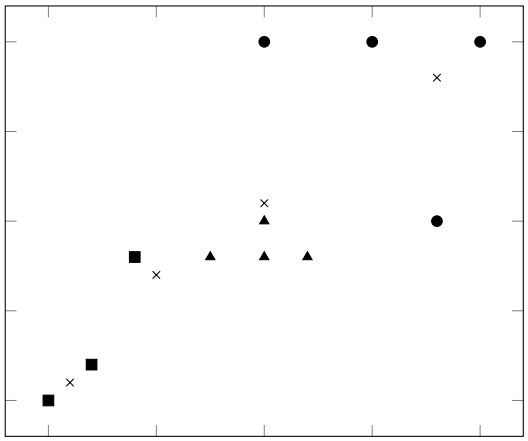
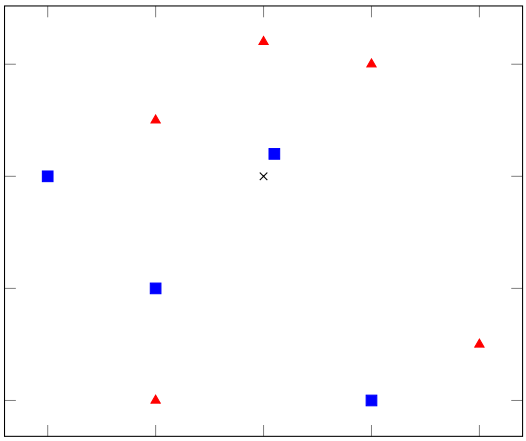
Nearest Neighbor Classifier

Question 1. When we say this is a *classification* problem, what does this mean? How does that contrast with a *regression* problem?

Question 2. When we say this is a *supervised learning* problem, what does that mean? How does it differ from an *unsupervised learning* problem?

Here are two examples of *training data* for a nearest neighbor classifier. The circles, squares, and triangles are *labels* for the associated data points. The spots marked with an x are *not* part of the training data, but will be used to illustrate the algorithm.





Question 3. Each data point marked with an x is a *query point*. For each x in the diagram on the left hand side, if it were instead one of the three other shapes, which shape do you suppose it would be? Why?

Question 4. What if we instead had the following data and *query point*? How would you classify it and why?

A **nearest neighbor** classifier works as follows:

Input:

- An integer, k
- A set of training examples, D
- A distance measure function, d

Algorithm:

for each test instance $z = (\boldsymbol{x}', y')$: **do**
  Compute $d(\boldsymbol{x}', \boldsymbol{x})$ between z and all $(\boldsymbol{x}, y) \in D$
  Select $D_z \subseteq D$, the k closest to z.
  $y' \leftarrow \arg\max_v \sum_{(\boldsymbol{x}_i, y_i) \in D_z} I(v = y_i)$

Question 5. The nearest neighbor classifier is described as a *lazy learner*, in contrast to a *rote learner*. What do these terms mean?

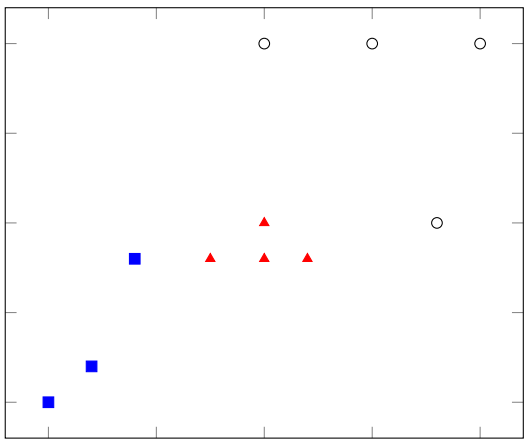
Measuring the performance of a classifier

We have two related terms we measure in evaluating a classifier:

- *Accuracy* is the percent of test points correctly classified.
- The *error rate* is the percent of test points *incorrectly* classified.

We often have to create our model before we have access to query points.

Suppose we have the following training data and wish to build a nearest neighbor classifier:

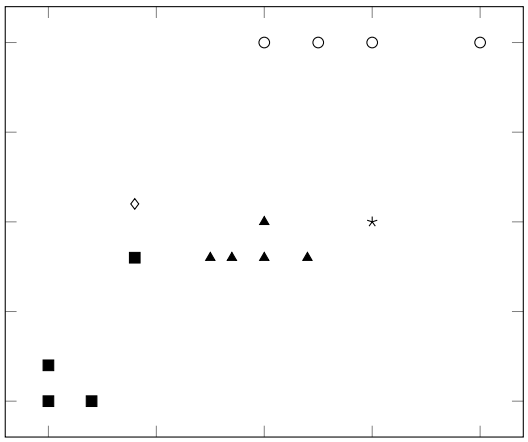


Question 6. If we use this training data directly, and use each point as a query point, what do we get for accuracy and error rate? Would another value of k help?

Question 7. What if we leave one out? That is, what if, for each point, we omit it from the training data and treat it as a query point? We do this independently for each point.

Exercises

Questions 8 through 11 deal with the following data, where squares, triangles, and open circles are three different classes of data in the training set and the diamond (♢) and star (*) are test points.



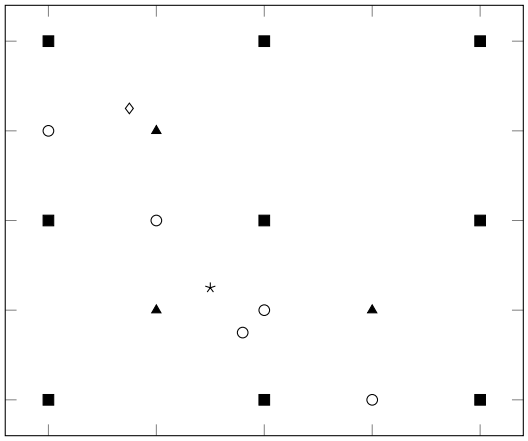
Question 8. Suppose I use all the training data as the full validation set (disregarding the two specially-marked test items) to classify the data using a KNN with k=1. How many of the 13 points will be correctly classified?

Question 9. For the KNN classifier with k=1, how many training data points will be misclassified with leave-one-out heuristic?

Question 10. What is the smallest value of k to always classify the diamond as class triangle?

Question 11. What label will we predict for star with $k = 3$?

Questions 12 through 14 deal with the following data, where squares, triangles, and open circles are three different classes of data in the training set and the diamond (♢) and star (*) are test points.



Question 12. For the KNN classifier with $k = 1$, how many **circles** from the training data points will be misclassified with leave-one-out heuristic?

Question 13. What is the smallest of the following values of k with which we will always classify the diamond as class square?

Question 14. What label will we predict for star with $k = 5$?