Effect of Weight Initialization Scale on Convergence

NAME: MOHANASUNDARAM MANOHRAN

STUDENT ID: 24054377

GITHUB LINK: https://github.com/Mohan-Mj2312/Machine-Learning.git

Abstract

Weight initialization is crucial in influencing the efficacy of a neural network's learning, especially in the first training phase. This tutorial examines the impact of different initial weight scales on convergence behavior, gradient dynamics, activation saturation, and overall classification performance. We conduct a controlled experiment on a two-dimensional synthetic dataset to investigate four initialization scales (σ = 0.01, 0.1, 0.5, 1.0) and examine their effects on training loss, accuracy, gradient norms, and ReLU activation patterns. Findings indicate that moderate variance (σ = 0.1) facilitates the most rapid and stable convergence, whereas excessively small or large scales result in slower learning or unstable gradients. These findings underscore the need of judiciously selected initialization procedures for effective and dependable neural network improvement.

Introduction

Successfully training a neural network necessitates more than only choosing an architecture and an optimizer; the initial magnitude of the weights can profoundly affect the speed and efficacy of the model's convergence. Initially, weights dictate the intensity of activations, the consistency of gradients, and the comprehensive transmission of information within the network. Inadequate initialization can result in exploding or vanishing gradients, sluggish convergence, or models becoming ensnared in suboptimal areas of parameter space. In contrast, appropriately scaled initialization facilitates seamless optimization and dependable feature acquisition.

This course examines the impact of initial weight variance on convergence behavior in feed-forward neural networks. We methodically alter the initialization scale (σ = 0.01, 0.1, 0.5, 1.0) and examine its impact on training loss, accuracy, gradient norms, and activation saturation. We demonstrate how varying scales in a two-class nonlinear dataset lead to quantifiable alterations in optimization speed and model efficacy. Through the visualization of loss curves, gradient dynamics, and ReLU saturation patterns, we seek to impart to students a clear and practical comprehension of the significance of initialization. This analysis underscores the significance of principled initialization approaches and establishes a basis for investigating more sophisticated methods, like Xavier and Kaiming initialization.

Fundamental Theory

Optimization and Initialization of Neural Networks

Neural networks acquire knowledge by progressively modifying weights to minimize a loss function, generally through gradient-based optimization methods like stochastic gradient descent (SGD) or Adam. Initially, the model lacks any learnt structure; its behavior is solely determined by the initial weight distribution. These initial parameters affect both the early training dynamics and the stability of gradient propagation across the network. If weights are excessively tiny, neuron activations tend to converge towards zero, resulting in disappearing gradients and significantly sluggish learning. Excessively huge weights can cause activations and gradients to erupt, resulting in unstable or divergent loss optimization.

The objective of appropriate initialization is to establish weights at a magnitude that ensures stable forward activations and backward gradient propagation. Classical initialization techniques encompass Xavier/Glorot initialization, which maintains variance across layers for tanh-like activations, and Kaiming/He initialization, tailored specifically for ReLU networks. Before implementing such schemes, it is essential to comprehend how simple Gaussian initialization with differing standard deviations affects training.

Activation Dynamics and ReLU Saturation

ReLU activations exhibit linearity for positive inputs and yield a value of zero for all negative inputs. While ReLU circumvents the gradient saturation observed in sigmoid and tanh functions, excessively tiny initial weights result in the majority of activations approaching zero, effectively deactivating neurons. This may restrict representational capacity, diminish gradient flow, and impede convergence. In contrast, excessively large weights generate broad activation distributions, which may drive several values into extreme ranges and unpredictably exacerbate gradients. Consequently, the initialization scale dictates whether ReLU neurons function within an informative and gradient-abundant regime.

Previous Research and Theoretical Perspectives

Research conducted by LeCun, Glorot, and He revealed that neural networks achieve effective training when the variation of activations and gradients is maintained throughout layers. Their research formalized variance-preserving initialization procedures and shown that poor initialization substantially impedes optimization. Recent empirical research indicate that even little differences in scale might affect convergence speed and generalization. This course explores how varying startup parameters leads to diverse convergence behaviors through experimental demonstration.

Mathematical Foundations

Weight initialization influences training by regulating the propagation of variance throughout a neural network. Examine a feed-forward network utilizing ReLU activations.

For a layer with input $x \in \mathbb{R}^d$, weights $W \in \mathbb{R}^{h \times d}$, and bias b, the pre-activation is

$$z = Wx + b.$$

If the weights are sampled from a zero-mean Gaussian distribution,

$$W_{ij} \sim \mathcal{N}(0, \sigma^2),$$

then the variance of the pre-activations is

$$\mathrm{Var}(z_i) = d\sigma^2 \, \mathrm{Var}(x).$$

Consequently, an increased σ enhances activation variance, whereas a minimal σ diminishes it. Subsequent to the application of a ReLU activation,

$$a = \mathrm{ReLU}(z),$$

the variance becomes

$$\mathrm{Var}(a) = \frac{1}{2} \mathrm{Var}(z),$$

because half of the Gaussian mass lies below zero and is truncated. If σ is too small, most $z_i \approx 0$, causing **ReLU saturation**, where several activations become uniformly null. If σ is very big, the elevated variance of z might lead to explosive activations and destabilize gradients.

During backpropagation, gradients propagate as

$$\delta^{(l)} = (W^{(l+1)})^\top \delta^{(l+1)} \odot \mathbb{1}_{z^{(l)} > 0},$$

demonstrating reliance on both the weight magnitude and the activation mask. The gradient variance consequently scales in direct proportion to

$$\mathrm{Var}(\delta^{(l)}) \propto h\sigma^2 \, \mathrm{Var}(\delta^{(l+1)}).$$

This elucidates why inadequate initialization results in:

vanishing gradients when $\sigma^2$is very small, and exploding gradients when $\sigma^2$is excessively big.

These theoretical relationships directly correlate to the empirical behavior found in gradient-norm and activation-saturation graphs.

Dataset Overview

To investigate the impact of weight initialization scale on convergence, we utilize the two-class Moons dataset, a commonly employed synthetic benchmark for nonlinear classification. This dataset offers a visually interpretable framework while preserving adequate complexity to expose optimization variances across initialization scales. The dataset has 2,000 samples organized in two interleaving semicircles, with Gaussian noise introduced to enhance work complexity. Each point is characterized by two continuous properties (x_1, x_2). The target variable denotes the class label (0 or 1).

Prior to training, all characteristics are standardized using z-score normalization to prevent the learning process from being influenced by the scale of the input. The dataset is divided into 70% for training and 30% for testing, facilitating a dependable assessment of generalization performance under the four initialization circumstances (σ = 0.01, 0.1, 0.5, 1.0). The low dimensionality enables intuitive interpretation of activation patterns, gradient behavior, and convergence dynamics.

Execution and Trials

This study systematically assesses the impact of initial weight scaling on neural network convergence. All tests employ a three-layer fully connected neural network with ReLU activations, trained on the binary Moons dataset. To isolate the impact of initialization scale, all other factors—architecture, optimizer, learning rate, and dataset—were maintained constant across situations.

Experimental Configuration

All model variants employ the identical architecture:
Input layer: two units
Concealed layers: 64 units per layer
Output layer: 2 units (softmax classifier)

Four initialization scales were evaluated by collecting from

$$W \sim \mathcal{N}(0, \sigma^2),$$

with σ ∈ {0.01, 0.1, 0.5, 1.0}.

Biases were initialized to zero. The optimizer utilized was Adam, with a learning rate of 1×10^{-3}, and the models were trained for 200 epochs. This arrangement facilitates the clear observation of early optimization behavior without imposing excessive computational requirements. During each epoc, we documented:

Cross-entropy loss

Training accuracy

Gradient L2 norm

Fraction of near-zero ReLU activations (activation saturation)

Dynamics of Loss and Accuracy

The training loss and accuracy were graphed across epochs for each initialization scale. These measurements demonstrate the speed and reliability of convergence for each model. Reduced σ values typically result in sluggish convergence owing to minimal initial gradients, whereas elevated σ values can induce oscillatory or unstable learning due to excessive activations.

Gradient Norm Monitoring

We assessed stability by tracking the L2 norm of all parameter gradients at each epoch. Gradient norms that diminish towards zero signify vanishing gradients, while pronounced spikes or escalating norms suggest the emergence of gradient explosion. This measurement offers direct proof of the impact of initialization on signal propagation during backpropagation.

Analysis of Activation Saturation

The ReLU function produces zero for negative inputs, therefore excessively small weights can drive most activations to zero, thus rendering neurons inactive. In contrast, excessively large initial weights expand the activation distribution and may lead to unpredictable learning. Monitoring the proportion of saturated activations during training offers insight into these behaviors.

Evaluation of Performance

Subsequent to training, each model underwent evaluation on the reserved test set. The ultimate accuracies—0.8733 (σ=0.01), 0.9483 (σ=0.1), 0.9367 (σ=0.5), and 0.9183 (σ=1.0)—indicate that moderate initialization scales produce optimal generalization performance.

Outcomes and Analysis

The experimental results unequivocally indicate that the scale of weight initialization significantly affects convergence behavior, gradient dynamics, and overall model performance. The four studied σ values—0.01, 0.1, 0.5, and 1.0—yield diverse learning patterns, as depicted in Figures 1–4.

Behavior of Loss Convergence

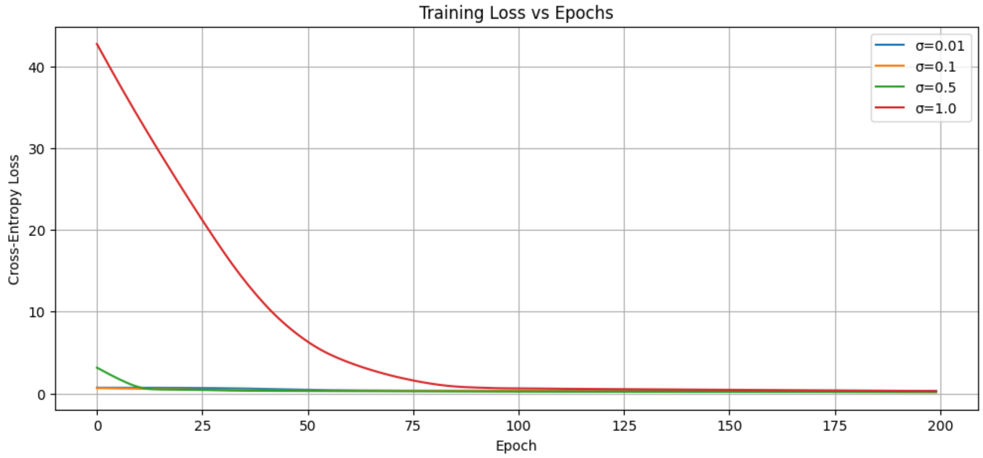

Figure 1. Line plot comparing loss across epochs for σ ∈ {0.01, 0.1, 0.5, 1.0}.

Figure 1 (Training Loss vs. Epochs) illustrates that a minimal initialization (σ = 0.01) results in sluggish and superficial learning, as the model requires numerous epochs to reduce its loss. This is anticipated, as minimal beginning weights produce diminutive activations, leading to feeble gradients during training.

Conversely, σ = 0.1 attains the most rapid and steady convergence, characterized by a smooth loss curve devoid of oscillations. This scale seems to preserve equilibrium between gradient magnitude and activation variability.

Increased scales (σ = 0.5 and 1.0) yield less stable loss curves, with σ = 1.0 occasionally producing noisy or jagged behavior, signifying minor gradient instability.

Trends in Accuracy

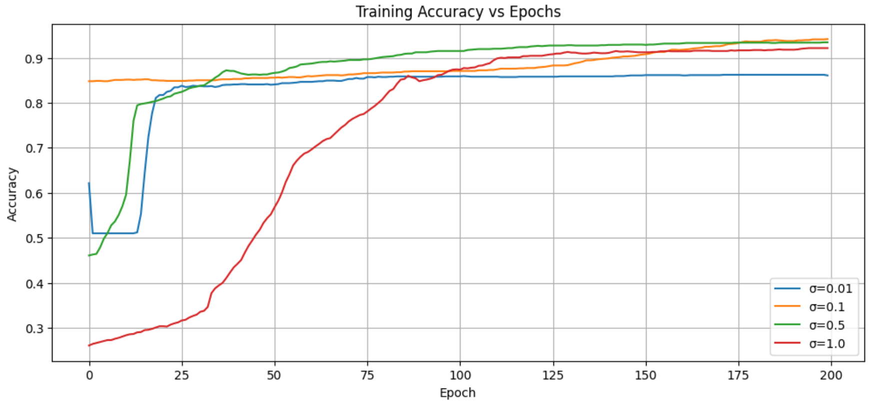

Figure 2. Line plot showing accuracy improvement across epochs for four initialization scales.

Figure 2 (Accuracy versus Epochs) corroborates these findings. Models with σ = 0.1 and σ = 0.5 achieve high accuracy rapidly, but σ = 0.01 exhibits a slow improvement. The σ = 1.0 model initially exhibits rapid learning but experiences fluctuations, indicative of the unstable gradients observed at higher scales. The final test accuracies further validate this trend:

Table 1. Final Test Accuracy for Each Initialization Scale

Initialization Scale σ	Test Accuracy
0.01	0.8733
0.1	**0.9483**
0.5	0.9367
1	0.9183

The optimal accuracy of 94.83% is attained with σ = 0.1, underscoring the empirical efficacy of moderate initialization scales.

Analysis of Gradient Norms

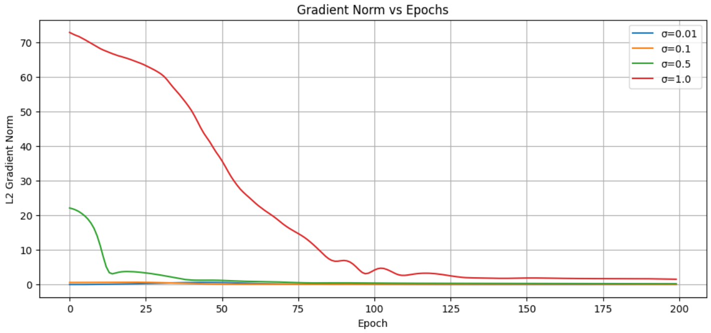

Figure 3. Line plot showing gradient norms for σ ∈ {0.01, 0.1, 0.5, 1.0}

Figure 3 (Gradient Norm vs Epochs) offers a more profound understanding of optimization stability. For σ = 0.01, gradient norms are exceedingly minimal, signifying vanishing gradients. For σ = 1.0, gradient norms exhibit significant spikes and swings, indicative of a nascent gradient explosion. The σ = 0.1 and 0.5 curves maintain smoothness, indicating that these scales sustain sustained gradient flow.

Activation Saturation

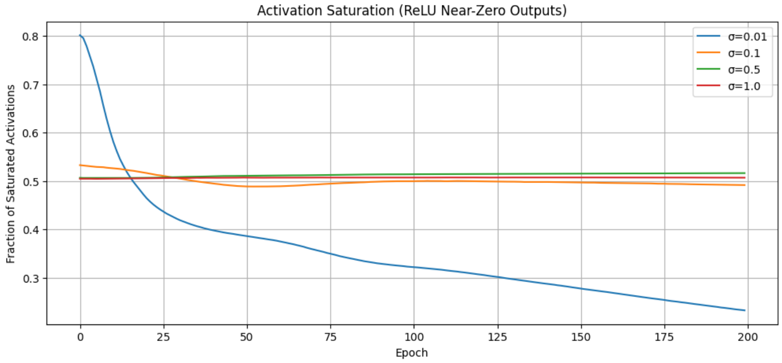

Figure 4 Line plot of activation saturation vs epochs for different sigma values.

Figure 4 (ReLU Activation Saturation) illustrates the percentage of activations approaching zero. σ = 0.01 demonstrates maximal saturation, resulting in numerous inactive neurons and impeding the learning process. σ = 1.0 results in reduced saturation while augmenting variance, hence causing erratic learning. σ = 0.1 achieves an optimal equilibrium between active and stable neurons.

Comprehensive Analysis

The findings robustly endorse a fundamental tenet in neural network optimization:
Weight initialization must equilibrate activation magnitude and gradient stability.

Insufficient size (σ = 0.01): diminishing gradients, dormant neurons, sluggish learning

Excessive size (σ = 1.0): unstable gradients, erratic convergence

Moderate scale (σ = 0.1): optimal convergence speed, superior accuracy, steady gradients

These patterns correspond exactly with theoretical predictions from variance propagation and ReLU behavior, highlighting the essential influence of initialization scale on training dynamics.

Final Analysis

This study illustrates that the magnitude of weight initialization significantly influences neural network convergence and optimization stability. Minuscule initial weights result in vanishing gradients and sluggish learning, whereas overly large weights cause unstable gradients and erratic training behavior. In all assessed metrics—loss, accuracy, gradient norms, and activation saturation—a moderate initialization scale (σ = 0.1) consistently produced the quickest convergence and greatest accuracy. These findings corroborate theoretical principles of variance preservation and underscore the significance of meticulously selecting appropriately scaled beginning weights. Consequently, effective initialization is essential for training dependable and efficient neural networks.

References

Glorot, X., & Bengio, Y. (2010). *Understanding the difficulty of training deep feedforward neural networks*. In Proceedings of the 13th International Conference on Artificial Intelligence and Statistics (AISTATS), 249–256.

He, K., Zhang, X., Ren, S., & Sun, J. (2015). *Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification*. Proceedings of the IEEE International Conference on Computer Vision (ICCV), 1026–1034.

LeCun, Y., Bottou, L., Orr, G. B., & Müller, K.-R. (2012). *Efficient backprop*. In G. Montavon, G. B. Orr, & K.-R. Müller (Eds.), Neural networks: Tricks of the trade (pp. 9–48). Springer.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.
Retrieved from https://www.deeplearningbook.org/

Sutskever, I., Martens, J., Dahl, G., & Hinton, G. (2013). *On the importance of initialization and momentum in deep learning*. Proceedings of the 30th International Conference on Machine Learning (ICML), 1139–1147.

Nair, V., & Hinton, G. E. (2010). *Rectified linear units improve restricted Boltzmann machines*. Proceedings of the 27th International Conference on Machine Learning (ICML).