Yurii Oleksandr Bibik

Dr. Kang

CS 422

23 April 2023

Homework 5

For the homework assignment, I implemented a Convolutional Neural Network (CNN) using the PyTorch framework to classify the MNIST dataset. The architecture of the network was designed to efficiently capture hierarchical patterns in the images while minimizing the risk of overfitting. The layers and their justifications are as follows:

1. A 2D convolutional layer with 32 filters, each with a kernel size of 3x3 and padding of 1. This layer was chosen to capture local patterns and edges in the images.
2. A ReLU activation function to introduce non-linearity and enable the network to learn complex features.
3. A max pooling layer with a 2x2 filter to reduce the spatial dimensions and improve the model's computational efficiency.
4. A second 2D convolutional layer with 64 filters, each with a kernel size of 3x3 and padding of 1, to detect higher-level features in the images.
5. A ReLU activation function for the same reasons mentioned earlier.
6. A max pooling layer with a 2x2 filter to further reduce spatial dimensions.
7. A dropout layer with a dropout rate of 0.25 to mitigate overfitting by randomly dropping out nodes during training.
8. A fully connected layer with 128 nodes to serve as a classifier based on the learned features.
9. A ReLU activation function to introduce non-linearity in the fully connected layer.
10. A dropout layer with a dropout rate of 0.25 to further prevent overfitting.
11. A fully connected layer with 10 nodes, which represents the 10 possible digits in the MNIST dataset, for the final classification step.

I used the cross-entropy loss as the cost function because it is suitable for multi-class classification problems. The Adam optimizer was chosen for its adaptive learning rate, which helps improve convergence speed. A learning rate of 0.001 was used as a balanced choice to allow the optimizer to converge without oscillating or overshooting the minimum. To evaluate the model's performance, I employed 5-fold cross-validation to ensure a more robust estimate of the model's accuracy. The code also plots the learning curves for each fold and displays the accuracies of each fold and their average as text.

Graphical user interface, diagram, application

Description automatically generated