Homework 1 – Intro to Deep Learning

Daniel Jolin[1]
Student ID: 801282735
Homework: 1
github.com/RocketDan11/ml/real_time/homework/homework.ipynb
Data: 1/32/25

abstract The following document explores image and housing datasets to examine the limitations of multi-layer perceptron (MLP) models. The neural networks' architectures will be systematically varied in both width and depth to analyze the relationship between model capacity and performance, with particular attention to overfitting behavior. Various encoding schemes are implemented to properly parameterize categorical variables that lack temporal relationships.

written in LaTeXusing overleaf for compilation.

1 Problem 1

1.1 Network Architecture

To train a fully connected network for the CIFAR-10 dataset the input channel needs to support 3 channel (32x32 pixel) images, which total 3072 parameters. 3 hidden layers start with 512 neurons, followed by a non-linear activation function (in this case ReLU), followed by another 256 neuron/activation pair, concluded with a final 10 neuron output layer to represent our image classes. This will be used as our baseline model for future comparison.

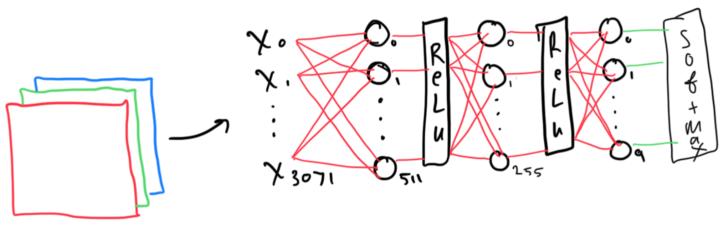

Figure 1: Network Architecture

1.2 Training Process

By using the nn.CrossEntropyLoss() function from pyTorch as our loss function, which already expects raw logits (unnormalized scores) as inputs and internally applies a softmax during its computation, we render an additional softmax operation redundant.

[1]djolin1@charlotte.edu

Training proceeded for 20 epochs with a batch size of 100, and the following results were achieved:

Metric	**Value**
Epoch	20
Training Accuracy	91.42%
Validation Accuracy	53.02%
Test Accuracy	52.04%
Precision	0.528
Recall	0.520
F1 Score	0.523

Table 1: Model Performance Metrics at Epoch 20

1.3 Results Analysis

The training results can be visualized through several metrics. First, the confusion matrix shows the distribution of predictions across classes:

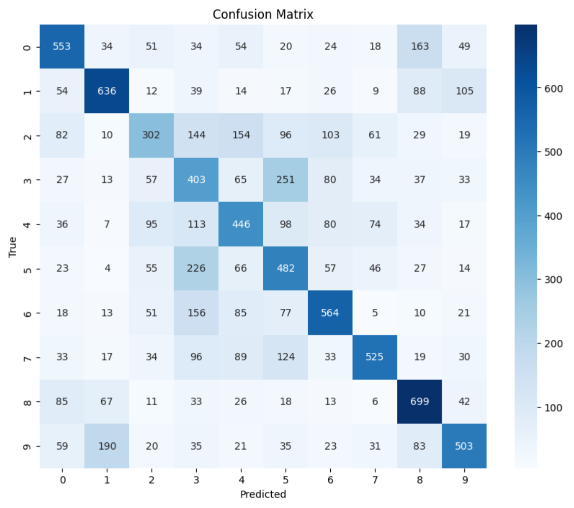

Figure 2: Confusion Matrix

The training process can be observed through the loss and accuracy curves:

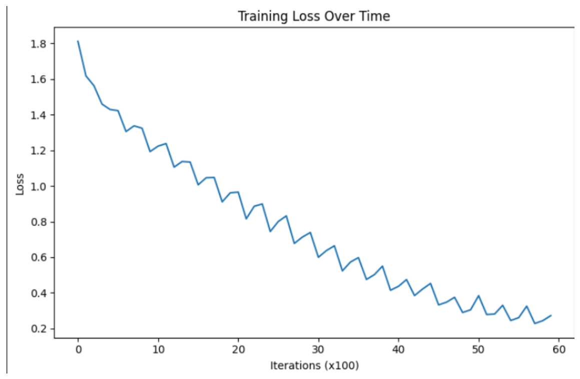

Figure 3: Training Loss Over Epochs

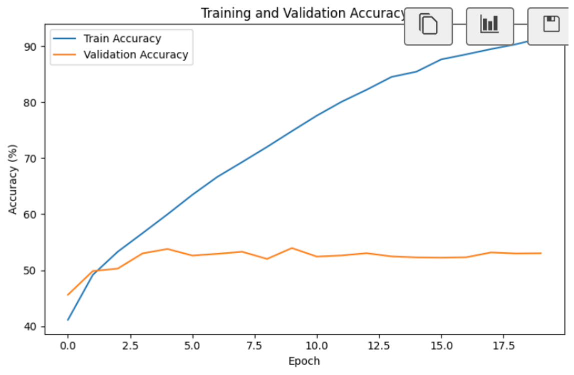

Figure 4: Training and Validation Accuracy

20 epochs at a batch size of 100 provided sufficient iterations to train the model. However, the significant discrepancy between training accuracy (91.42%) and validation accuracy (53.02%) suggests that the model is overfitting to the training set.

1.4 Increasing Complexity

By increasing the depth (modifying the architecture to include additional hidden layers) we observe a slight increase in the validation accuracy at the expense of computational resources and time to compute. The improvement does not justify the cost as the over-fitting issue persists.

2 Problem 2

2.1 Network Architecture

Training a fully connected network to regress housing price using the housing dataset requires parameterizing the input dimension to accommodate the number of classes we choose to train on. Since the model regresses a housing price prediction there should be a single output.

2.2 Base Case

Using the above architecture the following results are achieved through a naive encoding scheme. The model accuracy achieved is .52 MAE using 3457 parameters. Modest results for a tiny model.

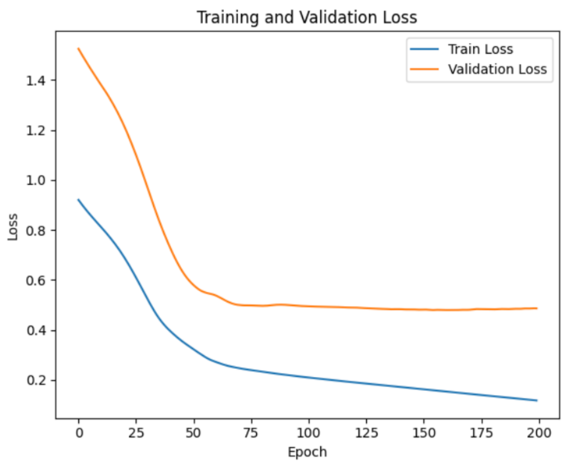

Figure 5: Base Model Training vs. Validation Loss

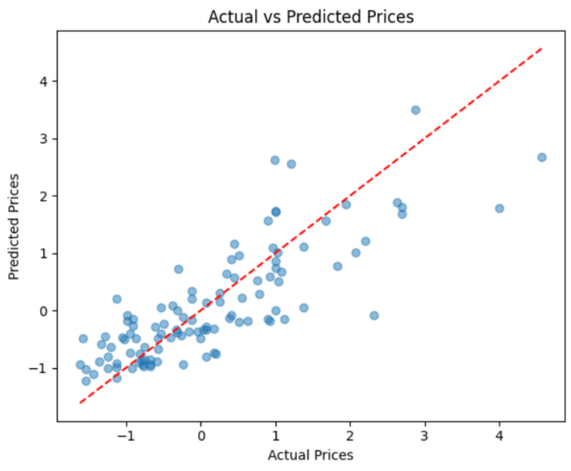

Figure 6: Price Predictions

2.3 One-Hot Encoding

By implementing a one-hot encoding scheme (OHES), the number of input parameters is increased (now 3521 parameters), and subsequently, the model performs better, achieving an MAE of .55

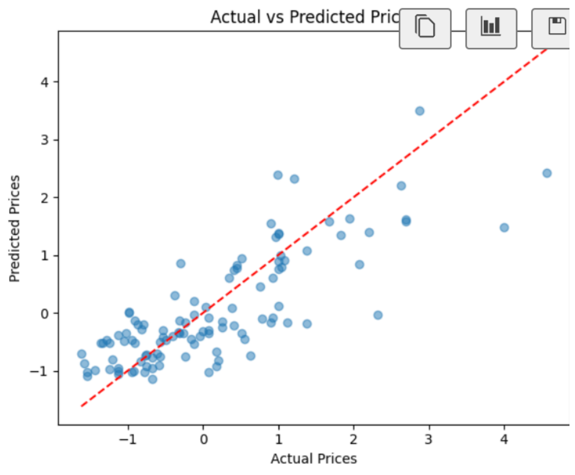

Figure 8: Price Predictions after OHES

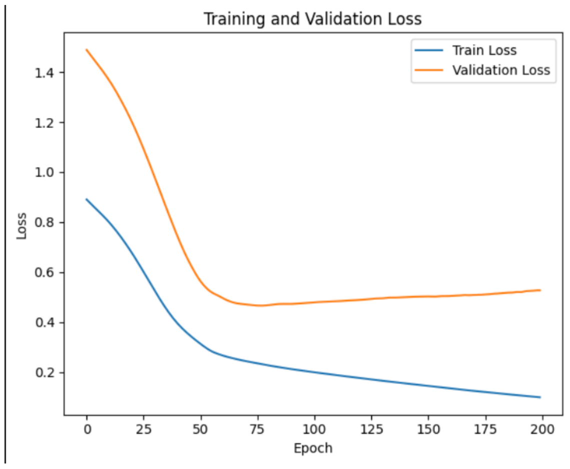

Figure 7: Training and Validation Loss After OHES

2.4 Increasing Complexity

Adding two more hidden layers significantly increases the model complexity to 16,073 parameters. However, the MAE score slightly degrades to 0.54, suggesting that merely increasing model size without architectural improvements does not yield better results. This indicates that the model's limitations may lie in its architecture rather than its capacity.