Assignment 1 - Supervised Learning

Mirko Morello Andrea Borghesi

March 2024

1 Introduction

In this task, our objective is to train various classifiers and assess their performance with the t-times k-fold cross-validation method. The dataset employed for training and testing the classifier comprises solely two numerical features along with a corresponding label.

2 Classifiers used

We used a total of 4 classifiers: linear SVM, SVM with a Gaussian kernel, k-NN, and a Tree Predictor. The hyperparameters used were found through a heuristic. We obtained consistent results through the runs, therefore the reported ones will vary only slightly if we were to reproduce the experiment.

Linear SVM

An SVM (Support Vector Machine) is an algorithm designed to identify the optimal hyperplane that effectively segregates data into distinct classes by maximizing the margin between them. The hyperplane is characterized by support vectors, representing the data points closest to it.

In this instance, a linear SVM was employed with the `KernelScale` parameter set to 0.5125. Linear SVMs are particularly useful when the relationship between the input variables and the output variable is linear. As we'll see, it is less flexible than the Gaussian kernel and can only model linear decision boundaries.

Gaussian SVM

Also in this case an SVM was used but with a Gaussian kernel, it maps input data to a higher-dimensional space using a Gaussian function.
The Gaussian kernel is used when the relationship between the input variables and the output variable is non-linear, it is more flexible than the linear kernel and can model complex decision boundaries.
The Gaussian function is defined by a width parameter `KernelScale`, which determines the spread of the function, in this case, a value of 0.4875 was used.

K-NN

The K-Nearest Neighbour (KNN) classifier operates without the need for a distinct training phase; instead, it stores the training set and labels a new instance by examining its k-closest neighbors. In our experiments, we opted to configure the algorithm with `k = 10`.

Tree predictors

Tree predictors enable the classification of data by employing a series of rules, each aimed at minimizing the entropy of the resulting split sets. However, excessive splitting can lead to overfitting of the predictor to the data. Hence, we have chosen to limit the number of `splits` to a maximum of 12 using the split criterion `gdi`.

3 Results

3.1 Performance analysis

We conducted a 2-fold cross-validation procedure five times to enhance the reliability of the classifiers' accuracy assessment. During each iteration of the 2-fold cross-validation, the dataset was divided into two distinct subsets. One subset served as the test set, while the other one was used as the training set and vice-versa for the second iteration.

	SVM Linear	SVM Gaussian	K-NN	Decision Tree
Dataset 1	1.0000	1.0000	1.0000	0.9947
Dataset 2	0.8813	0.8760	0.8673	0.8220
Dataset 3	0.6667	0.9257	0.9217	0.8967
Dataset 4	0.5697	0.9753	0.9450	0.9613

Table 1: Accuracy computed by a 5-times 2-fold cross-validation for each algorithm for each dataset

3.2 Ranking

The ranking process commences with the previously discussed performance analysis. A rank is assigned to each algorithm for every dataset, with the top position awarded to the algorithm that exhibits the highest accuracy.

	SVM Linear	SVM Gaussian	K-NN	Decision Tree
Dataset 1	2	2	2	4
Dataset 2	1	2	3	4
Dataset 3	4	1	2	3
Dataset 4	4	1	3	2

Table 2: Ranks based on the accuracy of the 5-times 2-fold cross-validation

For each algorithm is then computed the average rank.

Dataset	Classifier	Average Rank
Dataset 1	SVM Linear	2.75
Dataset 2	SVM Gaussian	1.50
Dataset 3	K-NN	2.50
Dataset 4	Decision Tree	3.25

Table 3: Average rank of each algorithm across all datasets

We've chosen $\alpha = 0.05$, and by looking at the table of the critical values for the two-tailed Nemenyi test, having 4 classifiers, we get a $q_\alpha = 2.569$ and with that we calculate CD, which is:

$$\text{CD} = q_\alpha \sqrt{\frac{k(k+1)}{6N}} = 2.569 \sqrt{\frac{4(4+1)}{6 \cdot 4}} = 2.569 \cdot 0.912 = 2.345$$

4 Conclusions

Having now calculated the CD and the average ranking, we can plot the Critical Difference Diagram:

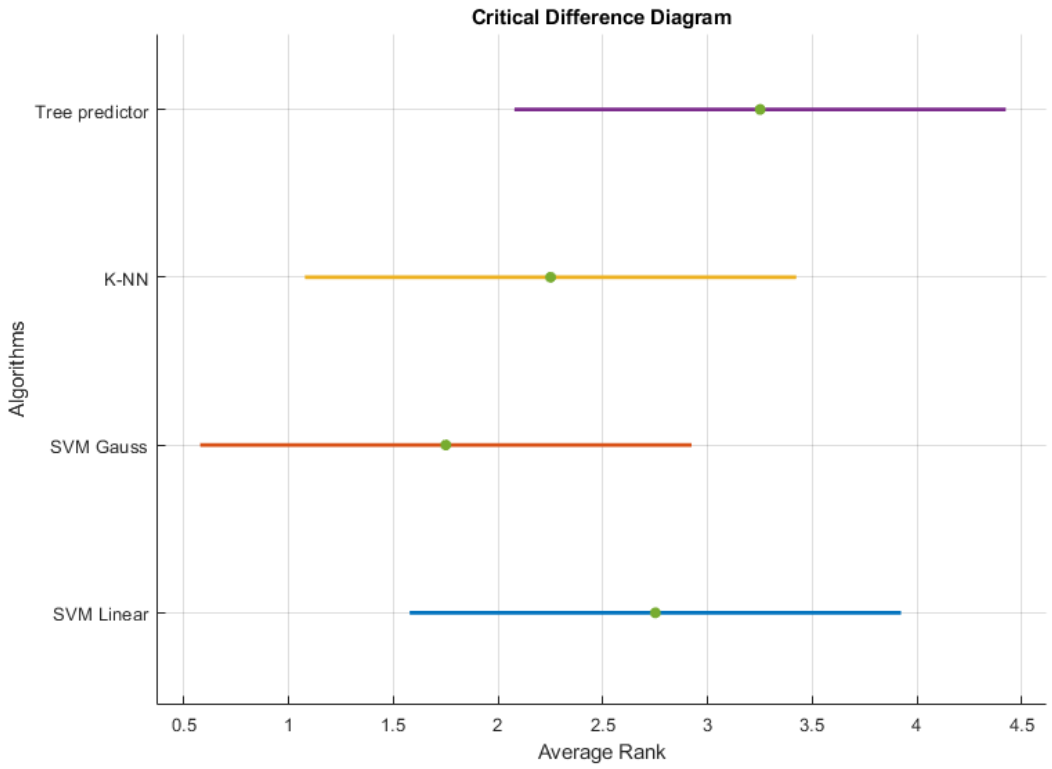

Figure 1: Critical difference diagram

As we can see, there is no significant difference between the classifiers, indeed, if we calculate the p value, we get 0.2276 which is larger than $\alpha = 0.05$ so we can't reject the null hypothesis, this furthermore confirms that the statistical difference between the classifiers is not significant.