Parnal Sinha
L04: CNN MNIST Reflection
ITAI 2376: Deep Learning

Reflective Journal L04: MNIST CNN Exploration & Insights

1. **Learning Insights**
    a. *What New Concepts about CNNs Did I Uncover During This Lab?*

Engaging with this lab illuminated the intricate dance of preprocessing, architectural design, and hyperparameter tuning in shaping a CNN's efficacy. A revelation was the transformative power of data augmentation—techniques like random rotations or flips trained the model to embrace variability, enhancing its adaptability to unseen MNIST digits. I also grasped how scaling the number of convolutional filters (e.g., from 32 to 128) enriched feature extraction, sharpening the model's ability to distinguish nuanced patterns like the loops in an "8" versus the straight lines in a "1."

Another eye-opener was the role of optimizer selection in training dynamics. Initially, I experimented with the default Stochastic Gradient Descent (SGD), which plodded toward convergence. Switching to the Adam optimizer, with its adaptive learning rate adjustments, catalyzed faster and steadier accuracy gains, illuminating why it's a staple in deep learning workflows. This would be quintessential for different models and data structures and as such analysis and examination of the dataset prior to any work is crucial to ensure what hyperparameters would be useful for this

    b. *How Does This Lab Connect to My Prior Knowledge of Neural Networks?*

My earlier exposure to neural networks centered on Multi-Layer Perceptrons (MLPs), where fully connected layers processed data without spatial awareness. This lab bridged that gap, revealing how CNNs leverage spatial hierarchies to detect edges, shapes, and patterns progressively—unlike MLPs, which treat all inputs as flat numbers. I now see why CNNs outshine MLPs for images: they preserve and exploit spatial relationships, a concept I only theorized before. Additionally, my theoretical grasp of gradient descent deepened through observing Adam's practical superiority over SGD, reinforcing its adaptive, momentum-driven approach in navigating complex loss landscapes.

    c. *What Surprised Me About Working with CNNs and the MNIST Dataset?*

I was struck by how delicately small changes in architecture ripple through performance. Increasing convolutional filters incrementally—from 32 to 64, then 128—yielded measurable accuracy boosts, underscoring the potency of deeper feature extraction. Equally surprising was data augmentation's impact: even on a seemingly straightforward dataset like MNIST, adding subtle distortions fortified generalization, thwarting overfitting I hadn't anticipated. Finally, I was amazed at how swiftly a CNN achieved high accuracy (~98%) with minimal epochs compared to the sluggish convergence I'd expect from traditional MLPs, highlighting CNNs' efficiency for visual tasks.

Parnal Sinha
L04: CNN MNIST Reflection
ITAI 2376: Deep Learning

2. **Challenges and Growth**
    a. **What Specific Challenges Did I Encounter While Implementing the CNN?**

A primary hurdle was striking a balance between model complexity and overfitting. Initially, adding more layers and filters boosted accuracy, but beyond a point, the model memorized training data, faltering on test examples. I also grappled with training instability—early runs showed erratic loss fluctuations due to an oversight in the optimizer step sequence, which I later rectified. Tuning hyperparameters like batch size, dropout rate, and learning rate demanded trial and error, as finding the sweet spot felt like navigating a maze.

    b. **How Did I Overcome These Challenges?**

To curb overfitting, I introduced a milder dropout rate of 0.3, preserving more features while preventing over-dependence, and used batch normalization to stabilize activation distributions, accelerating convergence. I resolved training instability by restructuring the training loop, ensuring proper loss computation, backpropagation, and optimization sequencing, which smoothed out loss curves. For hyperparameters, I ran small, controlled experiments—testing batch sizes (32, 64, 128) and optimizers (SGD vs. Adam)—settling on Adam with a 0.001 learning rate for optimal balance.

    c. **What Resources or Strategies Helped Me Understand Difficult Concepts?**

Visualizing loss and accuracy curves over epochs was a game-changer, revealing patterns like overfitting or learning rate issues at a glance. I leaned on TensorFlow's documentation for nuanced insights into optimizers, dropout, and convolutional operations, which clarified their mechanics. Most crucially, hands-on experimentation - iteratively tweaking parameters and observing outcomes - deepened my intuition, transforming abstract ideas into tangible understanding. It was difficult att first but also experimenting and writing down the exact mathematical formulation also illuminated the specific nuances and methodology utilized by the parameters, providing a much more

3. **Personal Development**
    a. *How Has This Lab Transformed My Understanding of Deep Learning?*

This lab shifted my perspective, showing deep learning as a synergy of data preparation, model design, and fine-tuning—not just theoretical constructs like backpropagation. I now appreciate how preprocessing (e.g., normalization, augmentation) and architectural choices (e.g., CNN layers) underpin performance, beyond mere training algorithms. It solidified my realization that CNNs excel in image tasks by harnessing spatial structures, a principle I'll carry into future computer vision endeavors.

    b. *What Aspects of CNNs Would I Like to Explore Further?*

I'm eager to delve into cutting-edge architectures like ResNet or EfficientNet, which address challenges like vanishing gradients or efficiency bottlenecks. I'm also intrigued by transfer learning—using pretrained models on vast datasets and adapting them for niche tasks. Exploring activation functions beyond ReLU, such as LeakyReLU or GELU, to understand their influence on learning dynamics is another avenue I'd pursue. I think experimenting with the coalesences of these with original models can provide very interesting comparative analyses and help with ascertaining the appropriate benchmarks for the models, further exemplifying nuanced methods to approach document accuracy and precision of said models.

c. If I Were Already Familiar with CNNs, What New Perspectives Did I Gain?

Though I had some familiarity, this lab reshaped my approach through structured experimentation. Previously, I leaned on theoretical best practices, but now I value the iterative, empirical process of testing and refining—seeing how minor tweaks like data augmentation or hyperparameter adjustments yield major gains. It also highlighted the nuanced interplay of preprocessing and regularization, reinforcing that top performance hinges on thoughtful data handling as much as model depth. This has also made me recognize that the foundations of physics and math ought to be student as boundary conditions and indexing errors may become extremely prevalent as the model tries to learn from complex datasets – understanding the input size, kernel size and the number of strides the convolutional filter will take is quintessential to determining whether padding is necessary or not.

Conclusion & Future Direction

This lab illuminated my understanding by providing theoretical and experimental workflows through model construction of Convolutional Neural Networks. The insightful dive into working with CNNs in the real world enabled me to learn how to balance building the model, preparing the data, and tweaking the settings just right. Through hands-on trial and error (figuring out concepts such as overfitting, fixing training hiccups, and deciding on the best design), I developed a much sharper feel for the practical side of deep learning. These lessons, from experimenting with the workflow alongside the code gave me the confidence to take on tougher image datasets and try out fancier CNN ideas down the road. Moving forward, this experience has sparked my curiosity to explore new ideas in neural networks, mixing what I've learned in theory with real-world testing.