CLASSIFICATION: THE NEAREST-NEIGHBORS ALGORITHM (K-NN)

AP

FROM THE INTRODUCTION:

1. Classification and class probability

Instance:

- a collection (dataset) of datapoints from $\mathbf{X}$
- a classification system $C = \{c_1, c_2, \ldots c_r\}$

Solution: classification function $\gamma : \mathbf{X} \to C$

Measure: misclassification

BINARY CLASSIFICATION

$r = 2$: positive and negative.

Misclassification is described by the *confusion matrix*, which scores the result of classification against labeled examples.

BINARY CLASSIFICATION IN 2D

With just two numerical dimensions, datapoint similarity can be interpreted as simple Euclideian distance.

Being very close $\Longleftrightarrow$ being very similar

Q: are 4 and 6 more similar to each other than -1 and +1?

Assumption: small changes in the values won't alter the classification, close points will receive the same classification

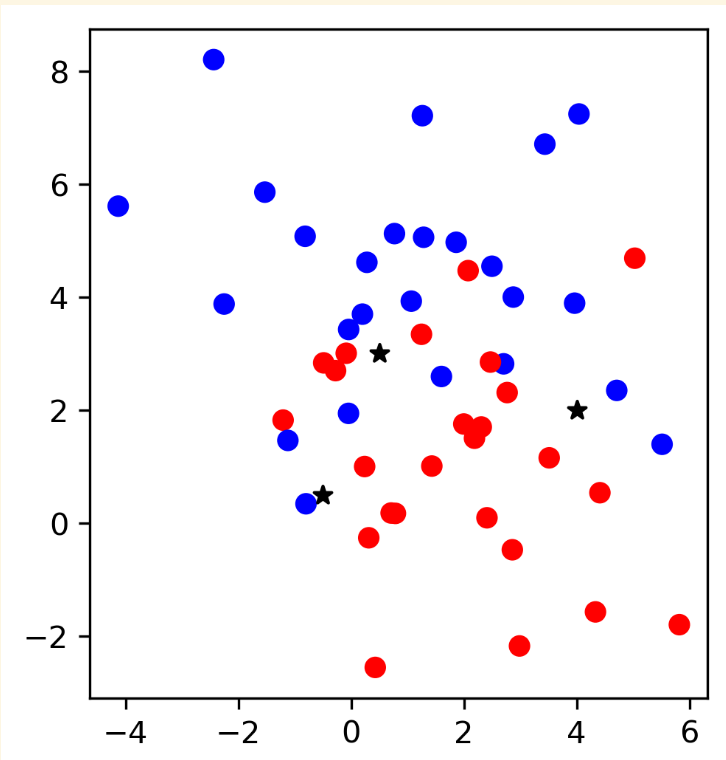
if a point is in close distance to a labeled one then assign the same class

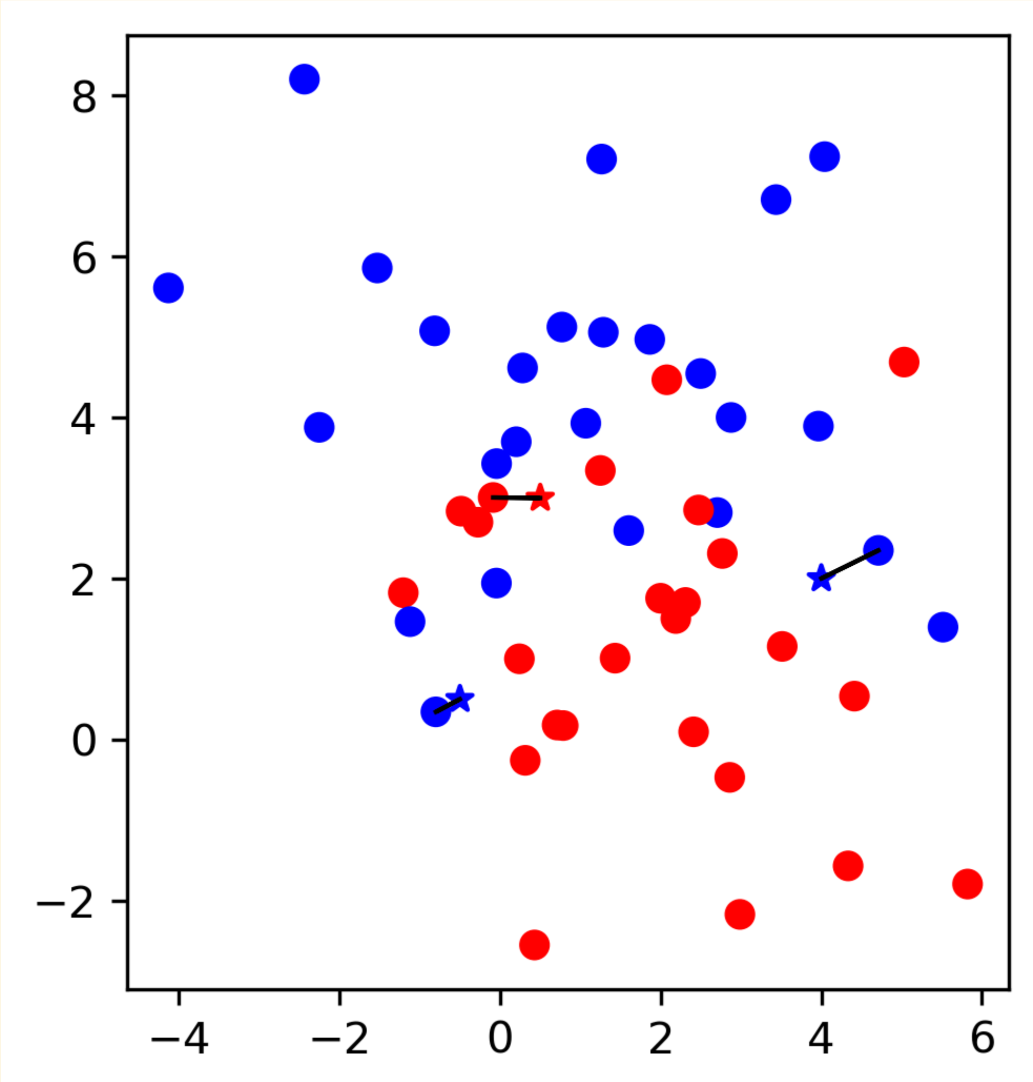
THE NEAREST NEIGH. ALGORITHM

Take a set of labeled points (the examples), all others are *blank* at the moment.

Whenever a blank point has a nearest neighbor datapoint which is labeled, give it the same label

This is the NN, or 1-NN algorithm.





$$\gamma(\mathbf{x}) = y_i, i = \text{argmin}_j ||\mathbf{x}_j - \mathbf{x}||$$

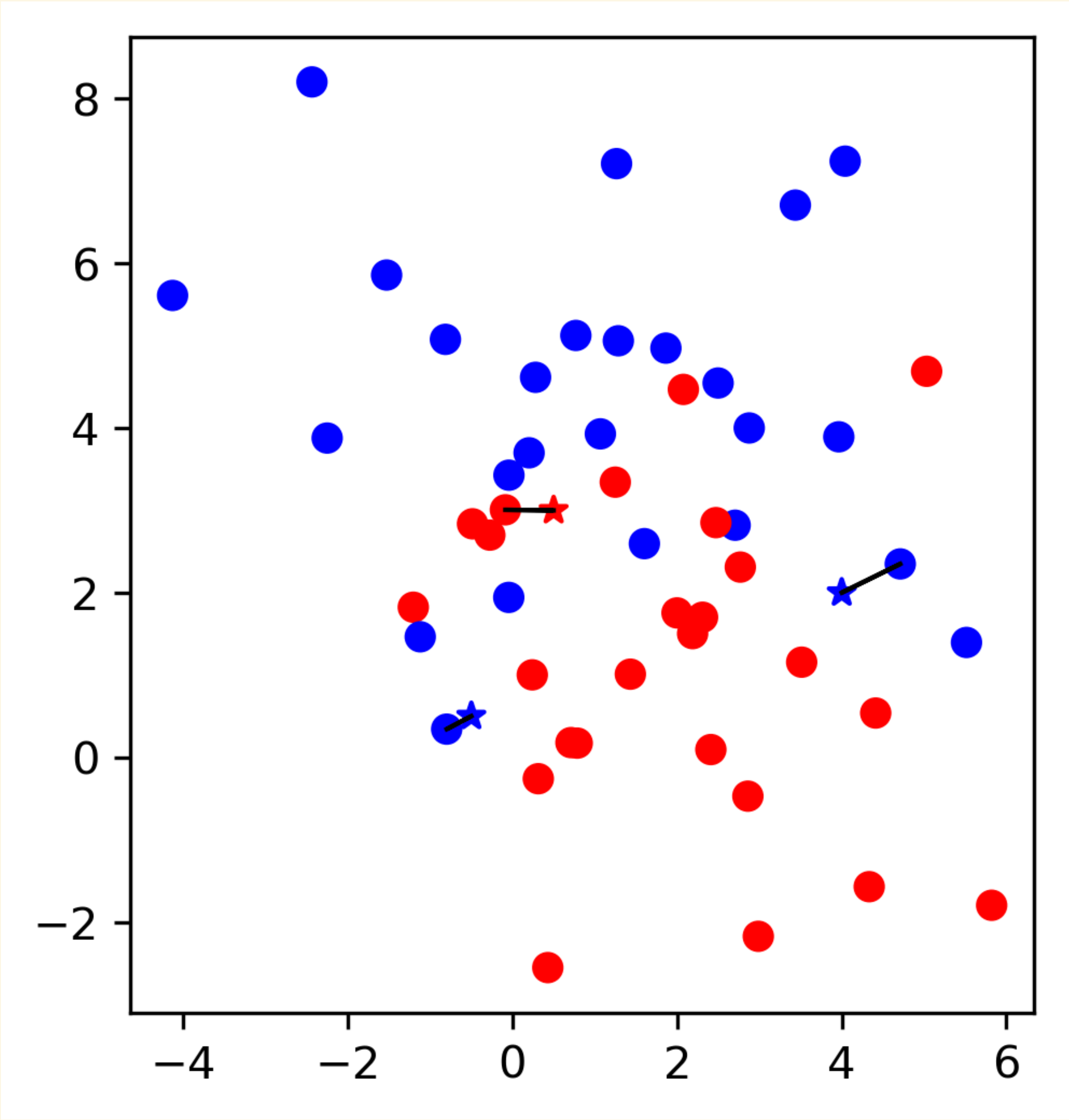
FROM 1-NN TO K-NN

Consider the *k* nearest neighbors

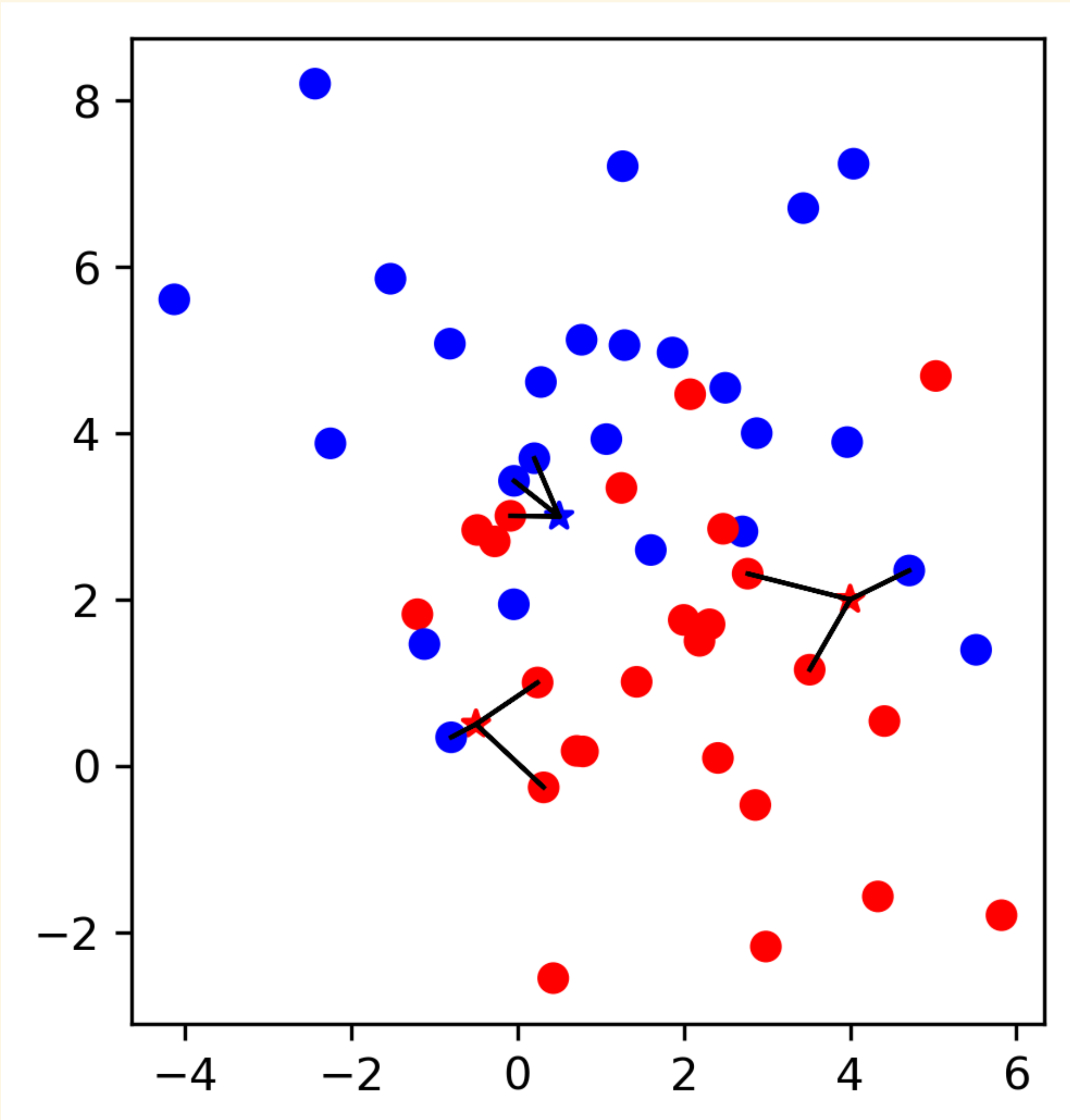
Assign the class that is the most common among them

Variation: consider each label relative to the effective distance of the neighbor.

1-NN



3-NN



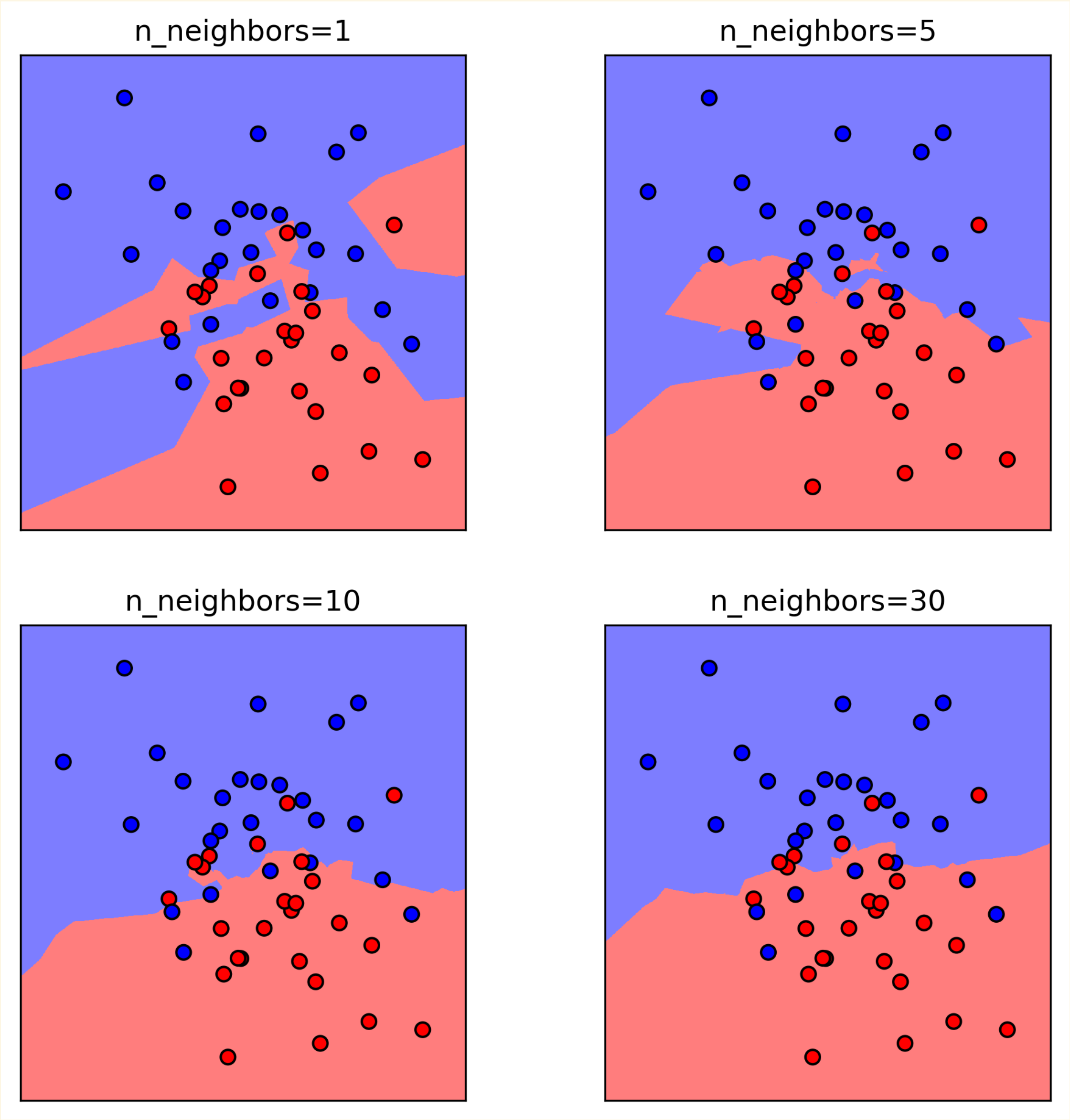
LEARNING

Given the labeled examples, k-NN determines the areas around each example which give a certain class.

k-NN learns an area or *surface* and applies it in classification

A larger k does not always mean a better classification

INFLUENCE OF K



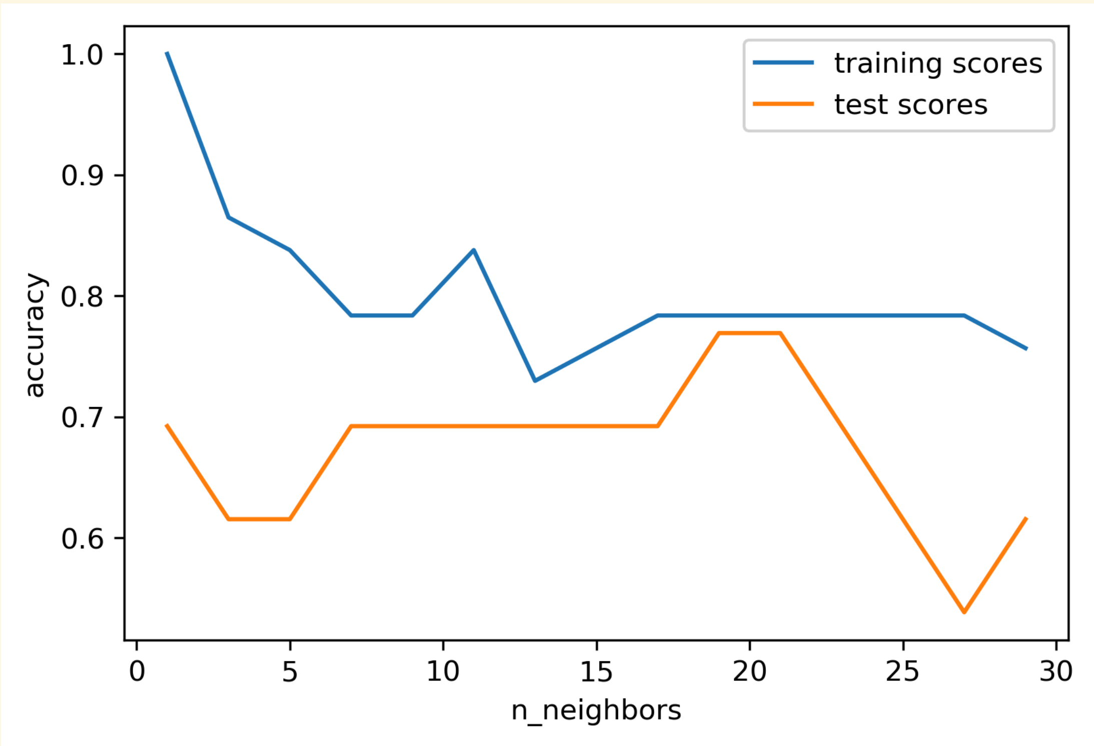
OBSERVATIONS

k-NN

- introduces us to *voting systems*
- is effective when the two classes are balanced, i.e., not *skewed*, in the dataset
- there is no standard way to choose k, yet it may greatly influence the outcome:
  - we face hyperparameter optimization.
- on large training data, even 1-NN approaches the *irreducible_error_rate* (2x).

TRADE-OFFS

Sometimes improving accuracy on the training data does not translate into improved accuracy in testing against *unseen* data



1-NN is perfect on training but 0.7 on test.

Increasing k does not improve much and *overfitting* creeps in.