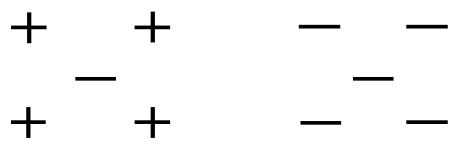
Practice Exercise 5 – kNN Classification (solution)

Question 1: Consider the following data set with two input attributes x and y (i.e. the coordinates of the points) and one binary output t (taking values + or -). We want to use k-nearest neighbours (K-NN) with Euclidean distance to predict t.



c) Describe how you would choose the number of neighbours K in K-NN in general.

A common approach to select k in k-nn classification is to split the data into training and validation datasets. The training of k-nn model is performed with various values of k, and the evaluation of these models is performed on the validation datasets. The model with highest accuracy is then chosen to select an appropriate value for k.

Question 2: Which of the following increases the likelihood of over-fitting and/or under-fitting? Why?

a) increasing the number of neighbours k in kNN?

With the increase in value of k, there are increased chances of under-fitting. The suitable value of k is model selection problem.

b) decreasing the number of neighbours k in kNN?

With the decrease in value of k, there are increased chances of over-fitting. The suitable value of k is model selection problem.

Question 5: How do you compare weighted kNN classifier with a regular non-weighted kNN classifier? Has one of them any advantages (or disadvantages) over the other?

Weighted k-nn can possibly use all training instances as neighbours rather than only k neighbours, thus making it possible to work as a global approximation function rather than a local approximation function as in typical k-nn. In such cases, the weighted k-nn will have a higher computational burden then regular k-nn. In general, weighted k-nn has benefit of giving more weight to nearest neighbours and less weights to farther neighbours. On the other hand, regular k-nn assigns equal weight to all of its neighbours regardless of how far they're from the sample.