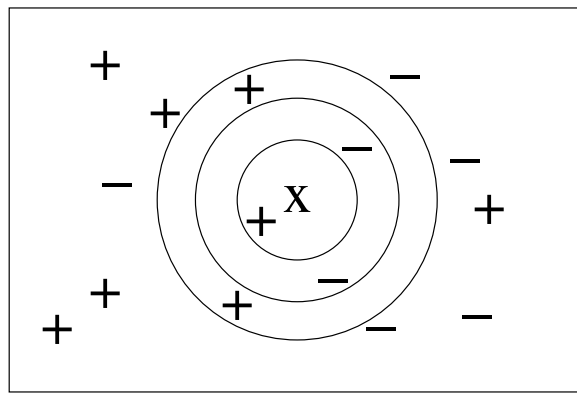


## Nearest Neighbors

The  $k$ NN algorithm predicts the outcome  $y$  for an example  $x$  by finding the  $k$  labeled examples  $(x_i, y_i) \in D$  closest to  $x$  and returning:

- (classification) the most common outcome  $y$ .
- (regression) the average outcome  $y$ .



---

## Algorithms

$k$ NN-LEARN( $D$ )

$h.D \leftarrow D$

$h.d \leftarrow$  distance measure based on  $D$

**return**  $h$

$k$ NN-PREDICT( $x, h$ )

$N \leftarrow$  the  $k$  examples in  $h.D$  closest to  $x$   
as measured by  $h.d$

**return** most common/average  $y$  in  $N$

## Details of $k$ NN

$x$  = example to be classified

$(x_1, y_1), \dots, (x_k, y_k)$  =  $x$ 's  $k$  nearest neighbors

$d(x, x_i)$  = distance between  $x$  and  $x_i$

Normally, closeness is measured by Euclidean distance  $\|x - x_i\|_2$ , where

$$\|(a_1, \dots, a_n)\|_2 = \sqrt{\sum_{j=1}^n a_j^2}$$

Sometimes, Manhattan/Hamming distance is used.

$$\|(a_1, \dots, a_n)\|_1 = \sum_{j=1}^n |a_j|$$

---

## Scaling

Attributes can have widely different ranges, e.g., Aluminum and Refractive Index. Consider:

- Normalization. Rescale attribute so that its minimum is 0 (or  $-1$ ) and its maximum is 1.
- Standardization. Rescale attribute so that its mean is 0 and its standard deviation is 1.

Attributes can be redundant, e.g., Petal Length and Petal Width. Consider Mahalanobis distance (Duda/Hart/Stork).

## Other Distance Issues

Attributes can be irrelevant. The textbook hints at sophisticated ways to address this issue, but consider multiplying an attribute times its correlation with the outcome (after scaling).

Nominal attributes are either equal or different. Consider being different as a difference of 1, or convert to binary attributes.

Attribute values can be missing. Consider using some fixed value for the difference.

---

## Distance-Weighting

Rather than treating each neighbor equally, give more weight to closer neighbors. Predict with:

- (classification) the class with the highest sum of weights.
- (regression) the weighted average, e.g.,

$$\frac{\sum_{i=1}^k y_i / d(x, x_i)}{\sum_{i=1}^k 1 / d(x, x_i)}$$

To avoid division by 0, add a small value to  $d$ .

## Convergence of KNN

1NN error converges to within a factor of two of optimal.

Suppose within a given region  $R$ :

$p$  = probability of positive label

$q = 1 - p$  = probability of negative label

If  $x$  is within  $R$ , then as the sample size increases, the probability that its NN  $x'$  is also in  $R$  approaches 1.

---

### Convergence of KNN (continued)

$x$  and  $x'$  are both positive with prob.  $p^2$

$x$  and  $x'$  are both negative with prob.  $q^2$

$x$  and  $x'$  diff. labels with prob.  $1 - p^2 - q^2$

Suppose  $p > q$ , then the optimal error is  $q$ .

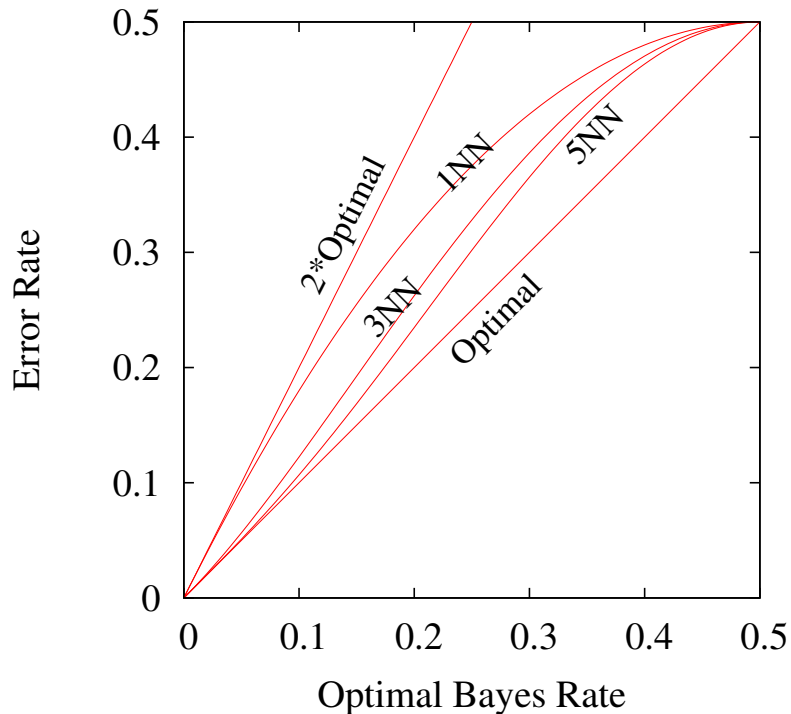
The NN probability of error is:

$$1 - p^2 - q^2 = 1 - (1 - q)^2 - q^2 = 2q - 2q^2$$

which is less than  $2q$ .

# Convergence of $k$ NN

Error Rates for  $k$ NN as Sample Size Increases



Note: Convergence can be very slow.

---

## Issues of $k$ NN

For basic  $k$ NN, no training is needed, but might be desired for scaling or selecting training exs.

An open problem is more efficient algorithms to find NN.

Roughly, case-based and analogical learning are based on closeness of symbolic descriptions.

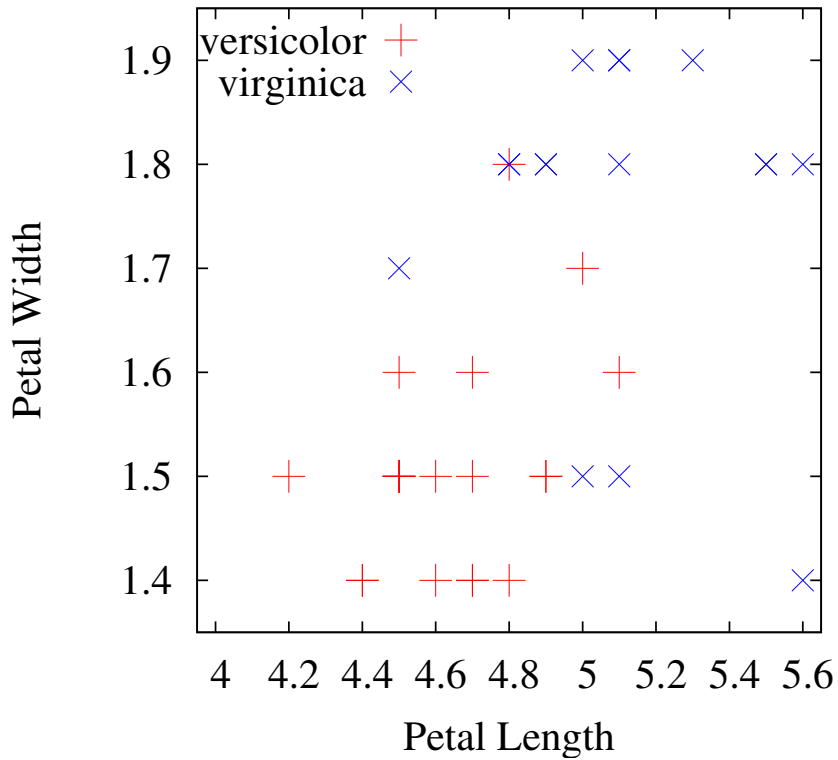
What is the inductive bias of  $k$ NN?

Does  $k$ NN have an overfitting problem?

Will increasing  $k$  always improve performance?

# Iris Dataset Example

A Region of the Two-Attribute Iris Dataset



# Glass Dataset Example

A Region of the Two-Attribute Glass Dataset

