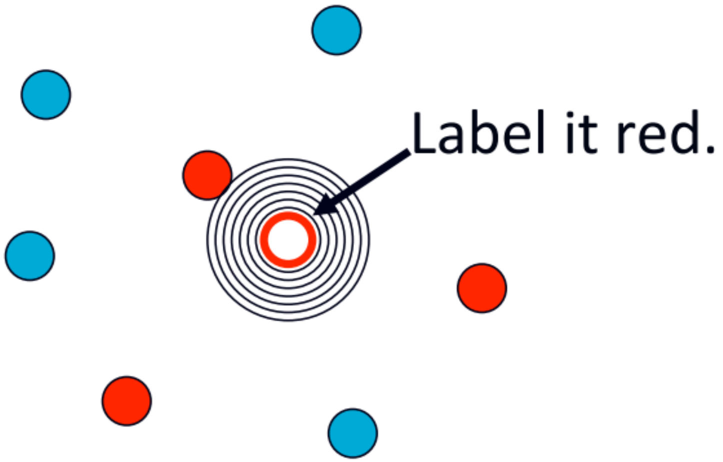
k-Nearest Neighbor

Machine Learning (AIM 5002-41)

Joon Hee Choi
Sungkyunkwan University

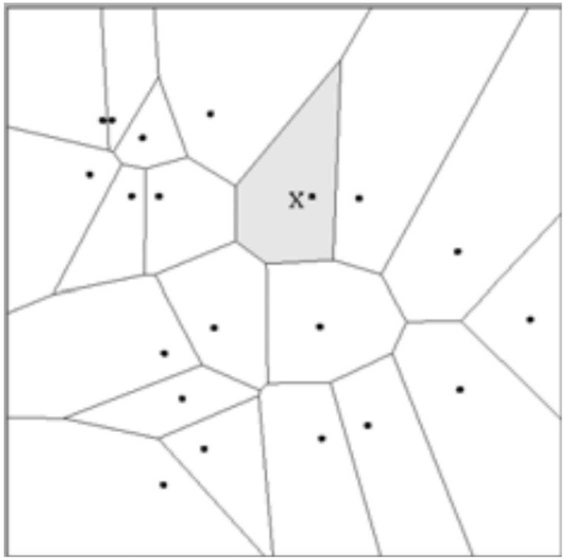
1-Nearest Neighbor

- One of the simplest of all machine learning classifiers
- Simple idea: label a new point the same as the closest known point



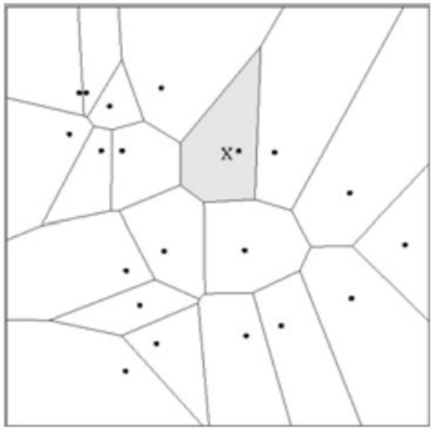
1-Nearest Neighbor

- A type of instance-based learning
  - Also known as “memory-based” learning
- Forms a Voronoi tessellation of the instance space

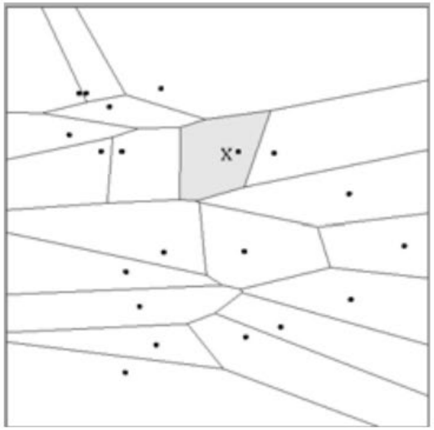


Distance Metrics

- Different metrics can change the decision surface



Dist(**a**,**b**) =$(a_1 - b_1)^2 + (a_2 - b_2)^2$



Dist(**a**,**b**) =$(a_1 - b_1)^2 + (3a_2 - 3b_2)^2$

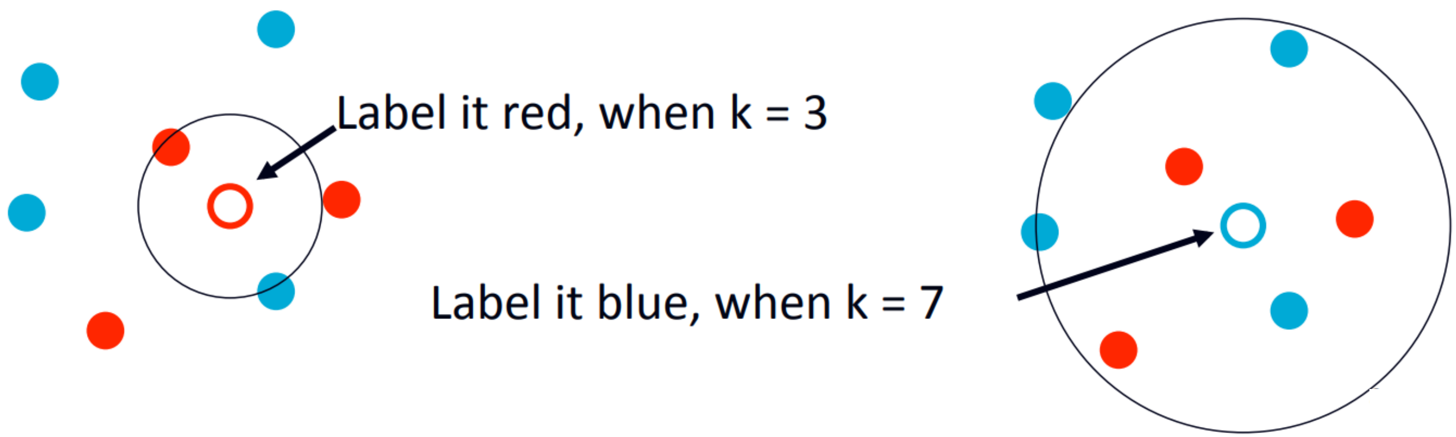
- Standard Euclidean distance metric:
  - **Two-dimensional**: $Dist(a, b) = \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2}$
  - **Multivariate**: $Dist(a, b) = \sqrt{\sum (a_i - b_i)^2}$

1-NN's Four Aspects as an Instance-Based Learner:

1. A distance metric
   – *Euclidian*
2. How many nearby neighbors to look at?
   – *One*
3. A weighting function (optional)
   – *Unused*
4. How to fit with the local points?
   – *Just predict the same output as the nearest neighbor.*

k-Nearest Neighbor

- Generalizes 1-NN to smooth away noise in the labels
- A new point is now assigned the most frequent label of its *k* nearest neighbors



k-Nearest Neighbor Classification

Given a training dataset $\mathcal{D} = \{y^{(n)}, \boldsymbol{x}^{(n)}\}_{n=1}^{N}$, $y \in \{1, \dots, C\}$, $\boldsymbol{x} \in \mathbb{R}^m$ and a test input $\boldsymbol{x}_{test}$, predict the class label, $\hat{y}_{test}$:

1) Find the closest k points in the training data to $\boldsymbol{x}_{test}$

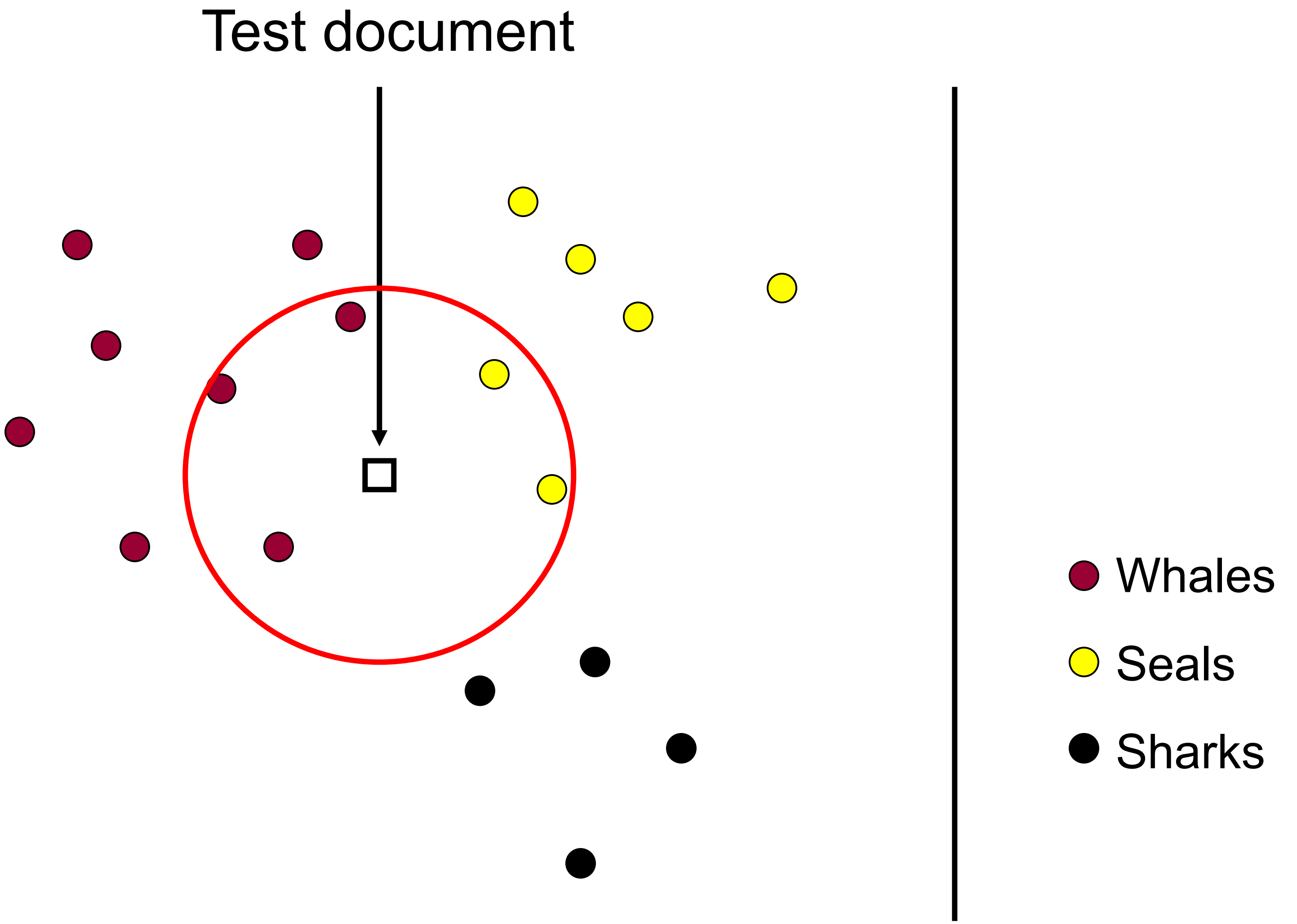
$$\mathcal{N}_k(\boldsymbol{x}_{test}, \mathcal{D})$$

2) Return the class label of that closest point

$$\begin{aligned}
\hat{y}_{test} &= \operatorname*{argmax}_c p(Y = c \mid \boldsymbol{x}_{test}, \mathcal{D}, k) \\
&= \operatorname*{argmax}_c \frac{1}{k} \sum_{i \in \mathcal{N}_k(\boldsymbol{x}_{test}, \mathcal{D})} \mathbb{I}\left(y^{(i)} = c\right) \\
&= \operatorname*{argmax}_c \frac{k_c}{k}
\end{aligned}$$

where k_c is the number of the k-neighbors with class label c

k-NN Classifier (*k*=5)



Reference

- https://www.cs.cmu.edu/~10315/