

CIFAR-10 picture recognition analysis

Lab report

Auer Thomas

September 15, 2022

Contents

1	Introduction	2
2	Methods and Resources	2
3	Results	3
3.1	Keras neural network	3
3.1.1	Hyperparameter search (Hyperband)	3
3.2	SKLearn multi-layer perceptron	3
3.2.1	Hyperparameter search (Gridsearch)	3
4	Discussion	3
5	Conclusion	3

1. Introduction

This report discusses neural network experimentation on the [CIFAR-10 dataset](#). The implementation is handled in the programming language [Python](#), using the libraries [Keras](#) and [SKLearn](#). The goal is to analyse various models and identify the impact of different parametrization of the underlying algorithms.

2. Methods and Resources

The datasets are imported from a remote destination with the use of the [keras.datasets](#) package. Beside additional utility libraries, such as [matplotlib](#) and [seaborn](#) for displaying graphics, the [Tensorflow](#) supplies the necessary algorithms for creating and executing neural network computations and hyperparameter evaluation. A secondary approach using [SKLearn multi-layer perceptron](#) serves as additional reference in terms of computational performance and use. Notably, SKlearn implements [GridSearch](#) for parameter search.

Approach

After the download of the CIFAR-10 data, it is required to perform some data transmutations on the set. The data values are normalized from the value range of (0..255) to (0..1), by simple division. For the Keras computations, the training and test labels are categorized from a class vector of integers to a binary class matrix. SKLearn requires a distinct approach by reshaping the available data of the form (Images, X-Pixel, Y-Pixel, RGB-Channel) to (Images, {X-Pixel, Y-Pixel, RGB-Channel}). Optionally, the size of the used data is reduced in order to reduce the time necessary to run analytics. Afterwards, the main computation on the data commences.

3. Results

3.1 Keras neural network

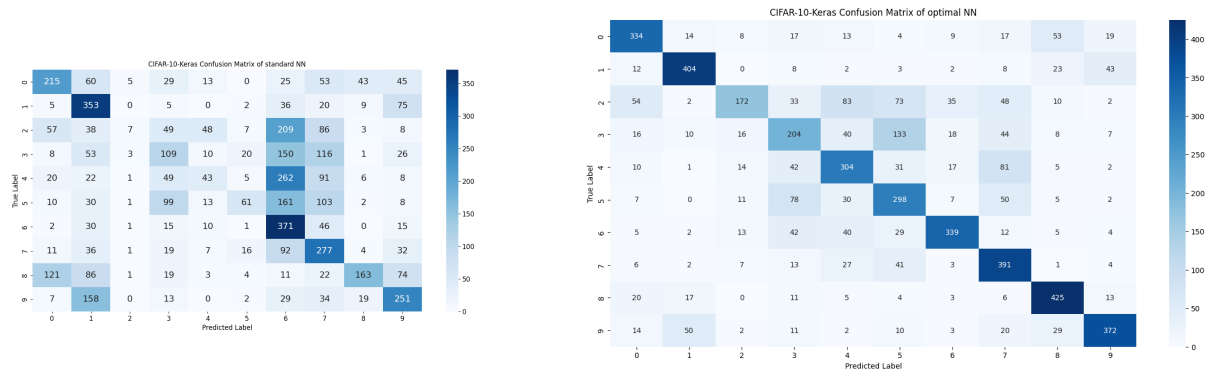


Figure 3.1: Confusion matrix comparison inbetween default network (left) and optimized network (right) parametrization.

3.1.1 Hyperparameter search (Hyperband)

3.2 SKLearn multi-layer perceptron

3.2.1 Hyperparameter search (Gridsearch)

4. Discussion

5. Conclusion