Statistical Comparisons of Classifiers over Multiple Data Sets (Supervised Learning Course - Lab1)

Luigi Daddario - MAT. 908294

March 15, 2024

University of Milano-Bicocca - 2023/2024

Abstract

In this study, I have conducted a statistical comparison of the performance of four machine learning classifiers: Linear Support Vector Machine (Linear SVM), SVM with Radial Basis Function kernel (SVM RBF), K-Nearest Neighbors (KNN), and Decision Tree. The analysis was performed over four distinct datasets and utilized a 5x2 cross-validation methodology. The Friedman test was employed as a non-parametric alternative to ANOVA. This choice was necessitated by the Friedman test's suitability for analyzing data where the normality assumption does not hold.

The investigation begins with the application of the Friedman test to the combined results, which returned a p-value of 0.7273. This value, being substantially greater than the commonly accepted significance level (α) of 0.05, indicates no significant discrepancies in classifier performance across the datasets. Pairwise differences were subsequently scrutinized using the Nemenyi post-hoc test, which concurred with the Friedman test, as evidenced by the absence of statistically significant differences between any pairs of classifiers.

Further examination through visual analysis represented in supplementary diagrams reveals no meaningful distinction between the mean ranks of the classifiers. Complementing these findings, the critical difference (CD) for multiple comparisons was calculated at an α of 0.05, resulting in a value of 2.3452. When this CD is considered alongside the consistent ranking of classifiers, it underscores the lack of a statistically significant advantage for any particular classifier.

The implications of these results suggest that, for the datasets evaluated, classifier selection should extend beyond accuracy metrics to consider other crucial factors such as model interpretability, computational demand, and suitability for the task at hand. These findings provide valuable insights for the application of machine learning classifiers in practice.

Keywords: Classifier Comparison · Friedman Test · Nemenyi Post-Hoc Test · Critical Difference (CD)

Contents

List of Tables

List of Figures

Chapter 1

Classifiers Used with Chosen Hyperparameters

Algorithms Overview

First of all I want to briefly describe how the algorithms involved actually works.

Linear SVM (Support Vector Machine)

How it works: Linear SVM is a supervised learning algorithm that finds the hyperplane that best separates the data into classes. In a two-dimensional space, this hyperplane is a line. The algorithm aims to maximize the margin between the data points of the two classes, which is the distance between the closest points to the line (support vectors).

Parameters:

- Kernel Function: Specifies the type of hyperplane used to separate the data. In Linear SVM, the kernel is linear.
- Kernel Scale: Used to scale the data before applying the kernel function. For a linear kernel, this parameter typically doesn't alter the hypothesis space.

SVM with RBF Kernel

How it works: Similar to Linear SVM, but uses a Radial Basis Function (RBF) kernel to enable non-linear separation. Instead of a straight line, the RBF kernel allows the SVM to fit a more complex, curved boundary between classes.

Parameters:

- Kernel Function: The RBF kernel uses distance calculations to determine the separation boundary.
- Kernel Scale: Controls the complexity of the boundary. A smaller scale results in a more complex, tightly-fitting boundary, which might risk overfitting.

K-Nearest Neighbors (KNN)

How it works: KNN is a non-parametric, instance-based learning algorithm. It classifies a new sample based on a majority vote of its 'k' nearest neighbors. If 'k' is 1, then the sample is simply assigned to the class of its nearest neighbor.

Parameters:

- Distance Metric: Determines how the distance between points is calculated. The Euclidean distance is the straight-line distance between two points in a multidimensional space.
- Number of Neighbors ('k'): The 'k' in KNN is a crucial parameter that affects the classification. A smaller 'k' makes the boundary more complex, while a larger 'k' makes it smoother.

Decision Tree

How it works: A Decision Tree is a flowchart-like tree structure where each internal node represents a test on an attribute, each branch represents the outcome of the test, and each leaf node represents a class label. The paths from the root to the leaf represent classification rules.

Parameters:

- Split Criterion: The criterion used to choose the feature on which to split at each node. Gini's Diversity Index (GDI) is a measure of impurity or variability and aims to maximize the purity of the split.
- Maximum Number of Splits: Specifies the maximum number of splits in the tree. More splits can capture more information about the data but can lead to a more complex model, which might risk overfitting.

The classifiers and Hyperparameters I've used in the Study are the following:

Algorithm	**Parameter**	**Value**
Linear SVM	Kernel Function	Linear
	Kernel Scale	1
SVM with RBF Kernel	Kernel Function	Gaussian
	Kernel Scale	0.1
K-Nearest Neighbors (KNN)	Distance Metric	Euclidean
	Number of Neighbors	10
Decision Tree	Split Criterion	Gini's Diversity Index (GDI)
	Maximum Number of Splits	10

Table 1.1: Overview of Algorithms and Their Hyperparameters

Chapter 2

Accuracy Obtained by the Classifiers on the 4 Datasets

2.1 Accuracy Obtained by the Classifiers on the 4 Datasets

The accuracy obtained by each classifier on the four datasets, as shown in the Table 2.1, is:

Dataset	Linear SVM	SVM RBF	KNN	Tree
Dataset 1	1.0000	0.9973	1.0000	0.9947
Dataset 2	0.8827	0.8060	0.8693	0.8207
Dataset 3	0.6667	0.9187	0.9220	0.9013
Dataset 4	0.5817	0.9517	0.9483	0.9557

Table 2.1: Accuracy of classifiers across datasets.

2.2 Average Rankings

The average rankings across all datasets for each classifier are:

- **Linear SVM**: 2.5000
- **SVM RBF**: 2.7500
- **KNN**: 2.0000
- **Decision Tree**: 2.7500

2.3 Meanranks computed in MATLAB

The average rankings across all datasets for each classifier, computed by the friedman function in MATLAB are:

- **Linear SVM**: 2.6250
- **SVM RBF**: 2.7500
- **KNN**: 1.8750
- **Decision Tree**: 2.7500

2.4 Observations

In the analysis of the average rankings, it is noted that the algorithm I developed simplifies the conventional ranking method used by MATLAB's `friedman` function. In my approach, the classifier with the lowest average ranking is deemed the top performer, in contrast to the `friedman` function where a higher average rank signifies better performance. This simplicity in my methodology leads to a slight difference in rankings among classifiers that achieve identical accuracy over a certain dataset. This deviation arises because, according to the reference paper, the `friedman` function handles classifiers with the same accuracy in a unique manner; in the case of ties between two classifiers on a specific dataset, average ranks are assigned. However, the order of classifiers remains consistent, suggesting a stable performance hierarchy across both ranking methods.

Despite these variations, the Friedman test indicates no statistically significant differences in performance among the classifiers. Therefore, the selection of a classifier might be influenced by aspects beyond performance, such as model interpretability or computational efficiency. To align the presentation of results in the Critical Difference (CD) diagram, I've used 1 - accuracy1 as a parameter for the `friedman` function in MATLAB.

How the Friedman Test Works

The Friedman test is a non-parametric statistical test used to detect differences in treatments across multiple test attempts It ranks each row (or block) together and then compares the ranks of the columns. If the column averages differ significantly, the null hypothesis (that there is no difference in treatments) can be rejected.

The formula for the Friedman statistic (χ_F^2) is:

$$\chi_F^2 = \frac{12N}{k(k+1)} \left(\sum_{j=1}^{k} R_j^2 - \frac{k(k+1)^2}{4} \right)$$

where N is the number of blocks (datasets in our case), k is the number of treatments (classifiers), and R_j is the sum of the ranks for the j-th treatment across all blocks.

The ranks R_j are calculated for each treatment within a block, with the lowest rank going to the best-performing treatment. Under the null hypothesis, all classifiers, for example, are equivalent, and therefore their ranks R_j should be approximately equal. If the Friedman statistic exceeds a critical value from the chi-squared distribution with $k-1$ degrees of freedom, the null hypothesis is rejected.

In my observation, the Friedman test computed by MATLAB indicates no significant difference in the performance of the classifiers. This is based on a calculated p-value that is higher than the conventional alpha level (typically 0.05), suggesting that any observed differences in classifier performance are not statistically significant. It's important to note

that the Friedman test assumes that the measurement scale of data is at least ordinal, there are no ties, and the blocks are randomly selected.

Critical Difference (CD)

To compute the Critical Difference (CD) value for the Nemenyi test, we use the following formula:

$$CD = q_\alpha \sqrt{\frac{k(k+1)}{6N}}$$

Here, k is the number of classifiers, N is the number of datasets, and q_α is the critical value based on the Studentized range statistic corresponding to the chosen α significance level. The critical value q_α can be obtained from a statistical table for the Studentized range distribution.

Table 2.2: Critical values for the two-tailed Nemenyi test

#classifiers	2	3	4	5	6	7	8	9	10
$q_{0.05}$	1.960	2.343	2.569	2.728	2.850	2.949	3.031	3.102	3.164
$q_{0.10}$	1.645	2.052	2.291	2.459	2.589	2.693	2.780	2.855	2.920

To apply this to our case:

1. We have $k = 4$ classifiers.
2. There are $N = 4$ datasets.
3. The significance level α is typically set at 0.05 for such tests.

Assuming the critical value q_α for $\alpha = 0.05$ and degrees of freedom corresponding to k and N is available, we should then substitute these into the formula to get the CD. Given $q_{0.05} = 2.569$, which is a common critical value for the Nemenyi test when $k = 4$ and $N = 4$. Using this formula, we would calculate the CD value and compare it to the differences in rank observed in the data to determine if the differences are statistically significant.

Critical Difference (CD) at alpha = 0.05 is: 2.3452

Looking at the Fig. 2.1 and at the table below, we can observe that the differences in performance between the classifiers are not statistically significant when considering the Critical Distance (CD) value. However, this conclusion can also be drawn from the associated p-values of the conducted pairwise comparisons.

Comparison	Mean Rank Difference	Result
SVM Linear vs SVM RBF	0.1250	Not higher than the CD of 2.3452
SVM Linear vs KNN	0.7500	Not higher than the CD of 2.3452
SVM Linear vs Tree	0.1250	Not higher than the CD of 2.3452
SVM RBF vs KNN	0.8750	Not higher than the CD of 2.3452
SVM RBF vs Tree	0.0000	Not higher than the CD of 2.3452
KNN vs Tree	0.8750	Not higher than the CD of 2.3452

In theory, comparisons between classifiers are performed by first determining if there are overall differences among the classifiers using a test like the Friedman test. If this test indicates significant differences, pairwise comparisons are conducted to determine which specific classifiers differ from each other. These pairwise comparisons are adjusted for multiple testing to control for the increased likelihood of a Type I error due to the multiple hypotheses being tested.

When the average ranks of two classifiers differ by more than the CD value, it is considered evidence that the classifiers perform differently with statistical significance. The CD value takes into account the number of comparisons being made and adjusts the significance level accordingly to avoid Type I errors (false positives) that are likely when making multiple comparisons. On the other hand the p-value for each comparison indicates whether the observed difference between those two specific classifiers could have occurred by random chance. A small p-value (typically less than 0.05) suggests that the difference in performance is unlikely to have occurred by chance, thus rejecting the null hypothesis.

2.5 Visualization of the Results

Comparison	p-value
SVM Linear vs SVM RBF	0.9990
SVM Linear vs KNN	0.8393
SVM Linear vs Tree	0.9990
SVM RBF vs KNN	0.7661
SVM RBF vs Tree	1.0000
KNN vs Tree	0.7661

Table 2.3: Pairwise comparisons of classifier performance with associated p-values.

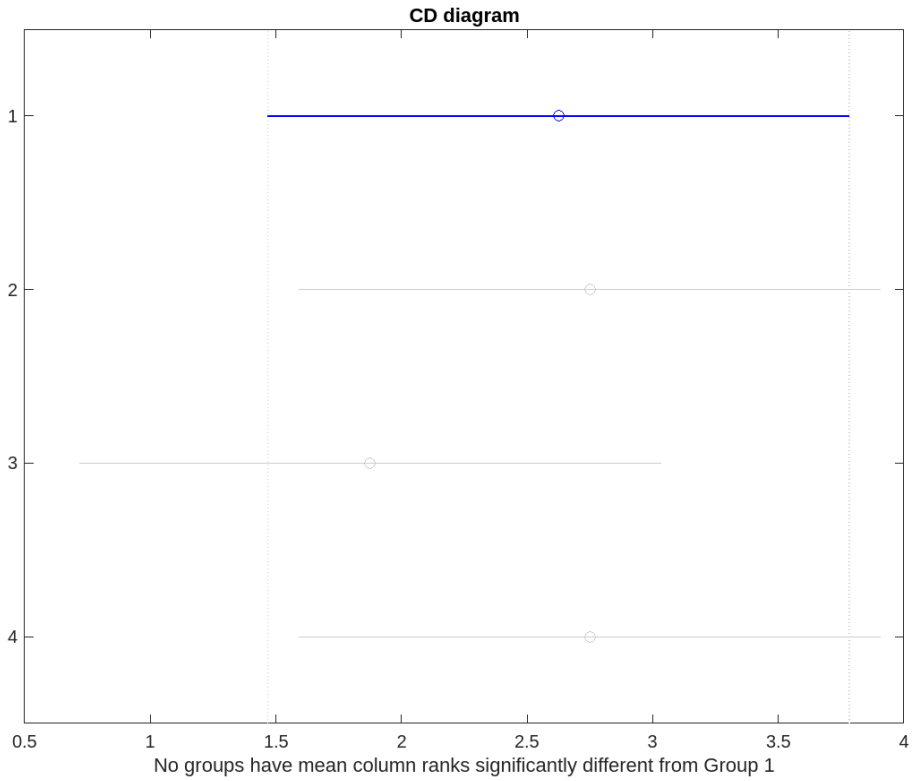

Fig. 2.1: Multiple comparison of ranks.

Chapter 3

Conclusion

It is worth noting that since the p-value from the Friedman test was higher than the typical alpha level of 0.05, indicating no significant differences among the classifiers, the subsequent Nemenyi post-hoc test was redundant. The initial Friedman test already suggested that further post-hoc analysis might not yield significant results.

Nonetheless, conducting a post-hoc test such as Nemenyi provides a more granular view and a double-check mechanism. However, in this case, the post-hoc analysis confirms the Friedman test's finding that the classifiers perform similarly across the datasets, and therefore, their choice can be based on other practical considerations, such as interpretability, computational demands, or compatibility with the problem domain.

3.1 References

The code used was developed starting from the skeleton provided in the laboratory, during the Supervised Learning course. In addition to the knowledge acquired during the course and the paper provided, other online resources were used, cited in the bibliography. [1] [2] [3]

Bibliography

[1] MATLAB. *Friedman's test*. https://it.mathworks.com/help/stats/friedman.html.

[2] MATLAB. *multcompare function*. https://it.mathworks.com/help/stats/multcompare.html.

[3] Wikipedia. *SVM*. https://en.wikipedia.org/wiki/Support_vector_machine.