Neural Network Analysis

Leon und Lucas

April 2024

Table of content

Math

- w_{ik}^l is the weight going from neuron i in layer $l-1$ to the neuron k in layer l
- b_k^l is the bias of neuron k in layer l
- The activation a_k^l of neuron k in layer l is:

$$a_k^l = W_{\bullet,k}^l \cdot y^{l-1} + b_k^l$$

- The output y_k^l of neuron k in layer l is

$$y_k^l = f(a_k^l)$$

- The input to the neural network is denoted as y^0

Math

- With $a^{l+1} = W^{l+1} \cdot f(a^l) + b^{l+1}$

$$\frac{\partial L}{\partial a_k^l} = \frac{\partial L}{\partial a^{l+1}} \cdot \frac{\partial a^{l+1}}{\partial a_k^l} = \frac{\partial L}{\partial a^{l+1}} \cdot W_{k,\bullet}^{l+1} f'(a_k^l)$$

- For a random variable X we have:

$$\mathbb{V}[X] = \mathbb{E}[X^2] - \mathbb{E}[X]^2$$

$$\sigma(X) = \sqrt{\mathbb{V}[X]}$$

- If X has zero mean then:

$$\mathbb{V}[X] = \mathbb{E}[X^2]$$

(This is the most important formula for this presentation, lets call it bert)

Xavier/Glorot initialization

- Problems with a bad initialization
  - Values might explode or vanish in the first forward pass
  - Gradients might vanish or explode in the first backward pass
- This leads to slow or no convergence

Xavier/Glorot initialization

- Lets make a few assumptions for Glorot to work
  - The activation is point symmetric around 0 and preserves variance around 0
    1. Tanh
    2. Does **not** work for sigmoid!
  - All w_{ik}^{l} are independent and indentically distributed (iid)
  - w_{ik}^{l} has zero mean
  - b^{l} are initialized as 0
  - y^{l-1} and w^{l} are independent
  - y^{0} are iid with a mean of 0 and a Variance of 1
- **What we want:** Variance of activations and gradients of activations should be the same for all layers
- **What we need:** Variance for weight initialization that can achieve this equality

Xavier/Glorot initialization

- $\mathbb{V}[y^0]$,
  - y^0 is a random Variable e.g. $y_k^0 \sim \mathcal{N}(0, 1)$
  - Since all y_k^0 have the same Variance, $\mathbb{V}[y^0]$ can be represented by a single value: $\mathbb{V}[y^0] \cong \mathbb{V}[y_1^0] = \mathbb{V}[y_2^0] = \ldots$
- $\mathbb{V}[w^l]$
  - w^l is a random Variable sampled from the initialization distribution
  - Again a single value due to the iid. assumption
  - **Is what we try to compute**
- $\mathbb{V}[a^l]$
  - a^l is a random Variable dependent on y^{l-1} and w^l
  - Can also be represented by a single value since all neurons before that are iid.

Xavier/Glorot initialization

- Lets analyse the Variance of the activations. We can look at a single (the first) neuron

$$\mathbb{V}[a^l] = \mathbb{V}[\sum_{i=0}^{n^{l-1}} w_{i,1}^l \cdot y^{l-1} + b_1^l] = \mathbb{V}[\sum_{i=0}^{n^{l-1}} w_{i,1}^l \cdot y_i^{l-1}] =$$

$$n^{l-1} \cdot \mathbb{V}[w^l \cdot y^{l-1}] = n^{l-1} \cdot \mathbb{V}[w^l] \cdot \mathbb{E}[(y^{l-1})^2]$$

Xavier/Glorot initialization

- Lets analyse the expected value of y^l

$$\mathbb{E}[y^l] = \mathbb{E}[f(\sum_{i=0}^{n^{l-1}} w_{i,1}^l y_i^{l-1} + b_1^l)] \stackrel{(1)}{=} \mathbb{E}[\sum_{i=0}^{n^{l-1}} w_{i,1}^l y_i^{l-1} + b_1^l] \stackrel{(2)}{=}$$

$$\sum_{i=0}^{n^{l-1}} \mathbb{E}[w_{i,1}^l y_i^{l-1}] \stackrel{(3)}{=} \sum_{i=0}^{n^{l-1}} \mathbb{E}[w_{i,1}^l]\mathbb{E}[y_i^{l-1}] = 0$$

(1) *Because f is symmetric and $w_{i,1}^l y_i^{l-1}$ is zero mean and symmetric*
(2) *Due to the linearity of $\mathbb{E}$ and $b^l = 0$*
(3) *Because w^l and y^{l-1} are independent*

Xavier/Glorot initialization

- With the knowledge that $\mathbb{E}[y^l] = 0$ for all l we can write $\mathbb{V}[a^l]$ as:

$$\mathbb{V}[a^l] = n^{l-1} \cdot \mathbb{V}[w^l] \cdot \mathbb{E}[(y^{l-1})^2] =$$

$$n^{l-1} \cdot \mathbb{V}[w^l] \cdot \mathbb{V}[(y^{l-1})] \overset{(1)}{\approx} n^{l-1} \cdot \mathbb{V}[w^l] \cdot \mathbb{V}[(a^{l-1})]$$

(1) *Activation function should be variance preserving*

- And by iteratively applying the formula:

$$\mathbb{V}[a^l] = \mathbb{V}[y^0] \prod_{i=1}^{l} n^{i-1} \cdot \mathbb{V}[w^i]$$

Xavier/Glorot initialization

- Lets look at the variance of the gradient (again looking at a single neuron is enough)

$$\mathbb{V}[\frac{\partial L}{\partial a^l}] = \mathbb{V}[\frac{\partial L}{\partial a^{l+1}} \cdot w_{1,\bullet}^{l+1} f'(a_1^l)] \approx \mathbb{V}[\frac{\partial L}{\partial a^{l+1}} \cdot w_{1,\bullet}^{l+1}] \stackrel{?}{=}$$

$$n^{l+1}\mathbb{V}[\frac{\partial L}{\partial a^{l+1}}] \cdot (\mathbb{V}[w^{l+1}] + \mathbb{E}[w^{l+1}]^2) = n^{l+1}\mathbb{V}[\frac{\partial L}{\partial a^{l+1}}] \cdot \mathbb{V}[w^{l+1}]$$

- For a Network of length L we have:

$$\mathbb{V}[\frac{\partial L}{\partial a^l}] = \mathbb{V}[\frac{\partial L}{\partial a^L}] \prod_{i=l+1}^{L} n^i \cdot \mathbb{V}[w^i]$$

Xavier/Glorot initialization

- For the forward pass we would like to have:

$$\mathbb{V}[a^i] = \mathbb{V}[a^k] \quad \forall (i, k)$$

  - From

$$\mathbb{V}[a^l] = \mathbb{V}[y^0] \prod_{i=1}^{l} n^{i-1} \cdot \mathbb{V}[w^i]$$

  - We can see that

$$n^{l-1} \mathbb{V}[w^l] = 1 \quad \forall l$$

  - So a good choice would be:

$$\mathbb{V}[w^l] = \frac{1}{n^{l-1}}$$

Xavier/Glorot initialization

- For the backward pass we would like to have:

$$\mathbb{V}[\frac{\partial L}{\partial a^i}] = \mathbb{V}[\frac{\partial L}{\partial a^k}] \quad \forall (i,k)$$

  - From

$$\mathbb{V}[\frac{\partial L}{\partial a^l}] = \mathbb{V}[\frac{\partial L}{\partial a^L}] \prod_{i=l+1}^{L} n^i \cdot \mathbb{V}[w^i]$$

  - We can see that

$$n^l \mathbb{V}[w^l] = 1 \quad \forall l$$

  - So a good choice would be:

$$\mathbb{V}[w^l] = \frac{1}{n^l}$$

Xavier/Glorot initialization

- So in total we want:

$$\mathbb{V}[w^l] = \frac{1}{n^{l-1}} \quad \wedge \quad \mathbb{V}[w^l] = \frac{1}{n^l}$$

- A middle ground would be:

$$\mathbb{V}[w^l] = \frac{2}{n^{l-1} + n^l}$$

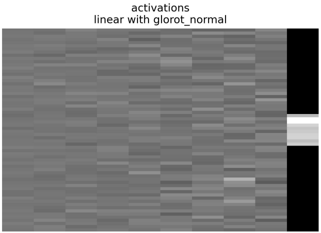
- Or as a distribution:

$$w^l \sim \mathcal{N}(0, \sqrt{\frac{2}{n^{l-1} + n^l}})$$

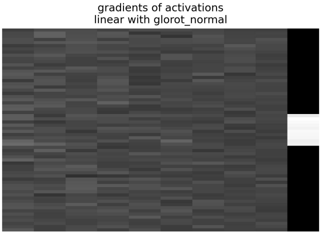
$$w^l \sim \mathcal{U}(-\sqrt{\frac{6}{n^{l-1} + n^l}}, \sqrt{\frac{6}{n^{l-1} + n^l}})$$

- n^{l-1} is also known as **fan_in** and is the size of the input of a layer, n^l is also known as **fan_out** and is the size of a layer

Xavier/Glorot initialization

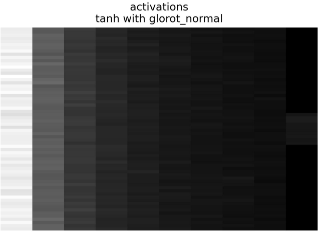


(a) variance of activations

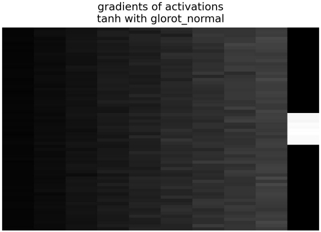


(b) variance of gradients of activations

Xavier/Glorot initialization



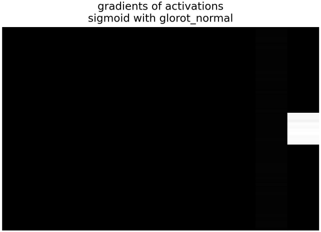
(c) variance of activations



(d) variance of gradients of activations

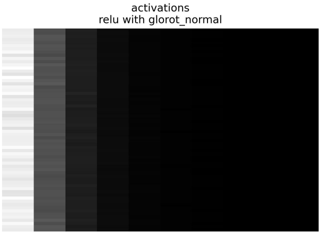
Xavier/Glorot initialization

(e) variance of activations

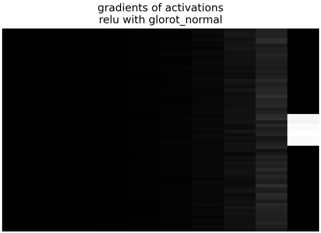


(f) variance of gradients of activations

Xavier/Glorot initialization

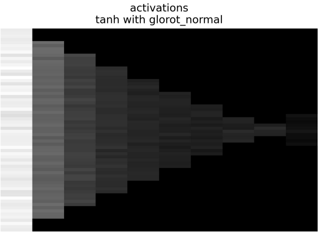


(g) variance of activations

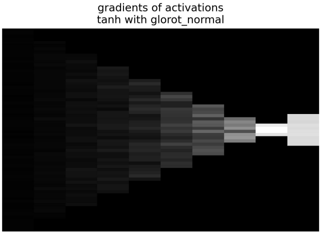


(h) variance of gradients of activations

Xavier/Glorot initialization

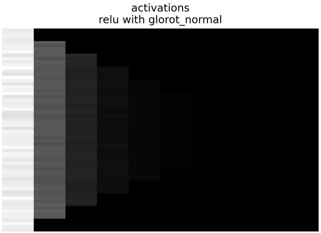


(i) variance of activations

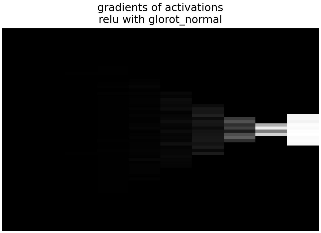


(j) variance of gradients of activations

Xavier/Glorot initialization

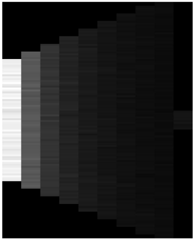


(k) variance of activations

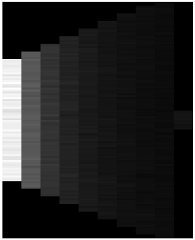


(l) variance of gradients of activations

Xavier/Glorot initialization



(m) variance of activations



(n) variance of gradients of activations

Kaiming/He initialization

- Assuming that f is linear around 0 is a strong assumption and not neccessarily true. We want to find a good initialization **for relu activation**.
- Lets revisit our old formula

$$\mathbb{V}[a^l] = n^{l-1} \cdot \mathbb{V}[w^l] \cdot \mathbb{E}[(y^{l-1})^2]$$

- $\mathbb{E}[(y^{l-1})]$ is usually not 0. So we have to find another solution.

Kaiming/He initialization

- We assume that a^l is symmetrically distributed around 0. For any layer l:

$$\mathbb{E}[(y^l)^2] = \mathbb{E}[f(a^l)^2] = \mathbb{E}[\max(0, a^l)^2]$$

$$= \int_{-\infty}^{\infty} \max(0,x)^2 p_{a^l}(x)dx = \int_{-\infty}^{0} 0^2 p_{a^l}(x)dx + \int_{0}^{\infty} x^2 p_{a^l}(x)dx$$

$$= \int_{0}^{\infty} x^2 p_{a^l}(x)dx \overset{(1)}{=} \frac{1}{2}\int_{-\infty}^{\infty} x^2 p_{a^l}(x)dx = \frac{1}{2}\mathbb{E}[(a^l)^2] \overset{(2)}{=} \frac{1}{2}\mathbb{V}[a^l]$$

(1) *Because p_{a^l} is symmetrical around 0 and $x^2 \geq 0$*
(2) *Because $\frac{1}{2}\mathbb{E}[a^l]^2 = 0$*

Kaiming/He initialization

- So in total we get:

$$\mathbb{V}[a^l] = n^{l-1} \cdot \mathbb{V}[w^l] \cdot \frac{1}{2}\mathbb{V}[a^{l-1}]$$

- So in order to keep the variance of the activations in the forward pass:

$$w^l \sim \mathcal{N}(0, \sqrt{\frac{2}{n^{l-1}}})$$

- Or more generally:

$$w^l \sim \mathcal{N}(0, \frac{gain}{\sqrt{fan\_in}})$$

- The gain for relu is $\sqrt{2}$

Kaiming/He initialization

- Lets calculate the gain for leaky relu with slope $s < 1$:

$$\mathbb{E}[(y^l)^2] = \mathbb{E}[f(a^l)^2] = \mathbb{E}[\max(s \cdot a^l, a^l)^2] =$$

$$\int_{-\infty}^{\infty} \max(s \cdot a^l, x)^2 p_{a^l}(x) dx =$$

$$\int_{-\infty}^{0} (s \cdot a^l)^2 p_{a^l}(x) dx + \int_{0}^{\infty} (a^l)^2 p_{a^l}(x) dx$$

$$= \frac{1}{2} \int_{-\infty}^{\infty} (s \cdot a^l)^2 p_{a^l}(x) dx + \frac{1}{2} \int_{-\infty}^{\infty} (a^l)^2 p_{a^l}(x) dx = \frac{1}{2}(\mathbb{V}[s \cdot a^l] + \mathbb{V}[a^l])$$

$$= \frac{1}{2}(s^2 \cdot \mathbb{V}[a^l] + \mathbb{V}[a^l]) = \frac{s^2 + 1}{2} \mathbb{V}[a^l]$$

Kaiming/He initialization

- So for leaky relu we have:

$$\mathbb{V}[a^l] = n^{l-1} \cdot \mathbb{V}[w^l] \cdot \frac{s^2+1}{2} \mathbb{V}[a^{l-1}]$$

- The desired variance for the weights would be

$$\mathbb{V}[w^l] = \frac{2}{n^{l-1} \cdot (s^2+1)}$$

- This results in the gain:

$$gain = \sqrt{\frac{2}{1+s^2}}$$

Kaiming/He initialization

- Enough math for today!! Just believe me that

$$w^l \sim \mathcal{N}(0, \frac{gain}{\sqrt{fan\_out}})$$

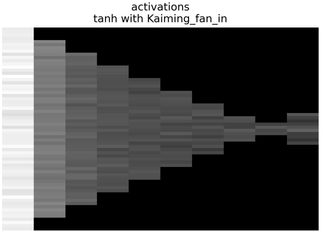
preserves the variance of gradients

Kaiming/He initialization

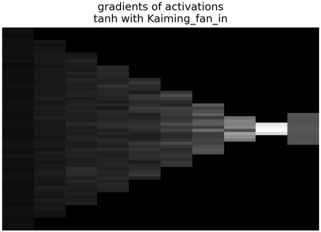
nonlinearity	gain
Linear / Identity	1
Conv{1,2,3}D	1
Sigmoid	1
Tanh	$\frac{5}{3}$
ReLU	$\sqrt{2}$
Leaky Relu	$\sqrt{\frac{2}{1+\text{negative\_slope}^2}}$
SELU	$\frac{3}{4}$

(o) https://pytorch.org/docs/stable/nn.init.html

Kaiming/He initialization

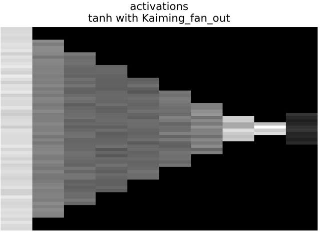


(p) variance of activations

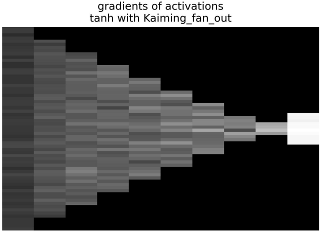


(q) variance of gradients of activations

Kaiming/He initialization

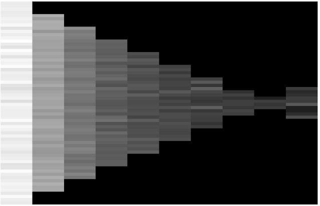


(r) variance of activations

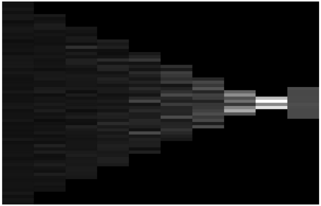


(s) variance of gradients of activations

Kaiming/He initialization

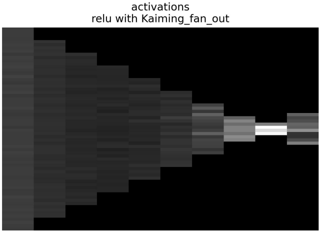


(t) variance of activations

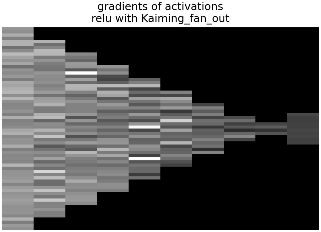


(u) variance of gradients of activations

Kaiming/He initialization

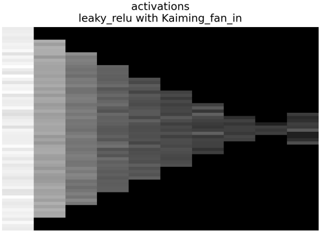


(v) variance of activations

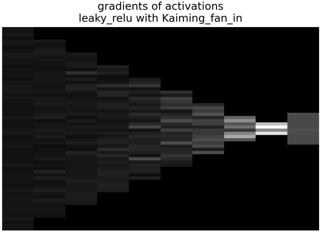


(w) variance of gradients of activations

Kaiming/He initialization

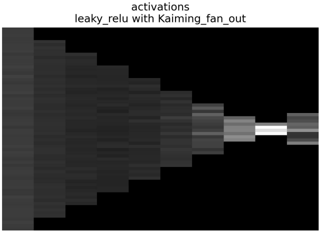


(x) variance of activations

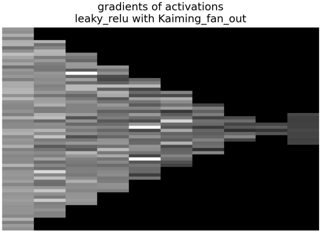


(y) variance of gradients of activations

Kaiming/He initialization



(z) variance of activations



() variance of gradients of activations

Code

LINK

Bibliography

[1] Kaiming He, Xiangyu Zhang, Shaoqing Ren, et al. *Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification*. 2015. arXiv: 1502.01852 [cs.CV].

[2] Xavier Glorot and Yoshua Bengio. "Understanding the difficulty of training deep feedforward neural networks". In: *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*. Ed. by Yee Whye Teh and Mike Titterington. Vol. 9. Proceedings of Machine Learning Research. Chia Laguna Resort, Sardinia, Italy: PMLR, 13–15 May 2010, pp. 249–256. URL: https://proceedings.mlr.press/v9/glorot10a.html.

[3] Maciej Skorski, Alessandro Temperoni, and Martin Theobald. *Revisiting Initialization of Neural Networks*. 2020. arXiv: 2004.09506 [cs.LG].

[4] Siddharth Krishna Kumar. *On weight initialization in deep neural networks*. 2017. arXiv: 1704.08863 [cs.LG].