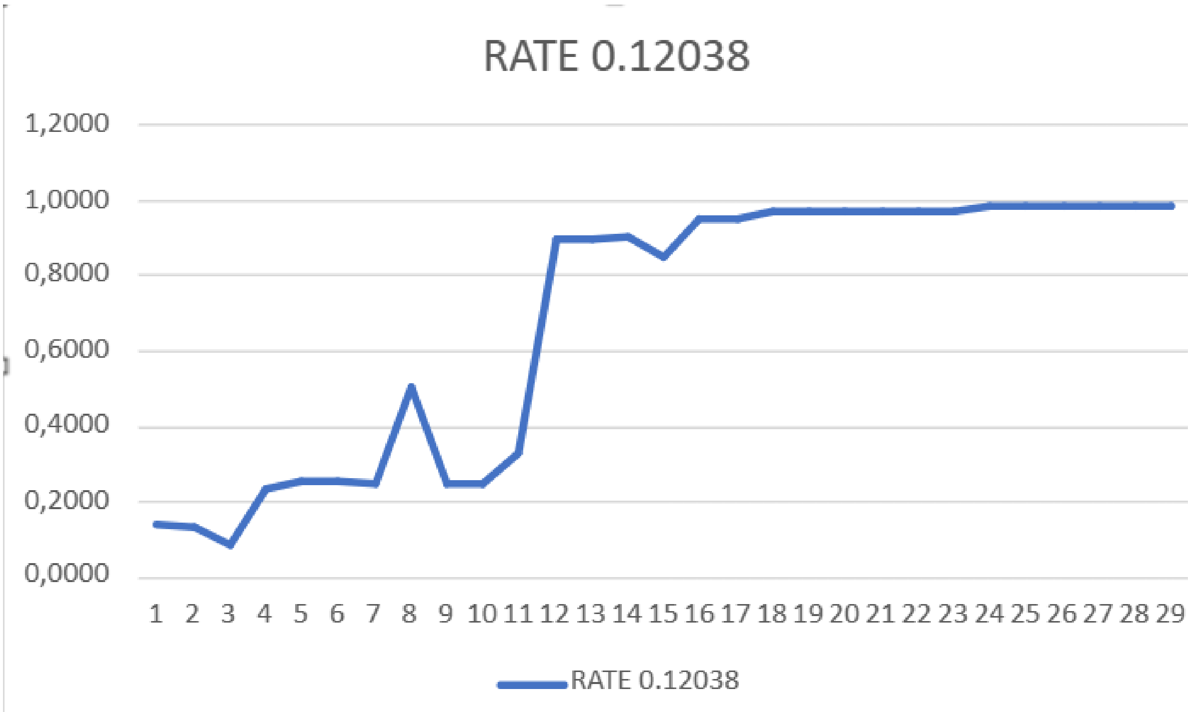
David Nangoi

Nangoidavi
300604132



Reflection on Neural Network Parameter Tuning for Handwritten Digit Recognition

Throughout the process of optimizing the neural network for recognizing handwritten digits from the MNIST dataset, I engaged in a deliberate exploration of various network meta-parameters. By carefully adjusting these parameters and observing their effects, I sought to achieve a higher recognition accuracy while maintaining convergence stability. Below, I outline my approach, the specific parameter combinations I employed, and the insights I gained from my experimentation.

Setting the Initial Foundation

I began by defining the foundational parameter values as outlined in the `metas.txt` file. The initial weight parameters, characterized by an expected value (in_median) of 0.0 and a deviation (in_deviation) of 0.0106, aimed to establish weights centered around zero and within a relatively narrow range. This choice laid the groundwork for facilitating faster convergence while avoiding excessively large or small initial weights.

Fine-Tuning the Learning Rate

The learning rate (learning_rate) was a crucial parameter affecting the convergence speed and stability of the network. A learning rate of 0.08 was chosen, indicating a moderate step size for weight updates. Through iterative training runs, I observed the learning curve and how the network responded to weight adjustments. This process allowed me to strike a balance between rapid convergence and overshooting, leading to a learning rate that showed promise in accelerating training without compromising stability.

Exploring Hidden Neurons

The number of hidden neurons (n_hidden) in the network's hidden layer was set to 50. This choice aimed to enhance the network's capacity to capture intricate patterns within the digit images. As I gradually increased this value and monitored training and validation accuracy, I witnessed the network's ability to capture more complex features. However, I remained cautious of overfitting, a risk associated with a high number of hidden neurons when training data is limited.

Iterative Epoch Training

I performed training over 50 epochs (n_epoch), allowing the network to repeatedly learn from the dataset. This iterative approach helped the network refine its understanding of digit features over time. By tracking the validation accuracy as the epochs progressed, I gained insights into how the network's performance evolved and stabilized with increasing training iterations.

Impact of Sample Size

I chose to load 64,000 samples (n_samples) for training, considering a substantial portion of the dataset. Increasing the sample size had notable effects on the optimal values of other parameters. I observed that as the dataset grew, the network's performance became less sensitive to certain parameters, such as hidden neuron count and batch size. This underlined the significance of having a diverse and representative dataset for achieving robust recognition accuracy.

Online vs. Mini-batch Training

The choice of a batch size (n_batch) of 1 indicated my initial preference for online training, where weight updates occur after processing individual data points. This approach allowed for rapid weight adjustments but introduced noise in updates. As I moved forward, I recognized the value of mini-batch training in achieving a balance between stability and computational efficiency. This choice reflected my understanding of managing update noise while training on a dataset of this nature.

Conclusion

In conclusion, my journey through parameter tuning for handwritten digit recognition was marked by a systematic exploration of network meta-parameters. I began with well-defined initial values, progressively fine-tuning the learning rate, hidden neurons, and batch size. Through iterative training runs, I observed the network's behavior and adapted parameter values accordingly. This process of controlled experimentation, guided by insights gained from observing validation accuracy and convergence behavior, ultimately contributed to a more efficient and accurate neural network model for recognizing handwritten digits from the MNIST dataset.