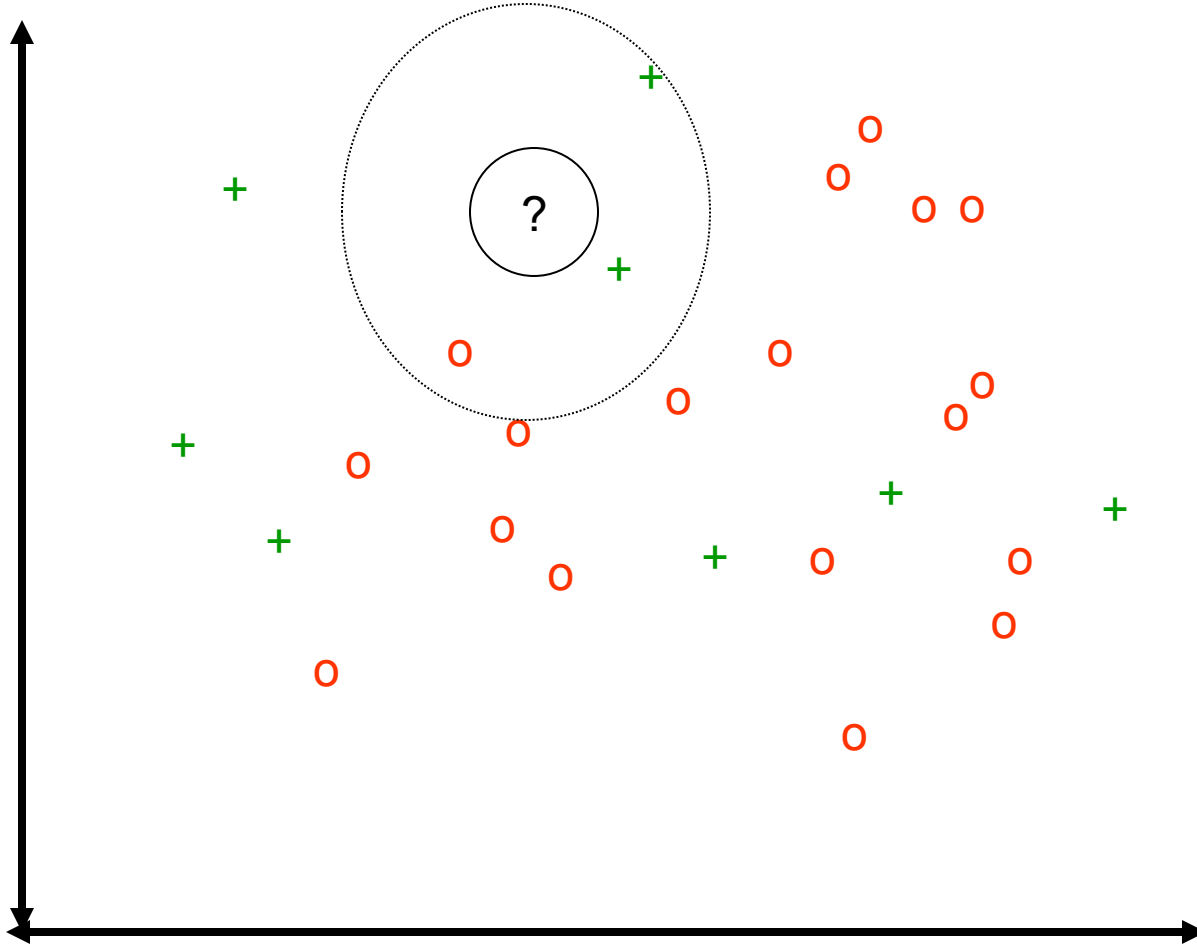
The Curse of Dimensionality

- Nearest neighbor breaks down in high-dimensional spaces because the “neighborhood” becomes very large.
- Suppose we have 5000 points uniformly distributed in the unit hypercube and we want to apply the 5-nearest neighbor algorithm.
- Suppose our query point is at the origin.
 - 1D –
 - On a one dimensional line, we must go a distance of $5/5000 = 0.001$ on average to capture the 5 nearest neighbors
 - 2D –
 - In two dimensions, we must go $\sqrt{0.001}$ to get a square that contains 0.001 of the volume
 - D –
 - In D dimensions, we must go $(0.001)^{1/D}$

K-NN and irrelevant features



K-NN and irrelevant features



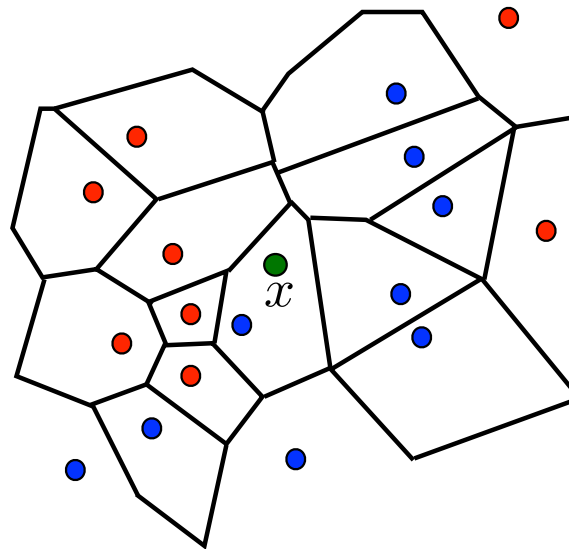
Nearest neighbor problem

- **Problem:** given sample $S = ((x_1, y_1), \dots, (x_m, y_m))$, find the nearest neighbor of test point x .
- general problem extensively studied in computer science.
- exact vs. approximate algorithms.
- dimensionality N crucial.
- better algorithms for small intrinsic dimension (e.g., limited doubling dimension).

Efficient Indexing: N=2

■ Algorithm:

- compute Voronoi diagram in $O(m \log m)$.
- **point location** data structure to determine NN.
- complexity: $O(m)$ space, $O(\log m)$ time.



KNN Advantages

- Easy to program
- No optimization or training required
- Classification accuracy can be very good; can outperform more complex models