

Machine Learning

KNN Classifier and r-fold Cross Validation

Indian Institute of Information Technology
Sri City, Chittoor



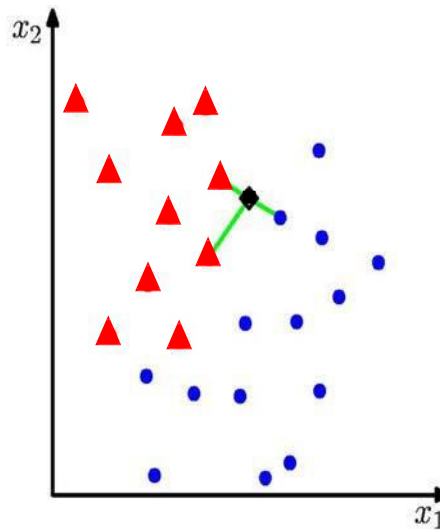
Today's Agenda

- K-Nearest Neighbor Classifier
- r-fold Cross validation

K-Nearest Neighbor Classifier

- Algorithm
 - For each test point, x , to be classified, find the K nearest samples in the training data
 - Classify the point, x , according to the majority vote of their class labels

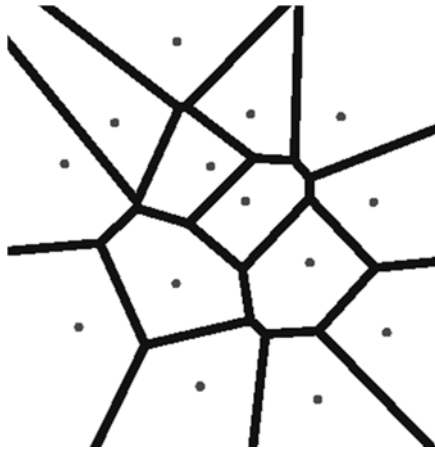
- e.g. $K = 3$



- applicable to multi-class case

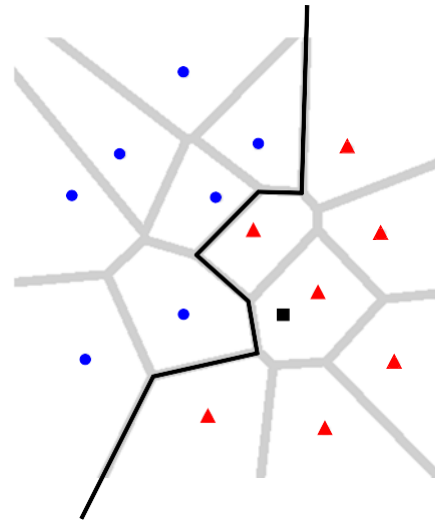
K-Nearest Neighbor Classifier

$K = 1$



Voronoi diagram:

- partitions the space into regions
- boundaries are equal distance from training points

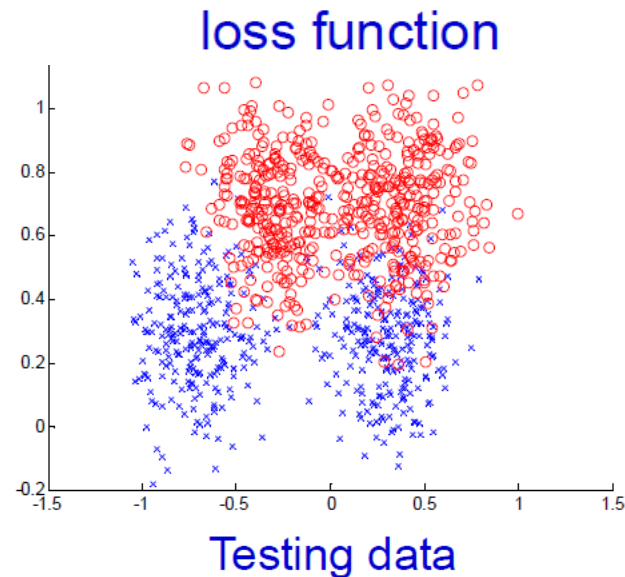
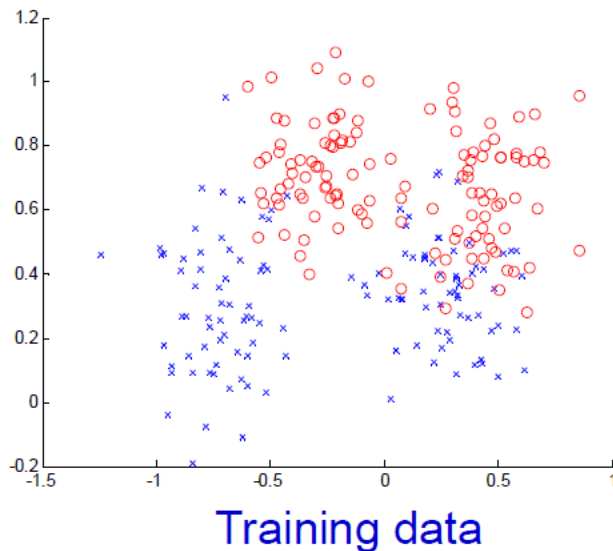


Classification boundary:

- non-linear

K-Nearest Neighbor Classifier

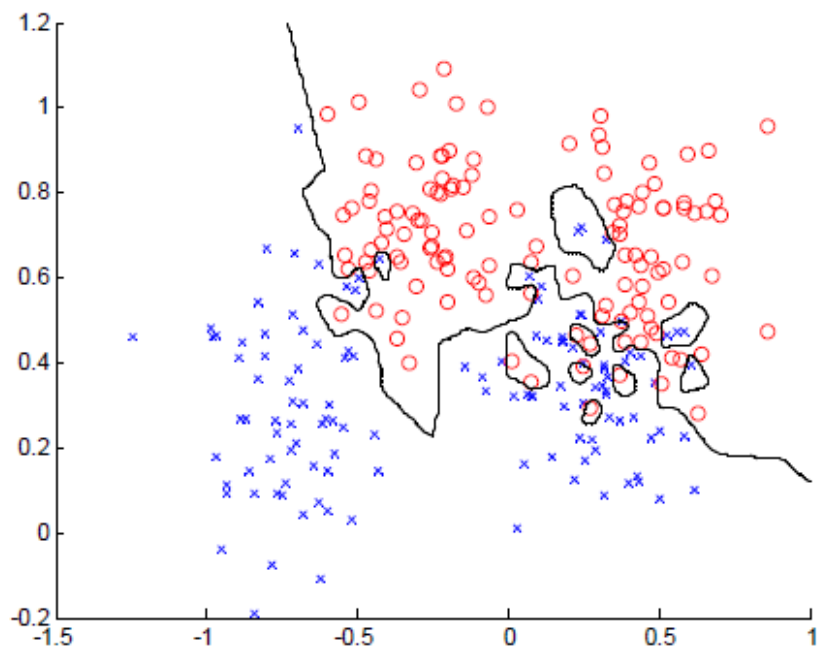
- Assume that the training examples are drawn independently from the set of all possible examples.
- This makes it very unlikely that a strong regularity in the training data will be absent in the test data.
- Measure classification error as = $\frac{1}{N} \sum_{i=1}^N \underbrace{[y_i \neq f(\mathbf{x}_i)]}_{\text{loss function}}$ The “risk”



K-Nearest Neighbor Classifier

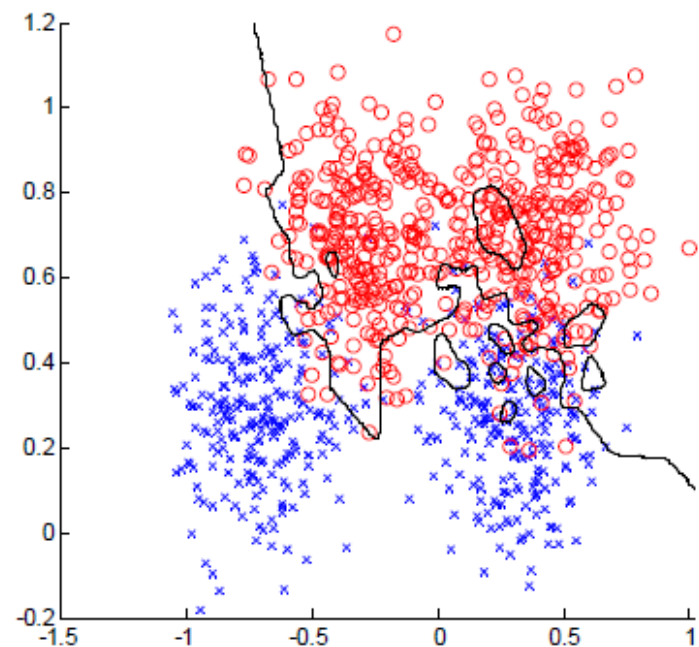
K=1

Training data



error = 0.0

Testing data

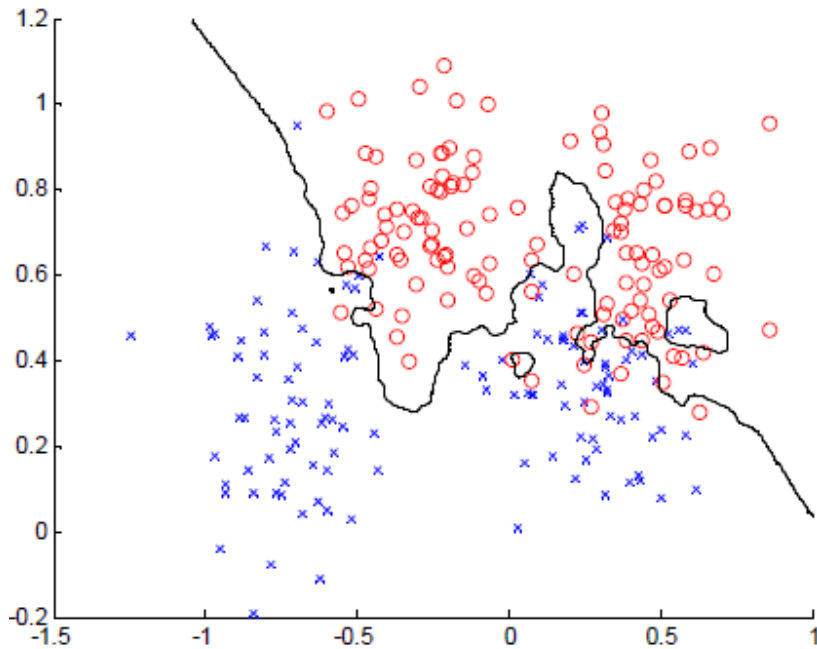


error = 0.15

K-Nearest Neighbor Classifier

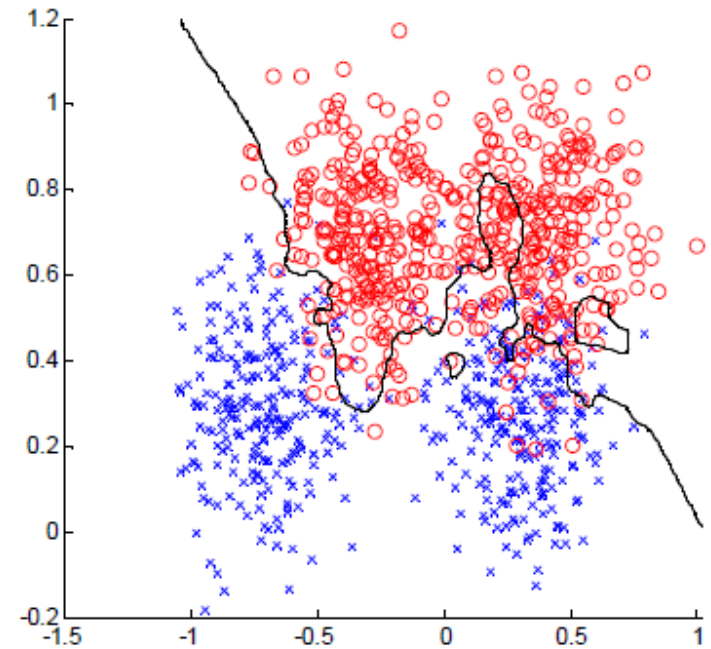
K=3

Training data



error = 0.0760

Testing data

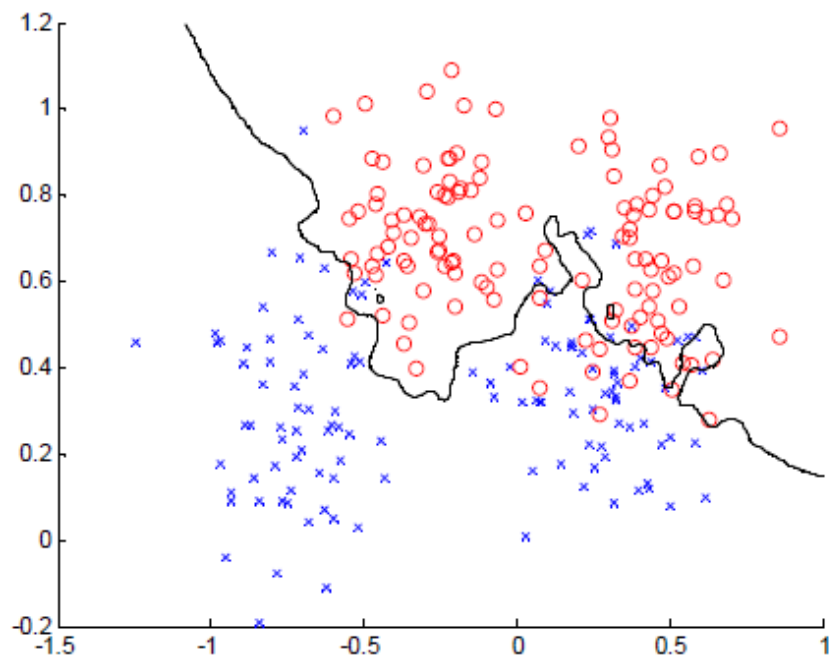


error = 0.1340

K-Nearest Neighbor Classifier

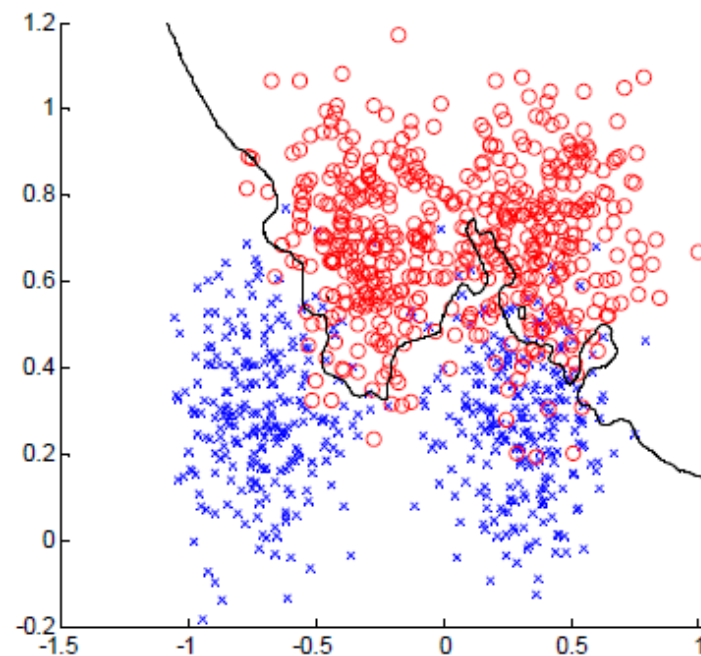
K=7

Training data



error = 0.1320

Testing data

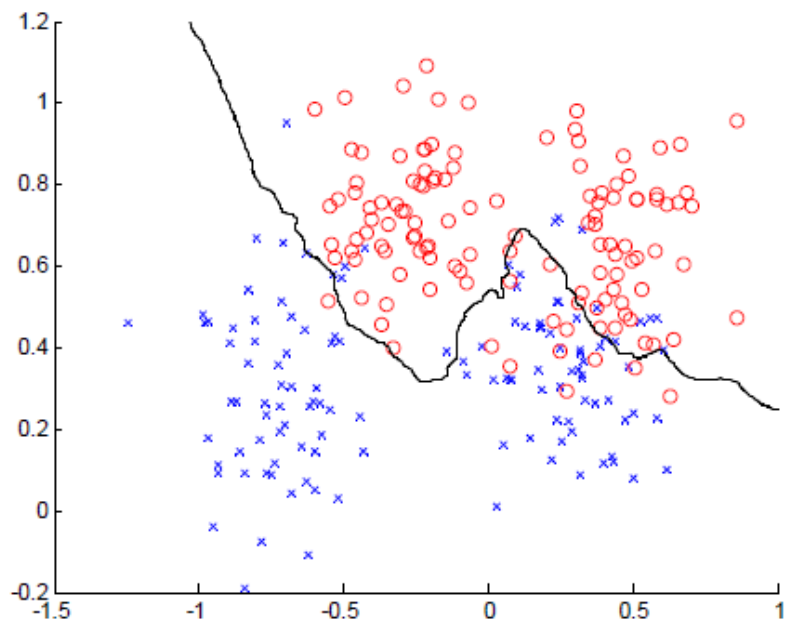


error = 0.1110

K-Nearest Neighbor Classifier

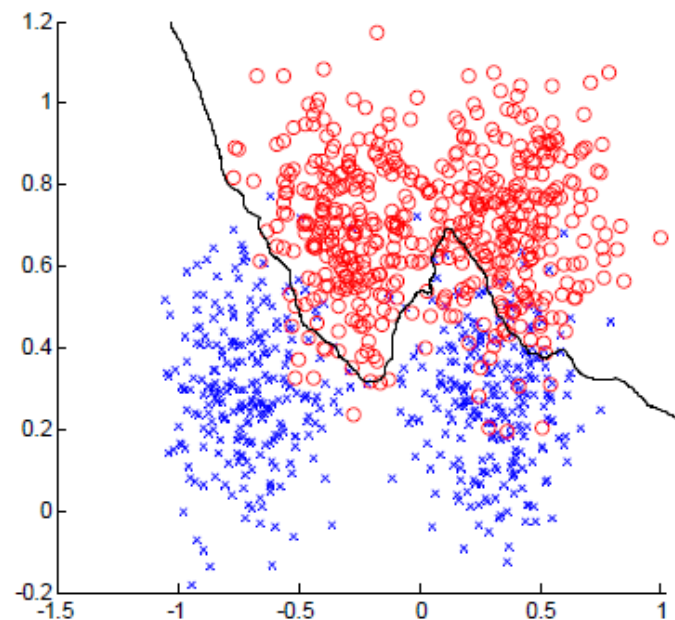
K=21

Training data



error = 0.1120

Testing data



error = 0.0920

K-Nearest Neighbor Classifier

- The real aim of supervised learning is to do well on test data that is not known during learning
- Choosing the values for the parameters that minimize the loss function on the training data is not necessarily the best policy
- We want the learning machine to model the true regularities in the data and to ignore the noise in the data.

K-Nearest Neighbor Classifier

As K increases:

- Classification boundary becomes smoother
- Training error can increase

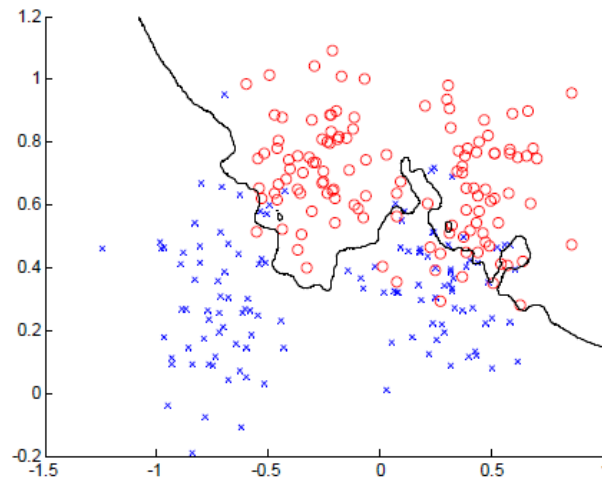
Choose (learn) K by cross-validation

- Split training data into training and validation
- Hold out validation data and measure error on this

K-Nearest Neighbor Classifier

Advantages:

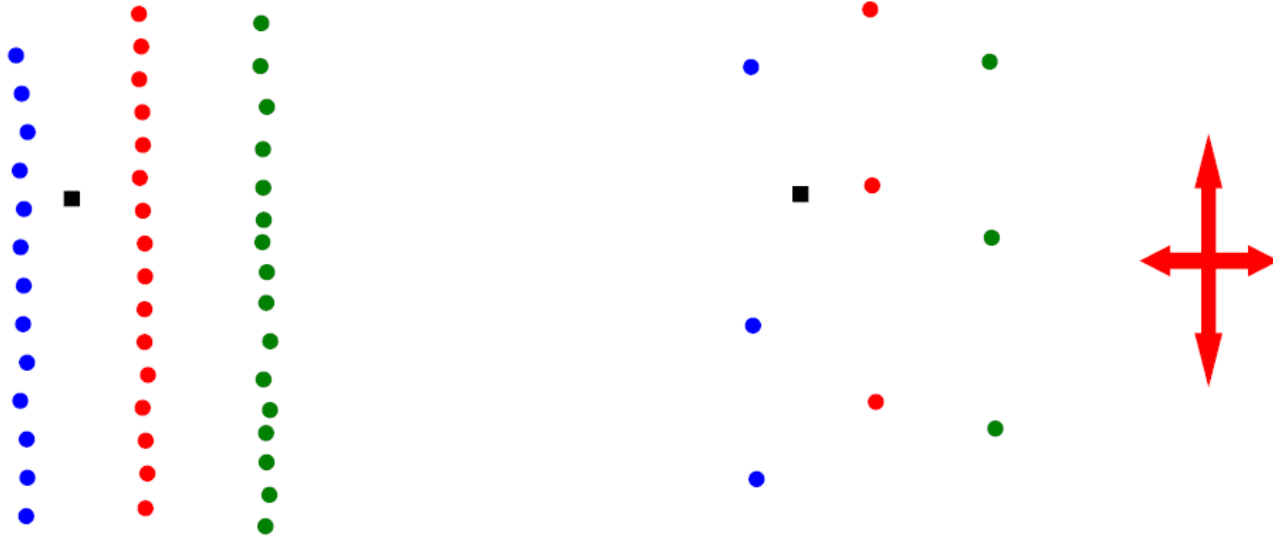
- K-NN is a simple but effective classification procedure
- Applies to multi-class classification
- Decision surfaces are non-linear
- Quality of predictions automatically improves with more training data
- Only a single parameter, K ; easily tuned by cross-validation



K-Nearest Neighbor Classifier

Disadvantages:

- What does nearest mean? Need to specify a distance metric.
- Computational cost: must **store** and **search** through the entire training set at test time. Can alleviate this problem by thinning, and use of efficient data structures like KD trees.



Cross Validation

- How to find appropriate k value for k -NNC.
- Some improvements to k -NNC

Cross Validation

r-fold cross validation:

1. Partition the training set into r blocks. Let these are D_1, D_2, \dots, D_r .
2. For $i = 1$ to r do
 - I. Consider $D - D_i$ as the training set and D_i as the validation set.
 - II. For a range of k values (say from 1 to m) find the error rates on the validation set.
 - III. Let these error rates are $e_{i1}, e_{i2}, \dots, e_{im}$
3. Take $e_i = \text{mean of } \{e_{1i}, e_{2i}, \dots, e_{ri}\}$, for $i = 1$ to m .
4. $k \text{ value} = \underset{j}{\operatorname{argmin}} \{e_1, e_2, \dots, e_j, \dots, e_m\}$

Cross Validation

- One should not use *the test set* to decide the value of K .
- Test set should be used only after fixing K , to get the final *error-rate* for the classifier.
- Cross validation is only to fix the value of parameters like K . So the error rates on validation sets should be called *validation error rates*.

Thank You: Question?