Polynomial Classifier, Branch-and-Bound

Exercise 30 The training set of a classifier consists of four two-dimensional samples per class:

$$S_1 = \left\{ \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 2 \end{pmatrix}, \begin{pmatrix} 2 \\ 1 \end{pmatrix}, \begin{pmatrix} 3 \\ 1 \end{pmatrix} \right\} \quad (1)$$

$$S_2 = \left\{ \begin{pmatrix} 2 \\ 4 \end{pmatrix}, \begin{pmatrix} 2 \\ 3 \end{pmatrix}, \begin{pmatrix} 3 \\ 2 \end{pmatrix}, \begin{pmatrix} 4 \\ 2 \end{pmatrix} \right\} \quad (2)$$

The decision boundary shall be parameterized by a function that is quadratic in the components of the input features. It shall evaluate to a value of 0 for samples from S_1, and a value of 1 for samples from S_2.

$$\hat{\delta}(\mathbf{c}_k) = \sum_{i,j=1; i \leq j}^{2} a_{ij} c_{k,i} c_{k,j} = \begin{cases} 0, & \text{if } \mathbf{c}_k \in S_1 \\ 1, & \text{if } \mathbf{c}_k \in S_2 \end{cases} \quad (3)$$

Here, $\mathbf{c}_k$ denotes the k-th sample, and $c_{k,i}$ the i-th component of $\mathbf{c}_k$.

(a) Compute the coefficient vector $\mathbf{a}$ for the decision boundary.

(b) Classify the two features $\mathbf{c}_9$, $\mathbf{c}_{10}$ below. Note that if the result is not precisely 0 or 1, we decide for the class which is closest to the classifier result.

$$\mathbf{c}_9 = \begin{pmatrix} 2 \\ 6 \end{pmatrix}, \qquad \mathbf{c}_{10} = \begin{pmatrix} 6 \\ 2 \end{pmatrix} \quad (4)$$

Exercise 31 **Programming Task:** The Branch-and-Bound method can be used to choose a subset of features.

(a) Describe the idea behind the method and how it could be applied in a depth-first-search (DFS) and a breadth-first-search (BFS) of the tree of possible solutions. What property is required of the rating function that allows us to prune subtrees?

(b) Use an exhaustive search as well as a DFS and BFS Branch-and-Bound approach to select seven out of ten features, given the following rating function:

$$\begin{aligned} G_i^9 &= 10 + i, && \text{for } i = 1, 2, 3, \ldots, 10 \\ G_{i,j}^8 &= 10 - \tfrac{1}{2}(i + j), && \text{for } i, j = 1, 2, 3, \ldots, 10 \\ G_{i,j,k}^7 &= 10 - \tfrac{1}{2}(i + j + k), && \text{for } i, j, k = 1, 2, 3, \ldots, 10 \end{aligned}$$

(c) Compare all three methods in terms of the number of rating function evaluations. Why could this be an appropriate measure of efficiency in typical use cases?