# EMBEDDED MACHINE LEARNING

## Exercise 6

**Group 6**
Mielke Max, Maximilian Burr, Sergej Bespalov
June 12, 2024

# Contents

# 1 Pruning

**We used the ResNet18 from the previos exercise with the CIFAR10 Dataset**

Neural networks, while powerful, can become cumbersome due to their large size. Pruning offers a technique to address this issue. It involves strategically removing unimportant connections or even entire neurons from a trained network. This "slimming down" process aims to reduce the model's size and computational cost without sacrificing its accuracy. Pruning techniques can target individual weights or neurons, with weight pruning being more common due to its lower impact on performance. By effectively pruning a network, we can achieve faster inference times, lower memory requirements, and potentially even deploy models on resource-constrained devices.

## 1.1 Unstructured Pruning

Unstructured pruning tackles neural network size by snipping away individual, less important connections between neurons. It's a simpler approach for slimming down models, but can create challenges for hardware due to the scattered "sparse" network it leaves behind. As illustrated in 1, unstructured pruning has no big influence on the accuracy of the used Resnet18 with the CIFAR-10 Dataset. However, the sparsity introduced can make the network less efficient on traditional hardware, which often isn't optimized for sparse computations.

## 1.2 Structed Pruning

Structured pruning cuts the fat from neural networks by removing entire filters or neurons, aiming to streamline the architecture and reduce model size. This targeted approach is valuable for deploying AI on resource-constrained devices. However, as shown in 1, structured pruning can lead to a trade-off between model size and test accuracy for specific architectures like ResNet-18. The figure illustrates that while pruning reduces size, it also impacts test accuracy in ResNet-18. This highlights the importance of tailoring the pruning strategy to the specific network architecture to achieve the optimal balance between efficiency and performance.

## 1.3 Discussion: Structured and Unstructured

**Structured pruning (SP) and unstructured pruning (UP)** are both techniques for reducing the size and computational cost of neural networks. However, they take different approaches with distinct advantages and disadvantages.

Structured pruning removes entire filters, channels, or neurons, resulting in a more compact and streamlined network architecture. This often leads to better hardware compatibility and potentially maintains good test accuracy. However, implementing SP can be more complex and might not be equally effective for all network designs.

Unstructured pruning, on the other hand, is simpler to implement but removes individual connections within the network. This offers fine-grained control over model size reduction. However, the resulting "sparse" network can be challenging for hardware and may lead to a more significant drop in accuracy compared to structured pruning.

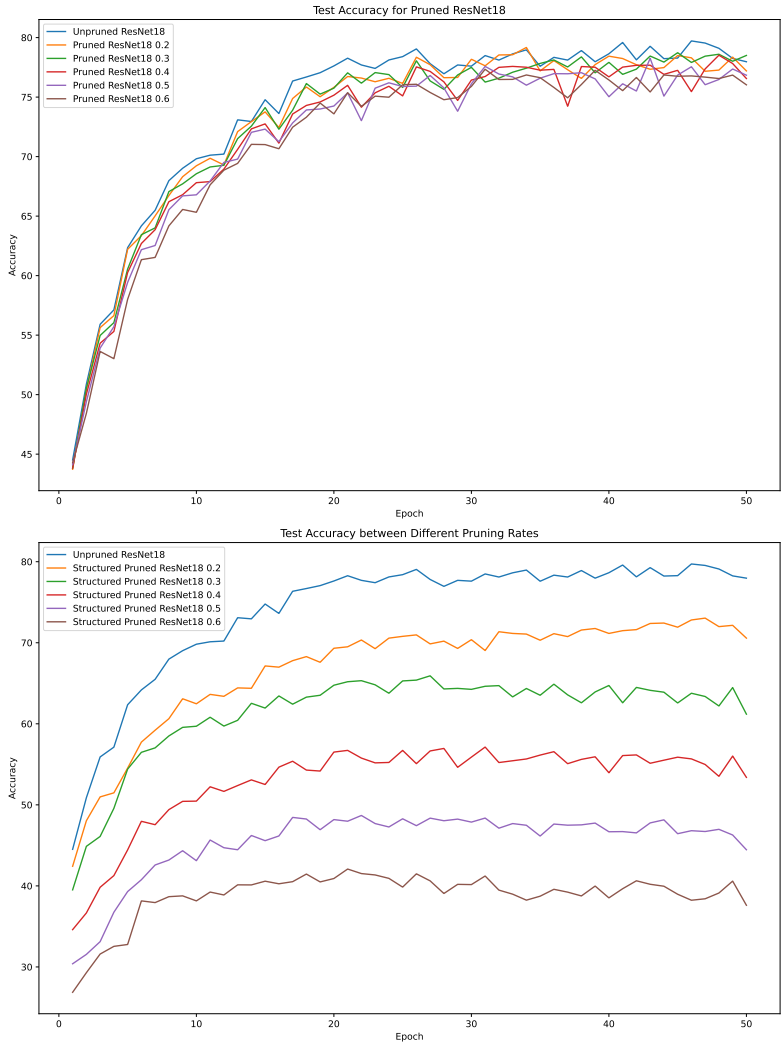


Figure 1: Test Accuracy Influence with Pruning on CIFAR10

The choice between SP and UP depends on several factors. If maintaining a well-defined architecture and hardware compatibility are crucial, SP might be preferable. When simplicity and fine-grained control are priorities, UP could be a good option. Ultimately, the best technique depends on the specific needs of the network and the desired balance between efficiency and performance.

## 2 Willingness to present

Exercise Quantization Yes