Machine Learning I

Unit 6: *The ScikitLearn Library*

Master in Artificial Intelligence 2023/2024

Lucía María Álvarez Crespo
lucia.maria.alvarez.crespo@udc.es

Alejandro Dopico Castro
alejandro.dopico2@udc.es

November 2, 2023

Contents

1 Introduction

This report is dedicated to presenting the outcomes derived from an extensive training process involving various iterations of machine learning models, conducted specifically for Unit 6 of the Machine Learning I course. In this pursuit, we meticulously examined diverse combinations of pivotal hyperparameters within each model, though it is crucial to underscore that an in-depth analysis is warranted for a comprehensive understanding of the merits and demerits associated with each model.

The primary evaluation metric employed is accuracy, with the added nuance of a 10-fold cross-validation technique (k = 10) being employed. Consequently, this report furnishes both the mean and standard deviation of accuracy scores for each model, contributing to a robust assessment of their performance.

Furthermore, this report incorporates a confusion matrix, highlighting the model that embodies the most effective hyperparameter configuration. Notably, this specific model was trained and evaluated on the complete dataset. In the event of multiple hyperparameter configurations yielding the highest mean accuracy, our selection criteria prioritizes the combination with the lowest standard deviation. In the rare case of a tie, we would opt for the simplest configuration, denoted by the highest position within the table's rows. It is imperative to clarify that the confusion matrix's rows correspond to the actual class labels, while the columns represent the model's predictions.

2 Experimentation

In this section, we will delve into the core phase of our study – the experimentation. Here, we will meticulously outline the process of training and evaluating a range of machine learning models on the well-known Iris dataset[1]. Throughout this section, we will explore the various hyperparameter combinations we have tested and scrutinize their individual performances in detail. Additionally, we will present a confusion matrix that highlights the model that has proven to be optimal based on the hyperparameter configuration used.

2.1 Artificial Neural Networks

When working with Artificial Neural Networks (ANNs), one of the key things to check is how many hidden layers are in the network. In Table 1, we're displaying the results for different network setups, where each setup has a different number of neurons in its hidden layers. Importantly, we're not including information about the input and output layers because those depend on the specific problem we're trying to solve, and they stay the same for all the networks. All these networks were trained for a maximum of 1,000 training epochs, with a learning rate of 0.01. If we didn't see any improvement in accuracy on our validation data (which is 10% of our training data) for 50 rounds, we stopped the training early. The confusion matrix for the best ANN configuration, with 8 neurons in one hidden layer, is shown in Table 2.

Topology	Mean Accuracy	Standard Deviation
4	0.94	0.06
8	**0.96**	**0.07**
2, 4	0.90	0.05
12, 12	0.94	0.7

Table 1: Artificial Neural Networks Results

	Iris-Setosa	Iris-Versicolor	Iris-Virginica
Iris-Setosa	10	0	0
Iris-Versicolor	0	7	0
Iris-Virginica	0	0	13

Table 2: Confusion Matrix for ANN

2.2 Support Vector Machines

In our exploration of Support Vector Machine (SVM) models, we have conducted experiments involving two distinct kernel functions. These kernels, serving as mathematical functions, represent distinct computational approaches. One of these kernels is the radial basis function, and the other is a polynomial kernel. Notably, each kernel possesses unique hyperparameters and configurations, including the 'C' parameter, as visualized in Table 3. The confusion matrix for the most optimal SVM configuration, employing a radial basis function (RBF) kernel with C = 1, is visually represented in Table 4.

[1] https://archive.ics.uci.edu/ml/datasets/iris

Kernel	C	Mean Accuracy	Standard Deviation
rbf	0.6	0.96	0.05
rbf	**1**	**0.96**	**0.05**
poly	0.6	0.95	0.04
poly	1	0.95	0.06

Table 3: Support Vector Machines Results

	Iris-Setosa	Iris-Versicolor	Iris-Virginica
Iris-Setosa	13	0	0
Iris-Versicolor	0	10	0
Iris-Virginica	0	0	7

Table 4: Confusion Matrix for SVM

2.3 k-Nearest Neighbours

In the case of the k-Nearest Neighbour (kNN) models, our experimentation exclusively involves varying the parameter that determines the number of neighbours considered for classification, denoted as 'k.' This parameter exploration is visually displayed in Table 5. The confusion matrix for the best kNN configuration, with k = 13, is shown in Table 6.

Neighbours	Mean Accuracy	Standard Deviation
7	0.96	0.06
11	0.95	0.05
13	**0.97**	**0.06**
17	0.96	0.06

Table 5: k-Nearest Neighbours Results

	Iris-Setosa	Iris-Versicolor	Iris-Virginica
Iris-Setosa	12	0	1
Iris-Versicolor	0	10	0
Iris-Virginica	0	0	7

Table 6: Confusion Matrix for k-NN

2.4 Decision Trees

We conducted experiments on the Decision Trees models by examining the hyperparameter associated with the maximum depth of the tree. The outcomes of these experiments are graphically presented in Table 7. Furthermore, we present the confusion matrix of the most optimal Decision Tree configuration, characterized by a maximum depth of 13, within Table 8.

MaxDepth	Mean Accuracy	Standard Deviation
5	0.94	0.07
7	0.94	0.07
13	**0.95**	**0.05**
15	0.93	0.09

Table 7: Decision Trees Results

	Iris-Setosa	Iris-Versicolor	Iris-Virginica
Iris-Setosa	12	0	1
Iris-Versicolor	0	10	0
Iris-Virginica	0	0	7

Table 8: Confusion Matrix for Decision Trees

3 Conclusions

The machine learning models we explored exhibited strong performance on the Iris dataset, with ANNs, SVMs, kNN, and decision trees showcasing notable accuracy. These findings underscore the importance of hyperparameter tuning in achieving optimal model performance. While all models performed exceptionally well in solving the problem, it's noteworthy that the highest mean accuracy was achieved the kNN model, showcasing their robustness and effectiveness in tackling the Iris dataset classification challenge.