Nearest Neighbor Predictors

COMPSCI 371D — Machine Learning

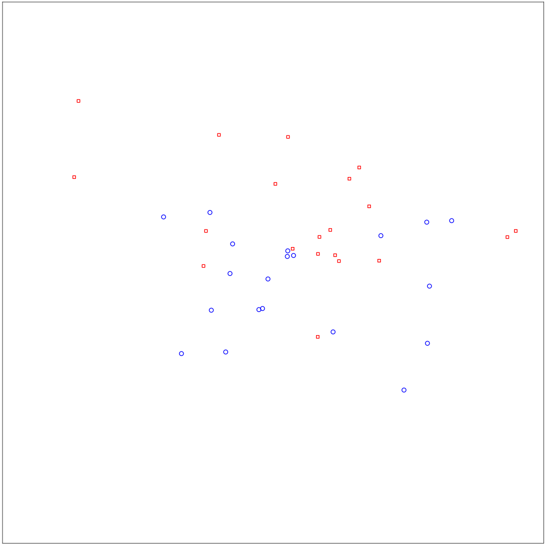
Outline

Nearest Neighbor Prediction

- NN is very simple: This is why we start here
- NN is very unusual:
    - No training!
    - Slow inference (using the predictor)
    - Y can be *anything*
    - Almost no difference between regression and classification
    - Hypothesis space hard to define

How it Works

- Given $T = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$
- Just store T (memorization)
- Need a distance in the data space X
- Perhaps $\Delta(\mathbf{x}, \mathbf{x}') = \|\mathbf{x} - \mathbf{x}'\|^2$
- Then, $h(\mathbf{x}) = y_{\nu(\mathbf{x})}$
  where $\nu(\mathbf{x}) \in \arg\min_{n=1,\ldots,N} \Delta(\mathbf{x}, \mathbf{x}_n)$
- Return the value y_n for the training point $\mathbf{x}_n$ that is nearest to $\mathbf{x}$

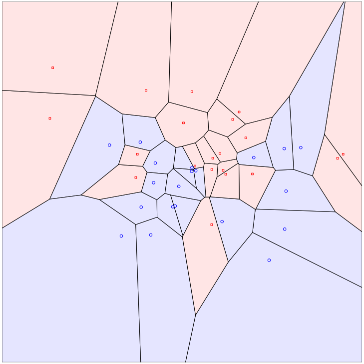


How to find $\nu(\mathbf{x})$?

$\nu(\mathbf{x}) = \arg\min_{n=1,\ldots,N} \Delta(\mathbf{x}, \mathbf{x}_n)$

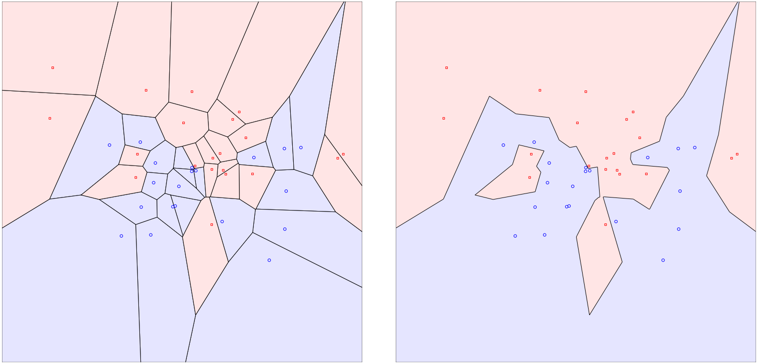
- Compute all $\Delta(\mathbf{x}, \mathbf{x}_n)$ and find the smallest
- $O(Nd)$ (where $\mathbf{x} \in \mathbb{R}^d$)
- Cannot do better *exactly*
- Can do better if we accept $\Delta(\mathbf{x}, \mathbf{x}_{\nu(\mathbf{x})}) < (1+\epsilon)\Delta(\mathbf{x}, \mathbf{x}_{\nu^*(\mathbf{x})})$ for some $\epsilon > 0$
- "Approximate NN" uses *k*-*d* trees, R-trees, locality sensitive hashing

The Voronoi Diagram

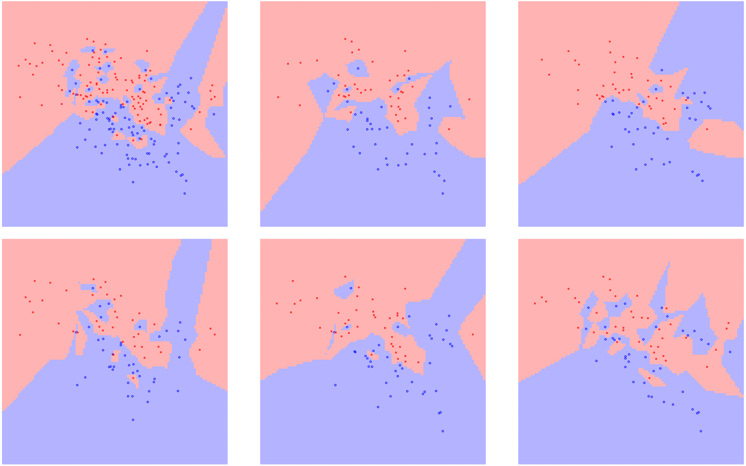


- Only conceptual, or for $d = 2, 3$, maybe 4
- $\Theta(N \log N + N^{\lceil d/2 \rceil})$

Decision Boundary



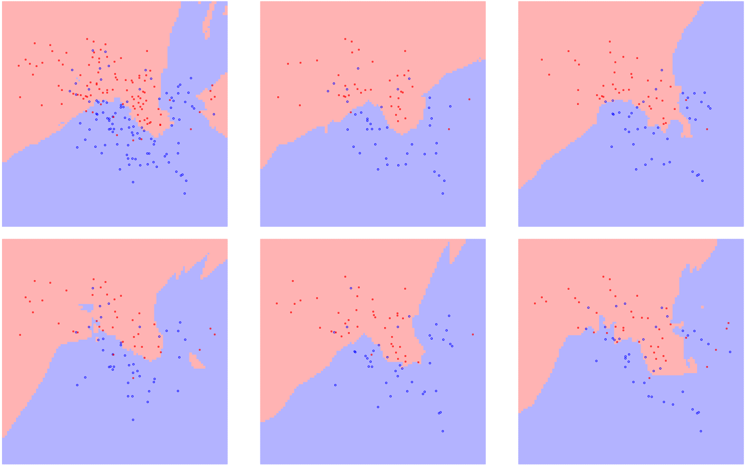
Overfitting



k Nearest Neighbors

- Retrieve the k nearest neighbors $\mathbf{x}_1, \ldots, \mathbf{x}_k$ of $\mathbf{x}$
- Return a *summary* of the corresponding $y_1, \ldots, y_k$
- Classification summary: majority
- Regression summary: Mean, median

Less Overfitting ($k = 9$)



A Simple Regression Example, $\mathbb{R} \to \mathbb{R}$