BOSTON UNIVERSITY

Lecture 09 Initialization

DL4DS – Spring 2026

Agenda

- Quick tips on how to read a research paper
- Model Initialization
- Example code

Where we are

=== Foundational Concepts ===

- ✓ 04 -- Shallow networks and their representation capacity
- ✓ 05 -- Deep networks and depth efficiency
- ✓ 06 -- Loss function in terms of maximizing likelihoods
- ✓ 07 – Fitting models with different optimizers
- ✓ 08 – Gradients on deep models and backpropagation
- 09 – Initialization to avoid vanishing and exploding weights & gradients
- 10 – Measuring performance, test sets, overfitting and double descent
- 11 – Regularization to improve fitting on test sets and unseen data

=== Network Architectures and Applications ===

- 12 – Convolutional Networks
- 13 – Residual Networks
- 14 – Transformers
- Large Language and other Foundational Models
- Generative Models
- Graph Neural Networks
- …

Model Initialization

- The need for weights initialization
- Expectations Refresher
- The He (Kaiming) Initialization
- The Lottery Ticket Hypothesis

Initialization

- Consider standard building block of NN in terms of pre-activations:

$$\begin{aligned}\mathbf{f}_k &= \boldsymbol{\beta}_k + \boldsymbol{\Omega}_k \mathbf{h}_k \\ &= \boldsymbol{\beta}_k + \boldsymbol{\Omega}_k \mathrm{a}[\mathbf{f}_{k-1}]\end{aligned}$$

- How do we initialize the biases and weights?
- Equivalent to choosing starting point in our gradient descent searches

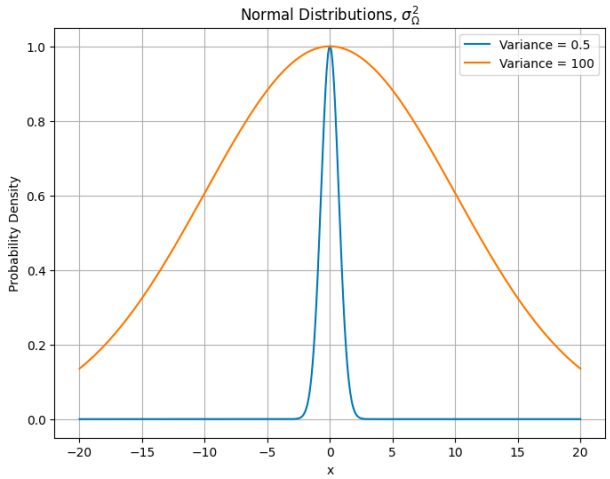
Forward Pass

- Consider standard building block of NN in terms of *pre-activations*:

$$\begin{aligned}\mathbf{f}_k &= \boldsymbol{\beta}_k + \boldsymbol{\Omega}_k \mathbf{h}_k \\ &= \boldsymbol{\beta}_k + \boldsymbol{\Omega}_k \mathrm{a}[\mathbf{f}_{k-1}]\end{aligned}$$

- Set all the biases to 0

$$\boldsymbol{\beta}_k = \mathbf{0}$$



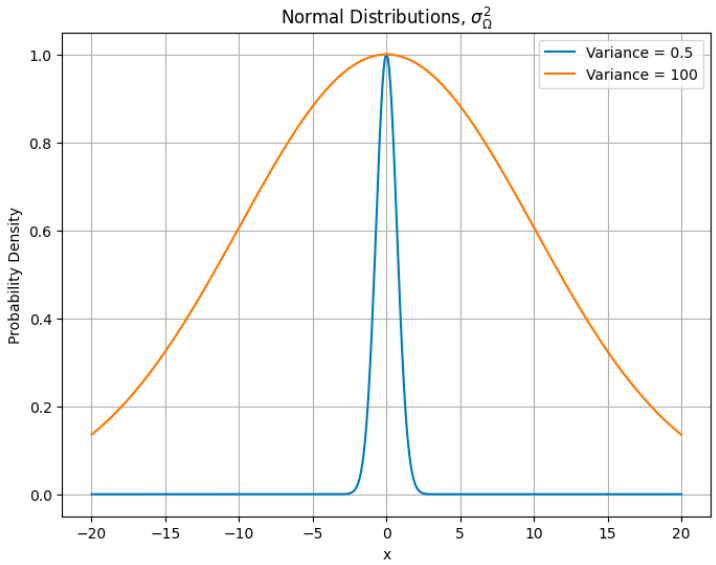
- Set weights to be normally distributed
  - mean 0
  - variance σ_Ω^2

- What will happen as we move through the network if σ_Ω^2 is very small?
- What will happen as we move through the network if σ_Ω^2 is very large?

Backward Pass

$$\frac{\partial \ell_i}{\partial \mathbf{f}_{k-1}} = \mathbb{I}[\mathbf{f}_{k-1} > 0] \odot \left(\boldsymbol{\Omega}_k^T \frac{\partial \ell_i}{\partial \mathbf{f}_k} \right), \qquad k \in \{K, K-1, \ldots 1\} \qquad (7.13)$$

- What will happen as we propagate backwards through the network if σ_Ω^2 is very small?
- What will happen as we propagate backwards through the network if σ_Ω^2 is very large?



Initialize weights to different variances

100D Input
~ $N(0,1)$

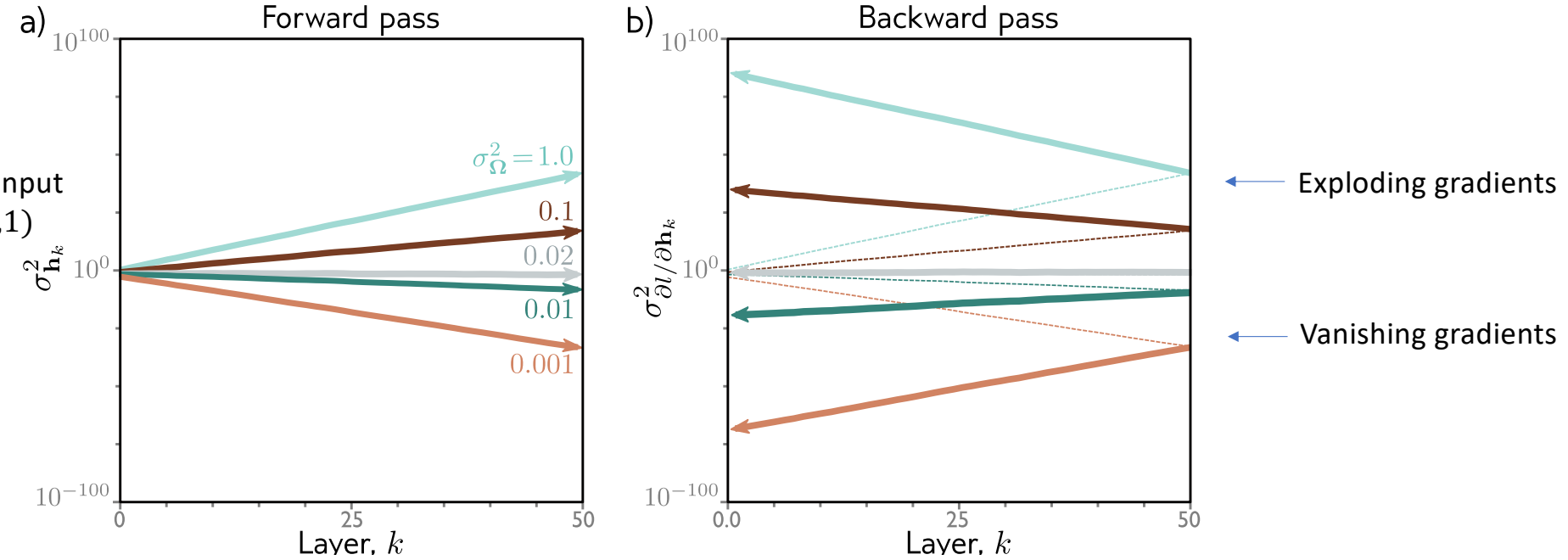


Figure 7.4 Weight initialization. Consider a deep network with 50 hidden layers and $D_h = 100$ hidden units per layer. The network has a 100 dimensional input $\mathbf{x}$ initialized with values from a standard normal distribution, a single output fixed at $y = 0$, and a least squares loss function. The bias vectors $\boldsymbol{\beta}_k$ are initialized to zero and the weight matrices $\boldsymbol{\Omega}_k$ are initialized with a normal distribution with mean zero and five different variances $\sigma^2_{\Omega} \in \{0.001, 0.01, 0.02, 0.1, 1.0\}$. a)

How do we initialize weights to keep variance stable across layers?

Aim: keep variance same between two layers

$$\mathbf{f}' = \boldsymbol{\beta} + \boldsymbol{\Omega}\mathbf{h}$$
$$\mathbf{h} = \mathbf{a}[\mathbf{f}],$$

Definition of variance:

$$\sigma_{f'}^2 = \mathbb{E}[(f_i' - \mathbb{E}[f_i'])^2]$$

Agenda

- The need for weights initialization
- Expectations Refresher
- The He (Kaiming) Initialization
- The Lottery Ticket Hypothesis

Expectations

$$\mathbb{E}\Big[\mathrm{g}[x]\Big] = \int \mathrm{g}[x] Pr(x) dx,$$

Interpretation: what is the average value of g[x] when taking into account the probability of x?

Consider discrete case and assume uniform probability so calculating g[x] reduces to taking average:

$$\mathbb{E}\Big[\mathrm{g}[x]\Big] \approx \frac{1}{N}\sum_{n=1}^{N} \mathrm{g}[x_n^*] \qquad \text{where} \qquad x_n^* \sim Pr(x)$$

Common Expectation Functions

Function g[•]	**Expectation**
x	mean, μ
x^k	kth moment about zero
$(x-\mu)^k$	kth moment about the mean
$(x-\mu)^2$	variance
$(x-\mu)^3$	skew
$(x-\mu)^4$	kurtosis

Table B.1 Special cases of expectation. For some functions $g[x]$, the expectation $\mathbb{E}[g[x]$ is given a special name. Here we use the notation μ_x to represent the mean with respect to random variable x.

Rules for manipulating expectation

$$\mathbb{E}\Big[k\Big] = k$$

$$\mathbb{E}\Big[k \cdot \mathrm{g}[x]\Big] = k \cdot \mathbb{E}\Big[\mathrm{g}[x]\Big]$$

$$\mathbb{E}\Big[\mathrm{f}[x] + \mathrm{g}[x]\Big] = \mathbb{E}\Big[\mathrm{f}[x]\Big] + \mathbb{E}\Big[\mathrm{g}[x]\Big]$$

$$\mathbb{E}\Big[\mathrm{f}[x]g[y]\Big] = \mathbb{E}\Big[\mathrm{f}[x]\Big]\mathbb{E}\Big[\mathrm{g}[y]\Big] \qquad \text{if} \quad x, y \quad \text{independent}$$

Agenda

- The need for weights initialization
- Expectations Refresher
- The He (Kaiming) Initialization
- The Lottery Ticket Hypothesis

Aim: keep variance same between two layers

$$\mathbf{h} = \mathbf{a}[\mathbf{f}],$$
$$\mathbf{f}' = \boldsymbol{\beta} + \boldsymbol{\Omega}\mathbf{h}$$

Definition of variance:

$$\sigma_{f_i'}^2 = \mathbb{E}[(f_i' - \mathbb{E}[f_i'])^2]$$

Now let's prove:

$$\mathbb{E}\left[(x-\mu)^2\right] = \mathbb{E}[x^2] - \mathbb{E}[x]^2$$

Keeping in mind:

$$\mathbb{E}[x] = \mu$$

Rule 1: $\mathbb{E}\Big[k\Big] = k$

Rule 2: $\mathbb{E}\Big[k \cdot \mathrm{g}[x]\Big] = k \cdot \mathbb{E}\Big[\mathrm{g}[x]\Big]$

Rule 3: $\mathbb{E}\Big[\mathrm{f}[x] + \mathrm{g}[x]\Big] = \mathbb{E}\Big[\mathrm{f}[x]\Big] + \mathbb{E}\Big[\mathrm{g}[x]\Big]$

Def'n $\mathbb{E}[x] = \mu$

$$\mathbb{E}[(x - \mu^2)] = \mathbb{E}[x^2 - 2x\mu + \mu^2]$$

Rule 1: $\mathbb{E}\Big[k\Big] = k$

Rule 2: $\mathbb{E}\Big[k \cdot \mathrm{g}[x]\Big] = k \cdot \mathbb{E}\Big[\mathrm{g}[x]\Big]$

Rule 3: $\mathbb{E}\Big[\mathrm{f}[x] + \mathrm{g}[x]\Big] = \mathbb{E}\Big[\mathrm{f}[x]\Big] + \mathbb{E}\Big[\mathrm{g}[x]\Big]$

Def'n $\mathbb{E}[x] = \mu$

$$\mathbb{E}[(x - \mu^2)] = \mathbb{E}[x^2 - 2x\mu + \mu^2]$$

$$= \mathbb{E}[x^2] - \mathbb{E}[2x\mu] + \mathbb{E}[\mu^2]$$

Rule 1: $\mathbb{E}\Big[k\Big] = k$

Rule 2: $\mathbb{E}\Big[k \cdot \mathrm{g}[x]\Big] = k \cdot \mathbb{E}\Big[\mathrm{g}[x]\Big]$

Rule 3: $\mathbb{E}\Big[\mathrm{f}[x] + \mathrm{g}[x]\Big] = \mathbb{E}\Big[\mathrm{f}[x]\Big] + \mathbb{E}\Big[\mathrm{g}[x]\Big]$

Def'n $\mathbb{E}[x] = \mu$

$$\begin{aligned}\mathbb{E}[(x-\mu^2)] &= \mathbb{E}[x^2 - 2x\mu + \mu^2] \\ &= \mathbb{E}[x^2] - \mathbb{E}[2x\mu] + \mathbb{E}[\mu^2] \\ &= \mathbb{E}[x^2] - 2\mu\mathbb{E}[x] + \mu^2\end{aligned}$$

Rule 1: $\mathbb{E}\Big[k\Big] = k$

Rule 2: $\mathbb{E}\Big[k \cdot \mathrm{g}[x]\Big] = k \cdot \mathbb{E}\Big[\mathrm{g}[x]\Big]$

Rule 3: $\mathbb{E}\Big[\mathrm{f}[x] + \mathrm{g}[x]\Big] = \mathbb{E}\Big[\mathrm{f}[x]\Big] + \mathbb{E}\Big[\mathrm{g}[x]\Big]$

Def'n $\mathbb{E}[x] = \mu$

$$\begin{aligned}
\mathbb{E}[(x-\mu^2)] &= \mathbb{E}[x^2 - 2x\mu + \mu^2] \\
&= \mathbb{E}[x^2] - \mathbb{E}[2x\mu] + \mathbb{E}[\mu^2] \\
&= \mathbb{E}[x^2] - 2\mu\mathbb{E}[x] + \mu^2 \\
&= \mathbb{E}[x^2] - 2\mu^2 + \mu^2
\end{aligned}$$

Rule 1: $\mathbb{E}\Big[k\Big] = k$

Rule 2: $\mathbb{E}\Big[k \cdot \mathrm{g}[x]\Big] = k \cdot \mathbb{E}\Big[\mathrm{g}[x]\Big]$

Rule 3: $\mathbb{E}\Big[\mathrm{f}[x] + \mathrm{g}[x]\Big] = \mathbb{E}\Big[\mathrm{f}[x]\Big] + \mathbb{E}\Big[\mathrm{g}[x]\Big]$

Def'n $\mathbb{E}[x] = \mu$

$$\begin{aligned}
\mathbb{E}[(x-\mu^2)] &= \mathbb{E}[x^2 - 2x\mu + \mu^2] \\
&= \mathbb{E}[x^2] - \mathbb{E}[2x\mu] + \mathbb{E}[\mu^2] \\
&= \mathbb{E}[x^2] - 2\mu\mathbb{E}[x] + \mu^2 \\
&= \mathbb{E}[x^2] - 2\mu^2 + \mu^2 \\
&= \mathbb{E}[x^2] - \mu^2
\end{aligned}$$

Rule 1: $\mathbb{E}\Big[k\Big] = k$

Rule 2: $\mathbb{E}\Big[k \cdot \mathrm{g}[x]\Big] = k \cdot \mathbb{E}\Big[\mathrm{g}[x]\Big]$

Rule 3: $\mathbb{E}\Big[\mathrm{f}[x] + \mathrm{g}[x]\Big] = \mathbb{E}\Big[\mathrm{f}[x]\Big] + \mathbb{E}\Big[\mathrm{g}[x]\Big]$

Def'n $\mathbb{E}[x] = \mu$

$$\begin{aligned}
\mathbb{E}[(x-\mu^2)] &= \mathbb{E}[x^2 - 2x\mu + \mu^2] \\
&= \mathbb{E}[x^2] - \mathbb{E}[2x\mu] + \mathbb{E}[\mu^2] \\
&= \mathbb{E}[x^2] - 2\mu\mathbb{E}[x] + \mu^2 \\
&= \mathbb{E}[x^2] - 2\mu^2 + \mu^2 \\
&= \mathbb{E}[x^2] - \mu^2 \\
&= \mathbb{E}[x^2] - E[x]^2
\end{aligned}$$

Aim: keep variance same between two layers

$$\mathbf{f}' = \boldsymbol{\beta} + \boldsymbol{\Omega}\mathbf{h}$$
$$\mathbf{h} = \mathbf{a}[\mathbf{f}],$$

$$\sigma_{f'}^2 = \mathbb{E}[(f_i' - \mathbb{E}[f_i'])^2]$$

$$\sigma_{f'}^2 = \mathbb{E}[f_i'^2] - \mathbb{E}[f_i']^2$$

From $\mathbb{E}\left[(x-\mu)^2\right] = \mathbb{E}[x^2] - \mathbb{E}[x]^2$

Aim: keep variance same between two layers

$$\mathbf{f}' = \boldsymbol{\beta} + \boldsymbol{\Omega}\mathbf{h}$$
$$\mathbf{h} = \mathbf{a}[\mathbf{f}],$$

$$\sigma_{f'}^2 = \mathbb{E}[(f_i' - \mathbb{E}[f_i'])^2]$$

$$\sigma_{f'}^2 = \mathbb{E}[f_i'^2] - \mathbb{E}[f_i']^2$$

Aim: keep variance same between two layers

$$\mathbf{f}' = \boldsymbol{\beta} + \boldsymbol{\Omega}\mathbf{h}$$

Consider the mean of the pre-activations:

$$\mathbb{E}[f_i'] = \mathbb{E}\left[\beta_i + \sum_{j=1}^{D_h} \Omega_{ij} h_j\right]$$

Rule 1: $\mathbb{E}\Big[k\Big] = k$

Rule 2: $\mathbb{E}\Big[k \cdot \mathrm{g}[x]\Big] = k \cdot \mathbb{E}\Big[\mathrm{g}[x]\Big]$

Rule 3: $\mathbb{E}\Big[\mathrm{f}[x] + \mathrm{g}[x]\Big] = \mathbb{E}\Big[\mathrm{f}[x]\Big] + \mathbb{E}\Big[\mathrm{g}[x]\Big]$

Rule 4: $\mathbb{E}\Big[\mathrm{f}[x]g[y]\Big] = \mathbb{E}\Big[\mathrm{f}[x]\Big]\mathbb{E}\Big[\mathrm{g}[y]\Big] \quad \text{if} \quad x, y \quad \text{independent}$

$$\mathbb{E}[f_i'] = \mathbb{E}\left[\beta_i + \sum_{j=1}^{D_h} \Omega_{ij} h_j\right]$$

$$= \mathbb{E}\left[\beta_i\right] + \sum_{j=1}^{D_h} \mathbb{E}\left[\Omega_{ij} h_j\right]$$

Rule 1: $$\mathbb{E}\Big[k\Big] = k$$

Rule 2: $$\mathbb{E}\Big[k \cdot \mathrm{g}[x]\Big] = k \cdot \mathbb{E}\Big[\mathrm{g}[x]\Big]$$

Rule 3: $$\mathbb{E}\Big[\mathrm{f}[x] + \mathrm{g}[x]\Big] = \mathbb{E}\Big[\mathrm{f}[x]\Big] + \mathbb{E}\Big[\mathrm{g}[x]\Big]$$

Rule 4: $$\mathbb{E}\Big[\mathrm{f}[x]g[y]\Big] = \mathbb{E}\Big[\mathrm{f}[x]\Big]\mathbb{E}\Big[\mathrm{g}[y]\Big] \quad \text{if} \quad x, y \quad \text{independent}$$

$$\mathbb{E}[f_i'] = \mathbb{E}\left[\beta_i + \sum_{j=1}^{D_h} \Omega_{ij} h_j\right]$$

$$= \mathbb{E}\left[\beta_i\right] + \sum_{j=1}^{D_h} \mathbb{E}\left[\Omega_{ij} h_j\right]$$

$$= \mathbb{E}\left[\beta_i\right] + \sum_{j=1}^{D_h} \mathbb{E}\left[\Omega_{ij}\right] \mathbb{E}\left[h_j\right]$$

Rule 1: $\mathbb{E}\Big[k\Big] = k$

Rule 2: $\mathbb{E}\Big[k \cdot \mathrm{g}[x]\Big] = k \cdot \mathbb{E}\Big[\mathrm{g}[x]\Big]$

Rule 3: $\mathbb{E}\Big[\mathrm{f}[x] + \mathrm{g}[x]\Big] = \mathbb{E}\Big[\mathrm{f}[x]\Big] + \mathbb{E}\Big[\mathrm{g}[x]\Big]$

Rule 4: $\mathbb{E}\Big[\mathrm{f}[x]g[y]\Big] = \mathbb{E}\Big[\mathrm{f}[x]\Big]\mathbb{E}\Big[\mathrm{g}[y]\Big]$ if x, y independent

$$\mathbb{E}[f_i'] = \mathbb{E}\left[\beta_i + \sum_{j=1}^{D_h} \Omega_{ij} h_j\right]$$

$$= \mathbb{E}\left[\beta_i\right] + \sum_{j=1}^{D_h} \mathbb{E}\left[\Omega_{ij} h_j\right]$$

$$= \mathbb{E}\left[\beta_i\right] + \sum_{j=1}^{D_h} \mathbb{E}\left[\Omega_{ij}\right]\mathbb{E}\left[h_j\right]$$

$$= 0 + \sum_{j=1}^{D_h} 0 \cdot \mathbb{E}\left[h_j\right] = 0$$

Start making initialization choices.

- Set all the biases to 0
- Weights normally distributed
  - mean 0
  - variance σ_Ω^2

Aim: keep variance same between two layers

$$\mathbf{f}' = \boldsymbol{\beta} + \boldsymbol{\Omega}\mathbf{h}$$
$$\mathbf{h} = \mathbf{a}[\mathbf{f}],$$

$$\sigma_{f'}^2 = \mathbb{E}[(f_i' - \mathbb{E}[f_i'])^2]$$

$$\sigma_{f'}^2 = \mathbb{E}[f_i'^2] - \mathbb{E}[f_i']^2 = \mathbb{E}[f_i'^2]$$

0

Rule 1: $$\mathbb{E}\Big[k\Big] = k$$

Rule 2: $$\mathbb{E}\Big[k \cdot \mathrm{g}[x]\Big] = k \cdot \mathbb{E}\Big[\mathrm{g}[x]\Big]$$

Rule 3: $$\mathbb{E}\Big[\mathrm{f}[x] + \mathrm{g}[x]\Big] = \mathbb{E}\Big[\mathrm{f}[x]\Big] + \mathbb{E}\Big[\mathrm{g}[x]\Big]$$

Rule 4: $$\mathbb{E}\Big[\mathrm{f}[x]g[y]\Big] = \mathbb{E}\Big[\mathrm{f}[x]\Big]\mathbb{E}\Big[\mathrm{g}[y]\Big] \quad \text{if} \quad x, y \quad \text{independent}$$

$$\begin{aligned}\sigma^2_{f'} &= \mathbb{E}[f_i'^2] - \mathbb{E}[f_i']^2 \\ &= \mathbb{E}\left[\left(\beta_i + \sum_{j=1}^{D_h} \Omega_{ij} h_j\right)^2\right] - 0\end{aligned}$$

Start making initialization choices.

- Set all the biases to 0
- Weights normally distributed
  - mean 0
  - variance σ^2_Ω

Rule 1: $\mathbb{E}\Big[k\Big] = k$

Rule 2: $\mathbb{E}\Big[k \cdot \mathrm{g}[x]\Big] = k \cdot \mathbb{E}\Big[\mathrm{g}[x]\Big]$

Rule 3: $\mathbb{E}\Big[\mathrm{f}[x] + \mathrm{g}[x]\Big] = \mathbb{E}\Big[\mathrm{f}[x]\Big] + \mathbb{E}\Big[\mathrm{g}[x]\Big]$

Rule 4: $\mathbb{E}\Big[\mathrm{f}[x]g[y]\Big] = \mathbb{E}\Big[\mathrm{f}[x]\Big]\mathbb{E}\Big[\mathrm{g}[y]\Big]$ if x, y independent

$$\sigma^2_{f'} = \mathbb{E}[f_i'^2] - \mathbb{E}[f_i']^2$$

$$= \mathbb{E}\left[\left(\beta_i + \sum_{j=1}^{D_h} \Omega_{ij} h_j\right)^2\right] - 0$$

$$= \mathbb{E}\left[\left(\sum_{j=1}^{D_h} \Omega_{ij} h_j\right)^2\right]$$

Start making initialization choices.

- Set all the biases to 0

- Weights normally distributed
  - mean 0
  - variance σ^2_Ω

Rule 1: $\mathbb{E}\left[k\right] = k$

Rule 2: $\mathbb{E}\left[k \cdot \mathrm{g}[x]\right] = k \cdot \mathbb{E}\left[\mathrm{g}[x]\right]$

Rule 3: $\mathbb{E}\left[\mathrm{f}[x] + \mathrm{g}[x]\right] = \mathbb{E}\left[\mathrm{f}[x]\right] + \mathbb{E}\left[\mathrm{g}[x]\right]$

Rule 4: $\mathbb{E}\left[\mathrm{f}[x]g[y]\right] = \mathbb{E}\left[\mathrm{f}[x]\right]\mathbb{E}\left[\mathrm{g}[y]\right]$ if x, y independent

$$\sigma_{f'}^2 = \mathbb{E}[f_i'^2] - \mathbb{E}[f_i']^2$$

$$= \mathbb{E}\left[\left(\beta_i + \sum_{j=1}^{D_h} \Omega_{ij} h_j\right)^2\right] - 0$$

$$= \mathbb{E}\left[\left(\sum_{j=1}^{D_h} \Omega_{ij} h_j\right)^2\right]$$

$$= \sum_{j=1}^{D_h} \mathbb{E}\left[\Omega_{ij}^2\right] \mathbb{E}\left[h_j^2\right]$$

For all the cross terms, $E\left[\Omega_{ij}\right] = 0$ so only the squared terms are left, then use independence.

Initialization choices.

- Set all the biases to 0
- Weights normally distributed
  - mean 0
  - variance σ_Ω^2

Rule 1: $\mathbb{E}\Big[k\Big] = k$

Rule 2: $\mathbb{E}\Big[k \cdot \mathrm{g}[x]\Big] = k \cdot \mathbb{E}\Big[\mathrm{g}[x]\Big]$

Rule 3: $\mathbb{E}\Big[\mathrm{f}[x] + \mathrm{g}[x]\Big] = \mathbb{E}\Big[\mathrm{f}[x]\Big] + \mathbb{E}\Big[\mathrm{g}[x]\Big]$

Rule 4: $\mathbb{E}\Big[\mathrm{f}[x]g[y]\Big] = \mathbb{E}\Big[\mathrm{f}[x]\Big]\mathbb{E}\Big[\mathrm{g}[y]\Big]$ if x, y independent

$$\begin{aligned}
\sigma^2_{f'} &= \mathbb{E}[f_i'^2] - \mathbb{E}[f_i']^2 \\
&= \mathbb{E}\left[\left(\beta_i + \sum_{j=1}^{D_h} \Omega_{ij} h_j\right)^2\right] - 0 \\
&= \mathbb{E}\left[\left(\sum_{j=1}^{D_h} \Omega_{ij} h_j\right)^2\right] \\
&= \sum_{j=1}^{D_h} \mathbb{E}\left[\Omega_{ij}^2\right] \mathbb{E}\left[h_j^2\right] \\
&= \sum_{j=1}^{D_h} \sigma_\Omega^2 \mathbb{E}\left[h_j^2\right] = \sigma_\Omega^2 \sum_{j=1}^{D_h} \mathbb{E}\left[h_j^2\right]
\end{aligned}$$

Initialization choices.

- Set all the biases to 0
- Weights normally distributed
  - mean 0
  - variance σ_Ω^2

Because the Ω's are zero mean, this is the variance.

$$\sigma^2_{f'} = \sigma^2_\Omega \sum_{j=1}^{D_h} \mathbb{E}\left[h_j^2\right]$$

$$= \sigma^2_\Omega \sum_{j=1}^{D_h} \mathbb{E}\left[\mathrm{ReLU}[f_j]^2\right]$$

$$= \sigma^2_\Omega \sum_{j=1}^{D_h} \int_{-\infty}^{\infty} \mathrm{ReLU}[f_j]^2 Pr(f_j) df_j$$ ← From the definition of expectation.

$$= \sigma^2_\Omega \sum_{j=1}^{D_h} \int_{-\infty}^{\infty} (\mathbb{I}[f_j > 0] f_j)^2 Pr(f_j) df_j$$

$$= \sigma^2_\Omega \sum_{j=1}^{D_h} \int_{0}^{\infty} f_j^2 Pr(f_j) df_j$$ ← Only positive integral limits because of ReLU

$$= \sigma^2_\Omega \sum_{j=1}^{D_h} \frac{\sigma_f^2}{2} = \frac{D_h \sigma^2_\Omega \sigma_f^2}{2}$$ ← ½ of the variance for zero mean distribution

Aim: keep variance same between two layers

Since:

$$\sigma_{f'}^2 = \frac{D_h \sigma_\Omega^2 \sigma_f^2}{2}$$

Should choose:

$$\sigma_\Omega^2 = \frac{2}{D_h}$$

To get:

$$\sigma_{f'}^2 = \sigma_f^2$$

Kaiming He 何恺明

https://people.csail.mit.edu/kaiming/

This is called He initialization or Kaiming initialization.

K. He, X. Zhang, S. Ren, and J. Sun, "Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification," Proc. IEEE International Conference on Computer Vision, 2015, pp. 1026–1034. Accessed: Feb. 11, 2024.

He initialization (assumes ReLU)

- Forward pass: want the variance of hidden unit activations in layer k+1 to be the same as variance of activations in layer k:

$$\sigma_\Omega^2 = \frac{2}{D_h}$$

Number of units at layer k (fan in)

- Backward pass: want the variance of gradients at layer k to be the same as variance of gradient in layer k+1:

$$\sigma_\Omega^2 = \frac{2}{D_{h'}}$$

Number of units at layer k+1 (fan out)

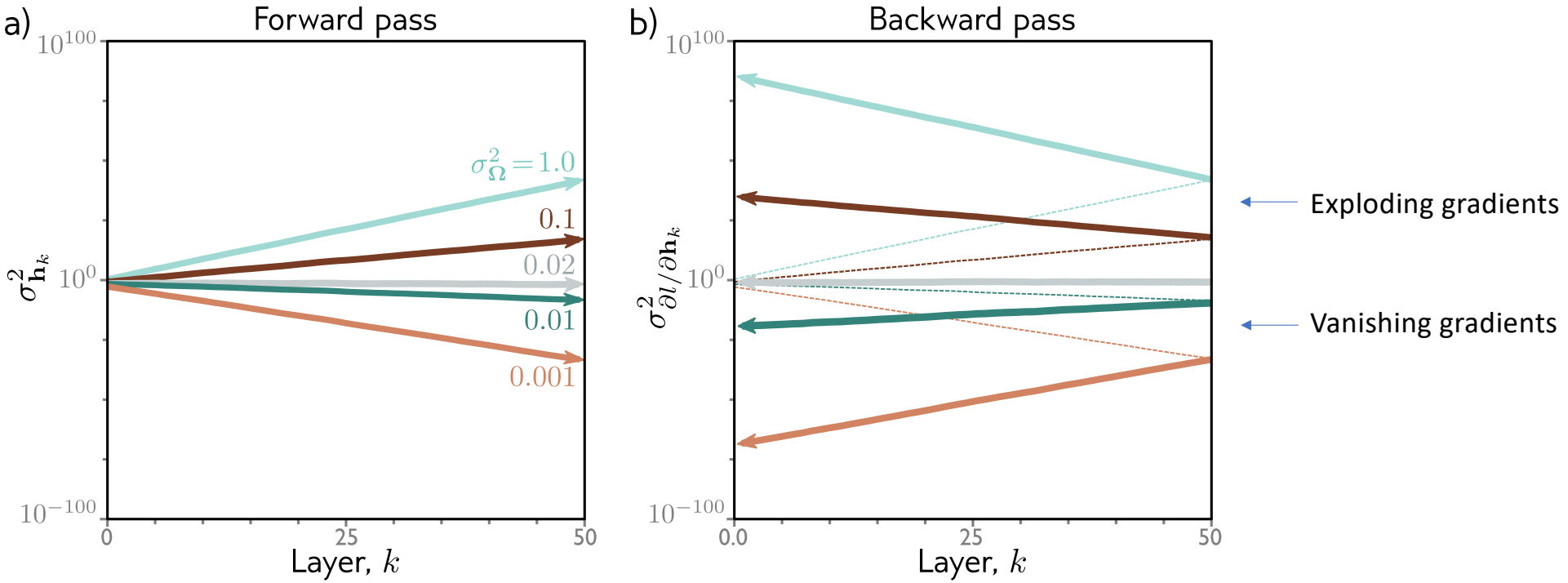


Figure 7.4 Weight initialization. Consider a deep network with 50 hidden layers and $D_h = 100$ hidden units per layer. The network has a 100 dimensional input $\mathbf{x}$ initialized with values from a standard normal distribution, a single output fixed at $y = 0$, and a least squares loss function. The bias vectors $\boldsymbol{\beta}_k$ are initialized to zero and the weight matrices $\boldsymbol{\Omega}_k$ are initialized with a normal distribution with mean zero and five different variances $\sigma^2_{\Omega} \in \{0.001, 0.01, 0.02, 0.1, 1.0\}$. a)

$$\sigma^2_{\Omega} = \frac{2}{D_h} = \frac{2}{100} = 0.02$$

Default Initialization in PyTorch

https://docs.pytorch.org/docs/stable/nn.init.html#torch.nn.init.kaiming_normal_

torch.nn.init.kaiming_normal_(*tensor*, *a=0*, *mode='fan_in'*, *nonlinearity='leaky_relu'*, *generator=None*) [source]

Fill the input *Tensor* with values using a Kaiming normal distribution.

The method is described in *Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification* - He, K. et al. (2015). The resulting tensor will have values sampled from $\mathcal{N}(0, \text{std}^2)$ where

$$\text{std} = \frac{\text{gain}}{\sqrt{\text{fan\_mode}}}$$

Also known as He initialization.

- **mode** (*Literal['fan_in', 'fan_out']*) – either `'fan_in'` (default) or `'fan_out'`. Choosing `'fan_in'` preserves the magnitude of the variance of the weights in the forward pass. Choosing `'fan_out'` preserves the magnitudes in the backwards pass.

Example Code

- Initialization Examples

- UDL Simple PyTorch Example

Initialization Note

A good initialization does not prevent gradient descent from changing the weights a lot.

- A good initialization keeps the initial gradients modestly sized,
- And modest gradients reduce wild swings in parameters with gradient descent
- Smaller learning rates also help with this.
- Next week's topic, regularization, will directly address this.

Limitations of Initialization

- No guarantees that the model will train to low losses
- No guarantees that training process won't lead to large values or gradients
- No guarantees that the model won't have lots of inactive units
  - In fact, the estimates adjusted for half being inactive!

- In fact, much of the network is often useless, and could be pruned away!

Any Questions?

???

- The need for weights initialization
- Expectations Refresher
- The (Kaiming) He initialization
- Lottery tickets

The Lottery Ticket Hypothesis

- **Core Hypothesis:** A randomly-initialized, dense neural network contains a subnetwork (a **"winning ticket"**) that is initialized such that—when trained in isolation—it can match the test accuracy of the original network in at most the same number of training iterations.
- **Identification Algorithm (Iterative Magnitude Pruning):**
  - Randomly initialize a dense network $f[x; \theta_0]$.
  - Train the network to convergence (θ_j).
  - Prune a percentage of weights with the smallest magnitudes, creating a mask m.
  - **The Critical Step:** Reset the remaining parameters to their values in the original initialization θ_0 This results in the winning ticket $f(x;\ m \odot \theta_0)$.
- **Key Finding:** Subnetworks found this way are often **10-20% the size** of the original network yet learn faster and reach higher test accuracy.

J. Frankle, M. Carbin, "The Lottery Ticket Hypothesis: Finding Sparse, Trainable Neural Networks", 2024, https://arxiv.org/abs/1803.03635

LTH: Critical Insights and Implications

- **Initialization vs. Structure:** The specific initial weights are vital. When winning ticket structures are **randomly re-initialized**, they train significantly slower and achieve lower accuracy than the original network.
- **Improved Generalization:** Winning tickets often exhibit better generalization than their dense counterparts, showing a smaller gap between training and test accuracy.
- **The Role of Overparameterization:** The authors conjecture that dense networks are easier to train because they contain a larger number of possible subnetworks, increasing the probability that one will be a "winning ticket".
- **Stability in Scale:** While effective on MNIST and CIFAR-10, identifying winning tickets on deeper architectures like ResNet-18 or VGG-19 requires **learning rate warmup** to be successful.

J. Frankle, M. Carbin, “The Lottery Ticket Hypothesis: Finding Sparse, Trainable Neural Networks”, 2024, https://arxiv.org/abs/1803.03635

Any Questions?

???

- The need for weights initialization
- Expectations Refresher
- The (Kaiming) He initialization
- Lottery tickets

Disclaimer

- Just because variance of gradients starts the same does not mean that the variance of gradients stays the same.

- You should still check the gradients if you are having training difficulties…

Bonus Tip

- If you are trying to implement a model based on a paper, and you are having trouble training, check if they shared their code.
  - 🕵️ Many papers omit important initialization details.
  - 🤔 Especially if they say that their method is not sensitive to initialization.
  - 🤬 Also, some paper descriptions of initialization don't match their code.

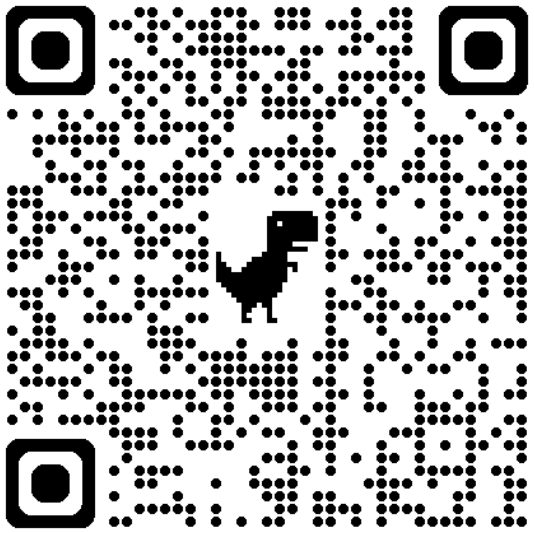
What is the primary goal of good weight initialization in deep networks?

With a very large initial weight variance, what problem are you most likely to observe?

For ReLU networks, He (Kaiming) initialization is designed so that:

The Lottery Ticket Hypothesis (LTH) claims that:

Feedback?



https://forms.gle/pXHM5nx1Ti9aFmpw6