Training Deep Neural Networks: Weights initialization

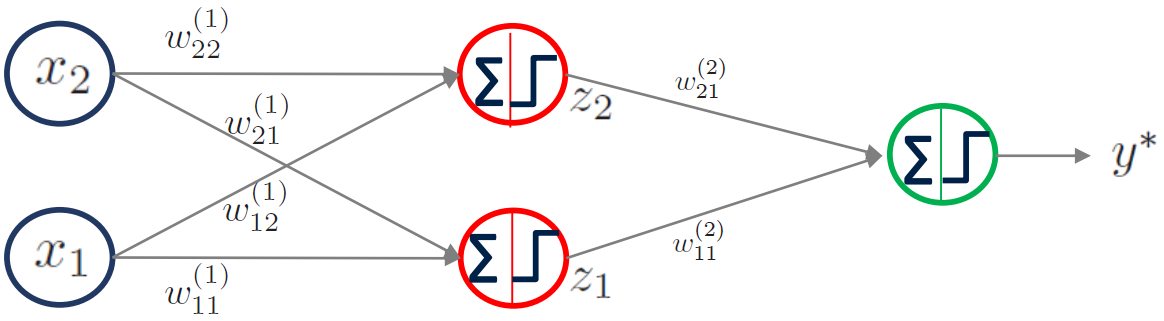
Dripta Mj
Department of Mathematics
Ramakrishna Mission Vivekananda Educational and Research Institute
Belur Math, India

What to expect……

"Modern initialization strategies are simple and heuristic. Designing improved initialization strategies is a difficult task because neural network optimization is not well understood. Most initialization strategies are based on achieving some nice properties when the network is initialized. However, we do not have a good understanding of which of these properties are preserved under which circumstances after learning begins to proceed……. Our understanding of how the initial points affects generalization is especially primitive, offering little to no guidance for how to select the initial points"

I. Goodfellow, Y. Bengio, A. Courville
Deep Learning, MIT Press

Simple case



- Activation outputs:

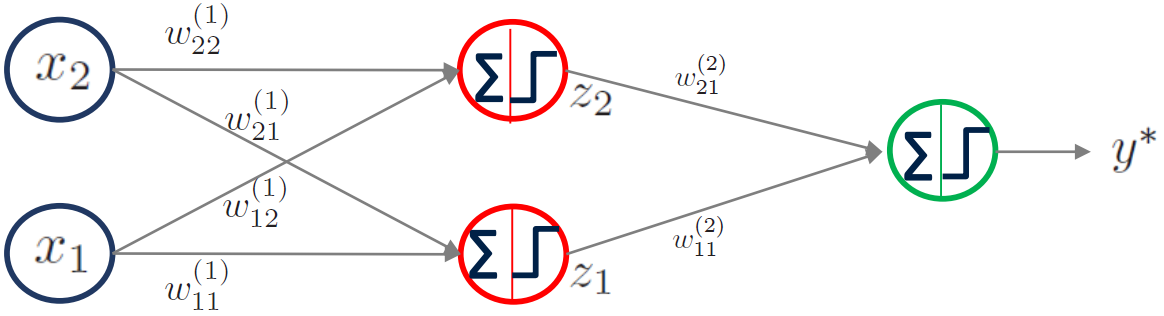
$$z_1 = \mathcal{A}\left(w_{11}^{(1)} x_1 + w_{21}^{(1)} x_2\right)$$

$$z_2 = \mathcal{A}\left(w_{21}^{(1)} x_1 + w_{22}^{(1)} x_2\right)$$

- Final outputs:

$$y^* = w_{11}^{(2)} z_1 + w_{21}^{(2)} z_2$$

Simple case: initialization with equal weights



- Consider the case in which all the weights are initialized with the same value

$$w_{11}^{(1)} = w_{21}^{(1)} = w_{21}^{(1)} = w_{22}^{(1)} = w_{11}^{(2)} = w_{21}^{(2)} = w_0$$

- In that case we have $z_1 = z_2$
- Partial derivative of the loss function w.r.t. the weights in the second layer:

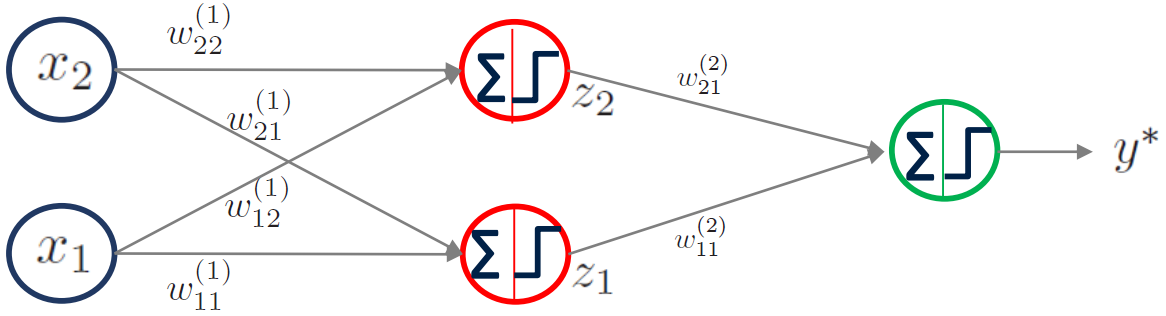
$$\frac{\partial \mathcal{L}}{\partial w_{11}^{(2)}} = \frac{\partial \mathcal{L}}{\partial y^*}\frac{\partial y^*}{\partial w_{11}^{(2)}} = \frac{\partial \mathcal{L}}{\partial y^*}\mathcal{A}'\left(w_{11}^{(2)} z_1 + w_{21}^{(2)} z_2\right) z_1$$

and

$$\frac{\partial \mathcal{L}}{\partial w_{21}^{(2)}} = \frac{\partial \mathcal{L}}{\partial y^*}\frac{\partial y^*}{\partial w_{21}^{(2)}} = \frac{\partial \mathcal{L}}{\partial y^*}\mathcal{A}'\left(w_{11}^{(2)} z_1 + w_{21}^{(2)} z_2\right) z_2$$

- Since $z_1 = z_2$, the partial derivative are the same, and so the updated value of the weights will also be the same.

Simple case: first layer



- Partial derivative of the loss function w.r.t. the weights in the first layer:

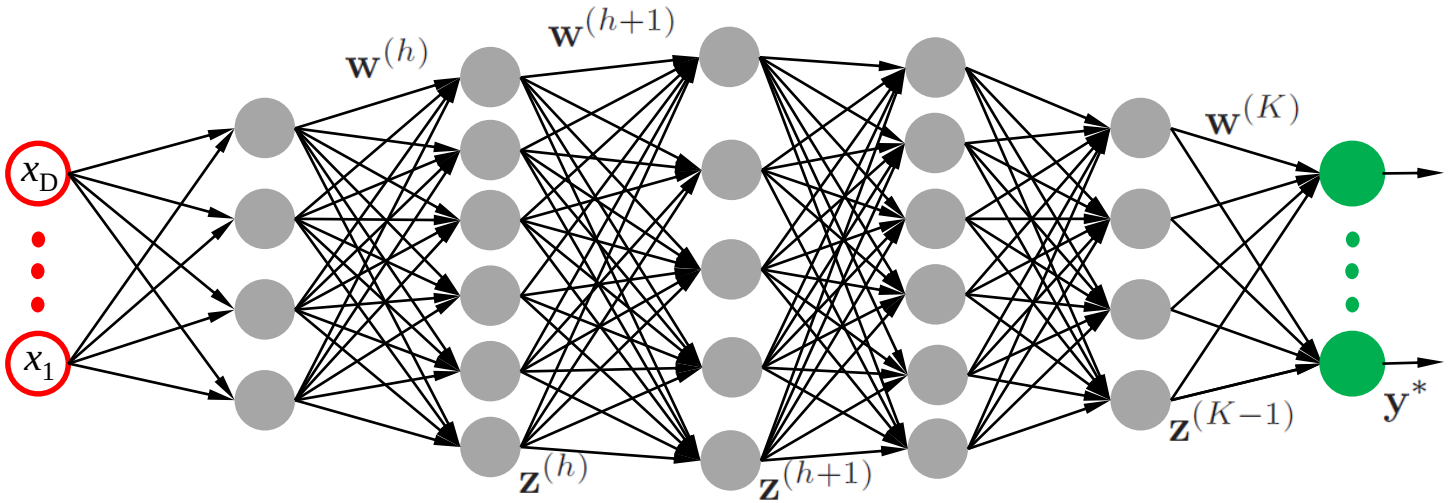
$$\frac{\partial \mathcal{L}}{\partial w_{11}^{(1)}} = \frac{\partial \mathcal{L}}{\partial y^*}\frac{\partial y^*}{\partial z_1}\frac{\partial z_1}{\partial w_{11}^{(1)}} = \frac{\partial \mathcal{L}}{\partial y^*}\frac{\partial y^*}{\partial z_1}\mathcal{A}'\left(w_0 x_1 + w_0 x_2\right)x_1$$

and

$$\frac{\partial \mathcal{L}}{\partial w_{12}^{(1)}} = \frac{\partial \mathcal{L}}{\partial y^*}\frac{\partial y^*}{\partial z_2}\frac{\partial z_2}{\partial w_{12}^{(1)}} = \frac{\partial \mathcal{L}}{\partial y^*}\frac{\partial y^*}{\partial z_2}\mathcal{A}'\left(w_0 x_1 + w_0 x_2\right)x_1$$

- Since $z_1 = z_2$, we have $\dfrac{\partial \mathcal{L}}{\partial w_{11}^{(1)}} = \dfrac{\partial \mathcal{L}}{\partial w_{12}^{(1)}}$.
- Similarly can show that $\dfrac{\partial \mathcal{L}}{\partial w_{22}^{(1)}} = \dfrac{\partial \mathcal{L}}{\partial w_{21}^{(1)}}$.
- This is known as the symmetry breaking problem.

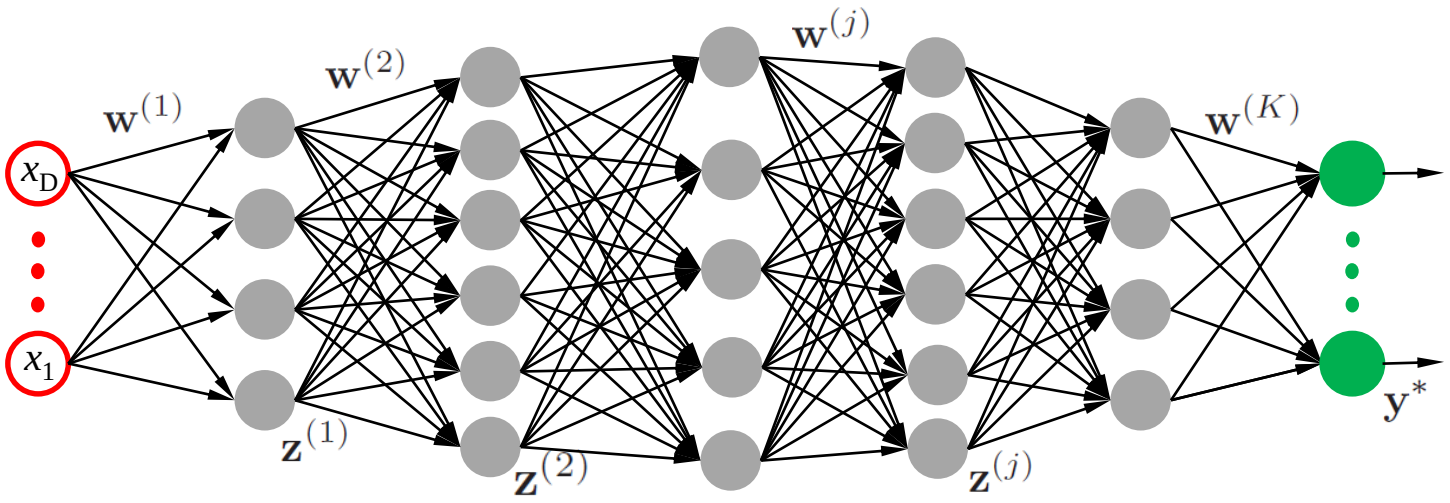
Large/small initialization of weights



$$\frac{\partial \mathcal{L}}{\partial \mathbf{w}^{(h)}} = \frac{\partial \mathcal{L}}{\partial \mathbf{y}^*} \frac{\partial \mathbf{y}^*}{\partial \mathbf{z}^{(K-1)}} \frac{\partial \mathbf{z}^{(K-1)}}{\partial \mathbf{z}^{(K-2)}} \frac{\partial \mathbf{z}^{(K-2)}}{\partial \mathbf{z}^{(K-3)}} \cdots \frac{\partial \mathbf{z}^{(h+1)}}{\partial \mathbf{z}^{(h)}} \frac{\partial \mathbf{z}^{(h)}}{\partial \mathbf{w}^{(h)}}$$

- Effect of the weight matrices on the chain product:
  - Expansion in directions where the singular values of the weight matrices are greater than one.
  - Shrink along directions where the singular values of the weight matrices are less than one.
- Multiplications by the weight matrices can lead to **exploding** or **vanishing** gradients.

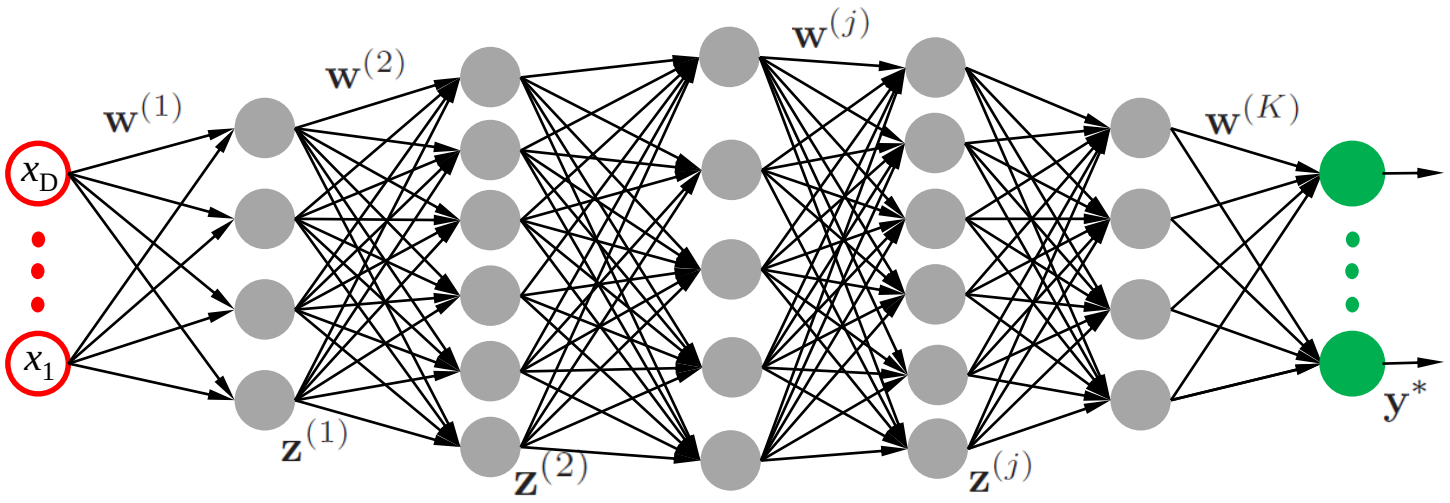
Xavier's approach



- Assumptions:
  - The mean of the activations should be zero.
  - The variance of the activations should be the same across every layer.
  - Weights are initialized with zero mean.
  - All weights across a layer are drawn from a same variance distribution.
- Assume that we are using the tanh activation function, i.e. $\mathcal{A}(\cdot) = \tanh(\cdot)$.

*Source paper: X. Glorot and Y. Bengio, *Understanding the difficulty of training deep feedforward neural networks*, AISTATS, 2010.

tanh activation



- Assume that our inputs are normalized, and the weights are initialized with small values.
  - This implies that in the beginning we are in the linear regime of the tanh, i.e. $\tanh(x) \approx x$.
- Output from layer k:

$$\begin{aligned}\mathbf{z}^{(k)} &= \mathcal{A}\left(\mathbf{w}^{(k)\mathrm{T}}\mathbf{z}^{(k-1)}\right) \\ &\approx \mathbf{w}^{(k)\mathrm{T}}\mathbf{z}^{(k-1)}\end{aligned}$$

Activation variance

- Variance of the output from the jth unit of the kth layer:

$$\mathrm{Var}\left(z_j^{(k)}\right) = \mathrm{Var}\left(\sum_{i=1}^{H_{k-1}} w_{ij}^{(k)} z_i^{(k-1)}\right)$$

- Under the following assumptions:
  - inputs are independent and identically distributed
  - weights are independent and identically distributed
  - inputs and weights are mutually independent

  we have

$$\begin{aligned}\mathrm{Var}\left(z_j^{(k)}\right) &= \mathrm{Var}\left(\sum_{i=1}^{H_{k-1}} w_{ij}^{(k)} z_i^{(k-1)}\right) \\ &= \sum_{i=1}^{H_{k-1}} \mathrm{Var}\left(w_{ij}^{(k)} z_i^{(k-1)}\right)\end{aligned}$$

Activation variance

- We now make use of the formula:

$$\mathrm{Var}(AB) = \mathbb{E}[A]^2\mathrm{Var}(B) + \mathrm{Var}(A)\mathbb{E}[B]^2 + \mathrm{Var}(A)\mathrm{Var}(B)$$

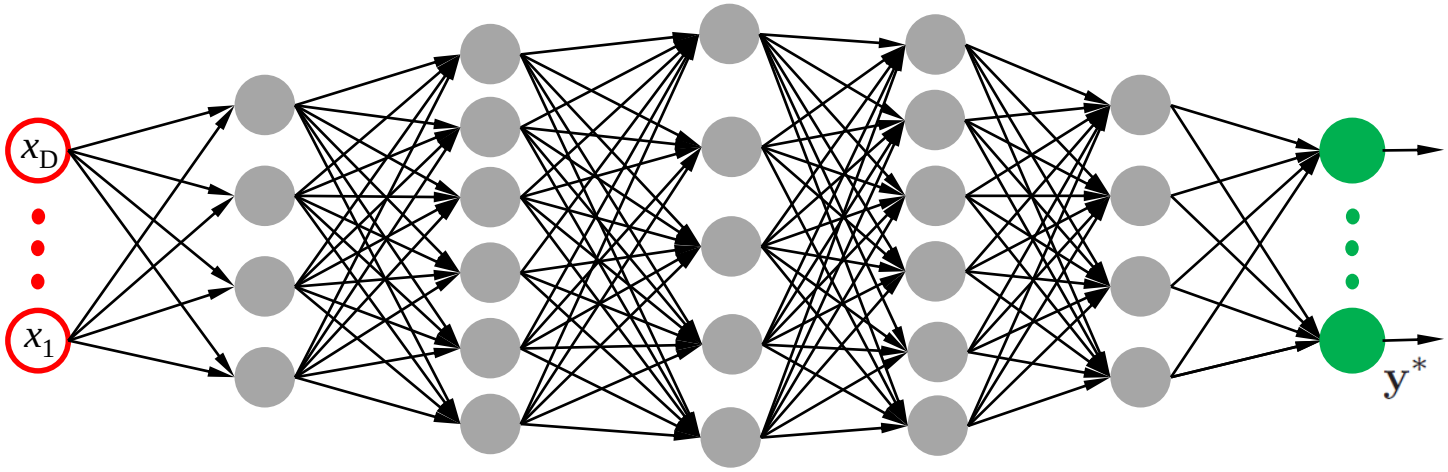
with $A = w_{ij}^{(k)}$ and $B = z_i^{(k-1)}$, then we have

$$\begin{aligned}\mathrm{Var}\left(z_j^{(k)}\right) &= \sum_{i=1}^{H_{k-1}} \mathrm{Var}\left(w_{ij}^{(k)} z_i^{(k-1)}\right) \\ &= \sum_{i=1}^{H_{k-1}} \left(\mathbb{E}\left[w_{ij}^{(k)}\right]^2 \mathrm{Var}\left(z_i^{(k-1)}\right) + \mathrm{Var}\left(w_{ij}^{(k)}\right) \mathbb{E}\left[z_i^{(k-1)}\right]^2 \right. \\ &\qquad \left. + \mathrm{Var}\left(w_{ij}^{(k)}\right) \mathrm{Var}\left(z_i^{(k-1)}\right) \right)\end{aligned}$$

- From our assumptions we have $\mathbb{E}\left[w_{ij}^{(k)}\right] = 0$ and $\mathbb{E}\left[z_i^{(k-1)}\right] = 0$, and so

$$\mathrm{Var}\left(z_j^{(k)}\right) = \sum_{i=1}^{H_{k-1}} \mathrm{Var}\left(w_{ij}^{(k)}\right) \mathrm{Var}\left(z_i^{(k-1)}\right)$$

….Assumptions



- Again from our assumptions we have

$$\mathrm{Var}\left(w_{11}^{(k)}\right) = \mathrm{Var}\left(w_{12}^{(k)}\right) = \cdots = \mathrm{Var}\left(\mathbf{w}^{(k)}\right)$$

where $\mathrm{Var}\left(\mathbf{w}^{(k)}\right)$ indicates the variance of an entry of $\mathbf{w}^{(k)}$ which are all same. Similarly

$$\mathrm{Var}\left(z_1^{(k-1)}\right) = \mathrm{Var}\left(z_2^{(k-1)}\right) = \cdots = \mathrm{Var}\left(\mathbf{z}^{(k-1)}\right)$$

Variance at the output layer

- Eventually we have

$$\mathrm{Var}\left(\mathbf{z}^{(k)}\right) = H_{k-1}\mathrm{Var}\left(\mathbf{w}^{(k)}\right)\mathrm{Var}\left(\mathbf{z}^{(k-1)}\right)$$

- The variance at the output layer can be expressed as

$$\begin{aligned}
\mathrm{Var}\left(\mathbf{y}^{*}\right) &= H_{K-1}\mathrm{Var}\left(\mathbf{w}^{(K)}\right)\mathrm{Var}\left(\mathbf{z}^{(K-1)}\right) \\
&= H_{K-1}\mathrm{Var}\left(\mathbf{w}^{(K)}\right)H_{k-2}\mathrm{Var}\left(\mathbf{w}^{(K-1)}\mathrm{Var}\left(\mathbf{z}^{(K-2)}\right)\right. \\
&\cdot \\
&\cdot \\
&= \left(\prod_{k=1}^{K} H_{k-1}\mathrm{Var}\left(\mathbf{w}^{(k)}\right)\right)\mathrm{Var}(\mathbf{x})
\end{aligned}$$

Variance of weights (initialization)

- We have the following three cases:

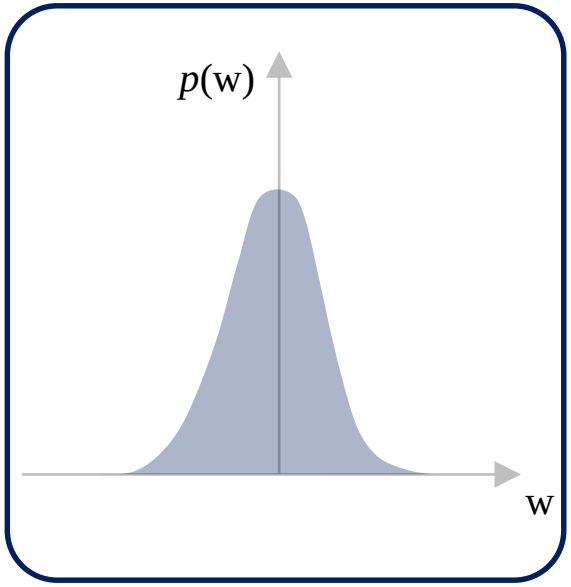
$$H_k \mathrm{Var}\left(\mathbf{w}^{(k)}\right) \begin{cases} < 1 & \text{Vanishing signal} \\ = 1 & \mathrm{Var}(\mathbf{y}^*) = \mathrm{Var}(\mathbf{x}) \\ > 1 & \text{Exploding signal} \end{cases}$$

- Therefore to avoid vanishing or exploding of the forward propagated signal, we must have

$$H_{k-1} \mathrm{Var}\left(\mathbf{w}^{(k)}\right) = 1$$

$$\mathrm{Var}\left(\mathbf{w}^{(k)}\right) = \frac{1}{H_{k-1}}$$

Weights initialization



- In practice, weights at a particular layer (say k) are initialized by randomly sampling from $\mathcal{N}\left(0, \frac{1}{H_{k-1}}\right)$ or $\mathcal{N}\left(0, \frac{2}{H_{k-1}+H_k}\right)$.
- Note that derived initialization is applicable for tanh activation.
- ReLU activation: He initialization (He *et. al.*, Delving deep into Rectifiers, ICCV 2015).