

## Kthperclass Classifier Description

### Introduction

The kthperclass classifier provides a way of classifying a test feature vector using a set of training feature vectors and associated labels. It has a parameter “k” that determines how many training vectors (per class) are used. It also has a distance metric and a rule for breaking ties. For the special case of k=1, it becomes equivalent to the k-nearest-neighbor (kNN) classifier.

### Classifier description

The kthperclass classifier is similar to the kNN classifier. While the kNN classifier first finds the k nearest neighbors regardless of class, the kthperclass classifier finds the k nearest neighbors for each class. Thus, for the first class, it would find the k nearest neighbors. The figure-of-merit (FOM) for the first class would then be the distance between the test vector and the kth nearest neighbor. Similarly, for the remaining classes, the k nearest neighbors would be found, and the distance to the kth nearest neighbor would give the FOM. The class with the best (smallest) FOM would determine the detected class.

### Distance metric

While a number of distance metrics can be used, here we will use the “Manhattan” or “cityblock” metric given by

$$d_c(\mathbf{x}, \mathbf{y}) = |x_1 - y_1| + |x_2 - y_2| + \cdots + |x_\ell - y_\ell|,$$

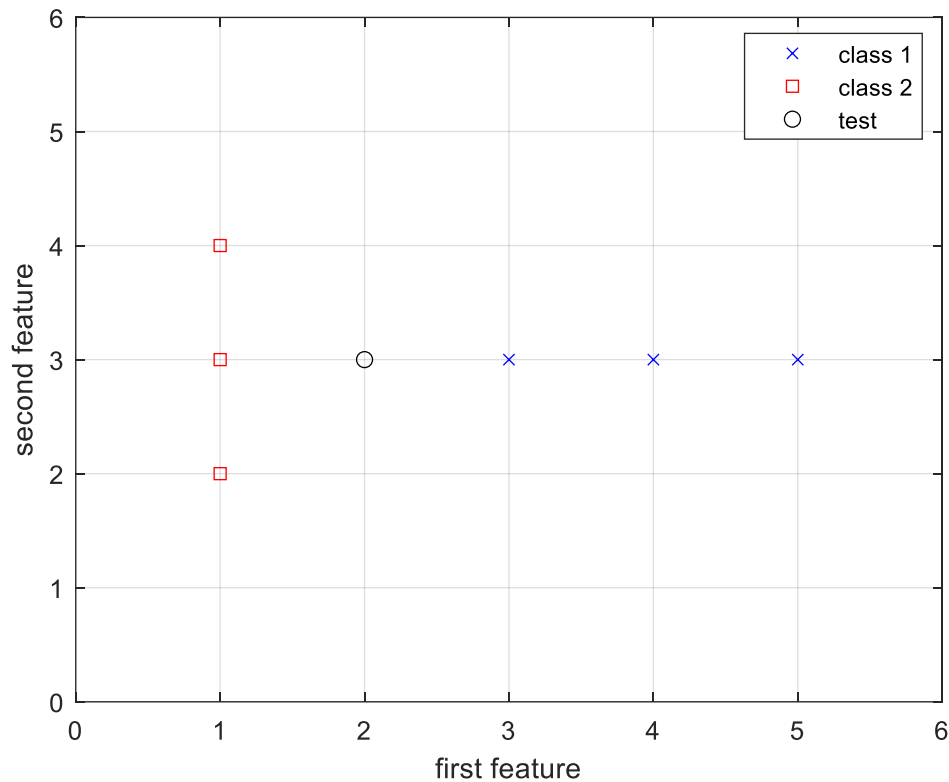
where  $\mathbf{x}$  and  $\mathbf{y}$  are feature vectors. For example, the distance between [4.2 1.1] and [4.0 1.4] is  $0.2 + 0.3 = 0.5$ . Observe that for length-2 vectors, such a distance is easily computed manually from a scatter plot.

### Breaking ties

Ties happen when the best FOM results from more than one class. While a number of tie breaking rules can be used, here we will use the “smallest” rule, which allocates the tie to the class with the smallest index. This assumes that there is an index associated with each class. For example, we can associate index 1 with setosa, index 2 with versicolor, and index 3 with virginica for the iris data set. In this case, if there is a tie between versicolor and virginica, then versicolor wins.

### Example

Consider the example specified in Figure 1. There are three training vectors for each of two classes. Also, there is a test vector. For ease of notation, row vectors will be used.



**Figure 1.** Example.

For  $k = 1$ , the nearest vector for class 1 is  $[3 \ 3]$  and the cityblock distance to test vector  $[2 \ 3]$  is 1. For class 2 the nearest vector is  $[1 \ 3]$  and the distance is also 1. As there is a tie, class 1 becomes the detected class.

For  $k = 2$ , the second-nearest vector for class 1 is  $[4 \ 3]$ , giving a distance metric of 2. For class 2, the second-nearest vector is either  $[1 \ 2]$  or  $[1 \ 4]$ , and the distance is 2. Again there is a tie, so class 1 is the detected class.

For  $k = 3$ , the third-nearest vector for class 1 is  $[5 \ 3]$ , giving a distance metric of 3. For class 2, the third-nearest vector is either  $[1 \ 2]$  or  $[1 \ 4]$ , and the distance is 2. As distance 2 is less than distance 3, class 2 is the detected class.