**ADITYA AMIN**

**ASSIGN : CV-03**

1. After each stride-2 conv, why do we double the number of filters?

It's important to note that doubling the number of filters is not a universal rule and may vary depending on the specific network architecture and problem domain. The choice of the number of filters depends on factors such as dataset complexity, network depth, computational resources, and empirical observations during model development and training. Experimentation and model evaluation can help determine the optimal number of filters for a given CNN architecture.

1. Why do we use a larger kernel with MNIST (with simple cnn) in the first conv?

the choice of kernel size is a hyperparameter that can vary depending on the specific architecture and problem at hand. Experimentation and evaluation can help determine the optimal kernel size for a given task. While a larger kernel can capture broader patterns, it may also increase the number of parameters and computation required, so there needs to be a balance based on the complexity of the dataset and available resources.

1. What data is saved by ActivationStats for each layer?

Activation Histogram: The histogram of activation values in the layer. This histogram provides insights into the distribution and range of activation values, helping to analyze the activation patterns and identify any potential issues such as saturation or vanishing gradients.

Activation Mean: The mean value of activations in the layer. The mean can indicate the overall activation level and potential biases in the network.

Activation Variance: The variance of activations in the layer. The variance represents the spread or dispersion of activation values and can provide information about the diversity and dynamic range of activations.

Activation Sparsity: The sparsity of activations in the layer, which refers to the percentage of zero-valued activations.

Activation Distribution Metrics: Additional metrics such as median, standard deviation, maximum, and minimum activation values may also be recorded.

4. How do we get a learner's callback after they've completed training?

Define a Callback: Implement a callback class that inherits from the appropriate base class provided by your deep learning framework.

Override the Relevant Methods: In your callback class, override the relevant methods that correspond to the desired point after the completion of training.

Define Custom Actions: Inside the overridden method, define the actions you want to perform or the information you want to access.

Attach Callback to the Learner: Attach the callback to the learner or training process by registering it using the appropriate mechanism provided by your deep learning framework.

Run the Training Process: Start the training process as usual, running your training loop or using the provided training functions.

5. What are the drawbacks of activations above zero?

Saturation and Vanishing Gradients: If activations become too large, they may saturate, causing the gradients during backpropagation to become very small. This can lead to the problem of vanishing gradients, where the gradients diminish as they propagate backward through the network.

Overfitting: When activations are allowed to go significantly above zero, it increases the risk of overfitting. Overfitting occurs when the model learns to fit the training data too closely, capturing noise or irrelevant patterns.

Computational Challenges: Large activations require more computational resources, both during the forward pass and backward pass. They involve higher memory usage and increased numerical precision requirements.

Non-linearity Constraints: Many activation functions used in neural networks, such as ReLU (Rectified Linear Unit), are non-linear and produce activations above zero.

Gradient Clipping: During training, gradient clipping is often applied to prevent exploding gradients. However, activations above zero can make gradient clipping less effective since the gradients may already be saturated or high.

6.Draw up the benefits and drawbacks of practicing in larger batches?

Benefits of Larger Batches:

Improved Training Efficiency: Larger batches can lead to faster training convergence since more training samples are processed in each iteration. This can result in shorter training times and faster model development.

Better Utilization of Hardware: Larger batches can effectively utilize the parallel processing capabilities of modern hardware, such as GPUs. By processing more samples in parallel, larger batches can maximize the utilization of hardware resources, leading to faster training and inference times.

Drawbacks of Larger Batches:

Increased Memory Requirements: Larger batches require more memory to store the activations and gradients during the forward and backward passes. This can be a limitation, especially when dealing with large models or when memory resources are limited.

Slower Updates: Larger batches can slow down the training process as each iteration takes longer to compute due to the increased computational requirements. This can potentially slow down the overall training progress.

7. Why should we avoid starting training with a high learning rate?

it is generally recommended to start with a moderate learning rate and adjust it based on the observed training dynamics. Techniques such as learning rate warm-up or using adaptive learning rate algorithms can also help in avoiding the pitfalls of starting with a high learning rate and improve the training process.

8. What are the pros of studying with a high rate of learning?

Accelerated Learning: A high learning rate can facilitate faster acquisition and understanding of new concepts and skills.

Enhanced Cognitive Flexibility: Studying with a high learning rate can encourage cognitive flexibility, adaptability, and the ability to grasp new ideas and perspectives rapidly.

Improved Memory Consolidation: Intensive learning experiences with a high learning rate can enhance memory consolidation.

Intense Focus and Motivation: A high learning rate often requires a higher level of focus and concentration.

Rapid Skill Development: Learning at a high rate can expedite the development of specific skills or expertise. By engaging in intensive and concentrated learning, individuals can make substantial progress in a shorter time span, allowing them to acquire proficiency in a particular domain more rapidly.

9. Why do we want to end the training with a low learning rate?

using a low learning rate at the end of training allows for fine-tuning, smoother optimization, prevention of overfitting, exploration of narrow minima, avoidance of catastrophic forgetting, and increased robustness. It helps the model converge to a more optimal solution and improves its generalization performance on unseen data.