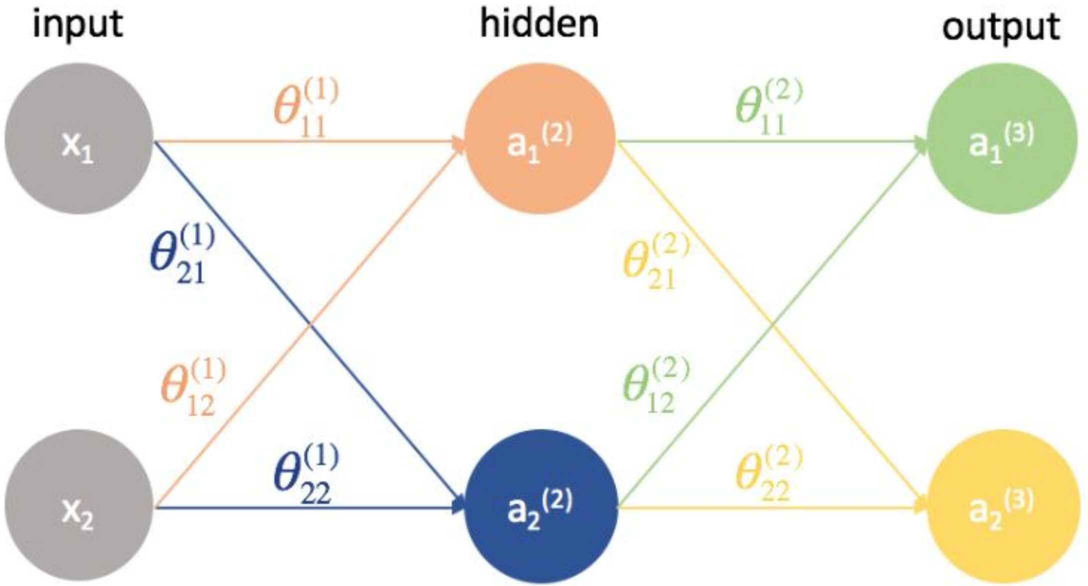
This is CS5045!

GCR:ioc7cdl

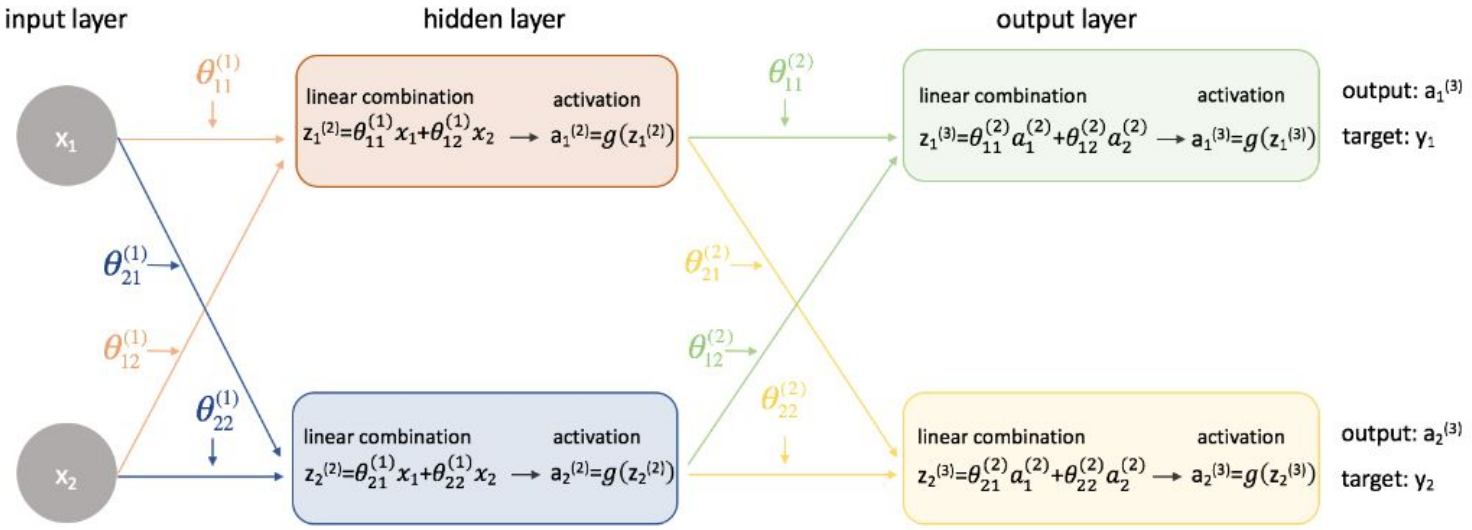
P.S. These slides are useless if you do not attend classes

Neural Networks

Neural network with multi Output



Neural network with multi Output

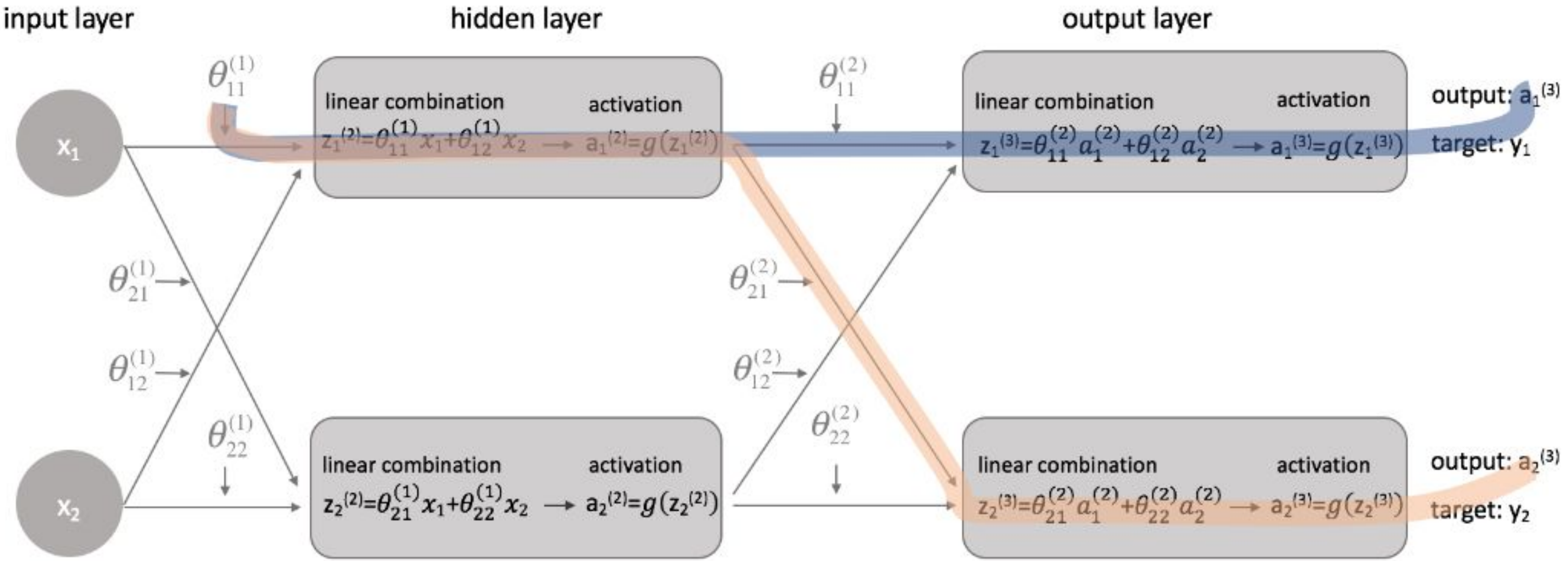


Neural network with multi Output

Loss calculation

$$J(\theta) = \frac{1}{2m} \sum \left(y_i - \mathrm{a}_i^{(2)} \right)^2$$

Neural network with multi Output



The derivative chain for the blue path is:

$$\left(\frac{\partial J(\theta)}{\partial \mathrm{a}_1^{(3)}}\right)\left(\frac{\partial \mathrm{a}_1^{(3)}}{\partial z_1^{(3)}}\right)\left(\frac{\partial z_1^{(3)}}{\partial \mathrm{a}_1^{(2)}}\right)\left(\frac{\partial \mathrm{a}_1^{(2)}}{\partial z_1^{(2)}}\right)\left(\frac{\partial z_1^{(2)}}{\partial \theta_{11}^{(1)}}\right)$$

The derivative chain for the orange path is:

$$\left(\frac{\partial J(\theta)}{\partial \mathrm{a}_2^{(3)}}\right)\left(\frac{\partial \mathrm{a}_2^{(3)}}{\partial z_2^{(3)}}\right)\left(\frac{\partial z_2^{(3)}}{\partial \mathrm{a}_1^{(2)}}\right)\left(\frac{\partial \mathrm{a}_1^{(2)}}{\partial z_1^{(2)}}\right)\left(\frac{\partial z_1^{(2)}}{\partial \theta_{11}^{(1)}}\right)$$

Combining these, we get the total expression for $\frac{\partial J(\theta)}{\partial \theta_{11}^{(1)}}$.

$$\frac{\partial J(\theta)}{\partial \theta_{11}^{(1)}}=\left(\frac{\partial J(\theta)}{\partial \mathrm{a}_1^{(3)}}\right)\left(\frac{\partial \mathrm{a}_1^{(3)}}{\partial z_1^{(3)}}\right)\left(\frac{\partial z_1^{(3)}}{\partial \mathrm{a}_1^{(2)}}\right)\left(\frac{\partial \mathrm{a}_1^{(2)}}{\partial z_1^{(2)}}\right)\left(\frac{\partial z_1^{(2)}}{\partial \theta_{11}^{(1)}}\right)+\left(\frac{\partial J(\theta)}{\partial \mathrm{a}_2^{(3)}}\right)\left(\frac{\partial \mathrm{a}_2^{(3)}}{\partial z_2^{(3)}}\right)\left(\frac{\partial z_2^{(3)}}{\partial \mathrm{a}_1^{(2)}}\right)\left(\frac{\partial \mathrm{a}_1^{(2)}}{\partial z_1^{(2)}}\right)\left(\frac{\partial z_1^{(2)}}{\partial \theta_{11}^{(1)}}\right)$$

Layer 2 Parameters

$$\frac{\partial J(\theta)}{\partial \theta_{11}^{(2)}} = \left(\frac{\partial J(\theta)}{\partial \mathrm{a}_1^{(3)}}\right)\left(\frac{\partial \mathrm{a}_1^{(3)}}{\partial z_1^{(3)}}\right)\left(\frac{\partial z_1^{(3)}}{\partial \theta_{11}^{(2)}}\right)$$

$$\frac{\partial J(\theta)}{\partial \theta_{12}^{(2)}} = \left(\frac{\partial J(\theta)}{\partial \mathrm{a}_1^{(3)}}\right)\left(\frac{\partial \mathrm{a}_1^{(3)}}{\partial z_1^{(3)}}\right)\left(\frac{\partial z_1^{(3)}}{\partial \theta_{12}^{(2)}}\right)$$

$$\frac{\partial J(\theta)}{\partial \theta_{21}^{(2)}} = \left(\frac{\partial J(\theta)}{\partial \mathrm{a}_2^{(3)}}\right)\left(\frac{\partial \mathrm{a}_2^{(3)}}{\partial z_2^{(3)}}\right)\left(\frac{\partial z_2^{(3)}}{\partial \theta_{21}^{(2)}}\right)$$

$$\frac{\partial J(\theta)}{\partial \theta_{22}^{(2)}} = \left(\frac{\partial J(\theta)}{\partial \mathrm{a}_2^{(3)}}\right)\left(\frac{\partial \mathrm{a}_2^{(3)}}{\partial z_2^{(3)}}\right)\left(\frac{\partial z_2^{(3)}}{\partial \theta_{22}^{(2)}}\right)$$

Layer 1 Parameters

$$\frac{\partial J(\theta)}{\partial \theta_{11}^{(1)}} = \left(\frac{\partial J(\theta)}{\partial \mathrm{a}_1^{(3)}}\right)\left(\frac{\partial \mathrm{a}_1^{(3)}}{\partial z_1^{(3)}}\right)\left(\frac{\partial z_1^{(3)}}{\partial \mathrm{a}_1^{(2)}}\right)\left(\frac{\partial \mathrm{a}_1^{(2)}}{\partial z_1^{(2)}}\right)\left(\frac{\partial z_1^{(2)}}{\partial \theta_{11}^{(1)}}\right) + \left(\frac{\partial J(\theta)}{\partial \mathrm{a}_2^{(3)}}\right)\left(\frac{\partial \mathrm{a}_2^{(3)}}{\partial z_2^{(3)}}\right)\left(\frac{\partial z_2^{(3)}}{\partial \mathrm{a}_1^{(2)}}\right)\left(\frac{\partial \mathrm{a}_1^{(2)}}{\partial z_1^{(2)}}\right)\left(\frac{\partial z_1^{(2)}}{\partial \theta_{11}^{(1)}}\right)$$

$$\frac{\partial J(\theta)}{\partial \theta_{12}^{(1)}} = \left(\frac{\partial J(\theta)}{\partial \mathrm{a}_1^{(3)}}\right)\left(\frac{\partial \mathrm{a}_1^{(3)}}{\partial z_1^{(3)}}\right)\left(\frac{\partial z_1^{(3)}}{\partial \mathrm{a}_1^{(2)}}\right)\left(\frac{\partial \mathrm{a}_1^{(2)}}{\partial z_1^{(2)}}\right)\left(\frac{\partial z_1^{(2)}}{\partial \theta_{12}^{(1)}}\right) + \left(\frac{\partial J(\theta)}{\partial \mathrm{a}_2^{(3)}}\right)\left(\frac{\partial \mathrm{a}_2^{(3)}}{\partial z_2^{(3)}}\right)\left(\frac{\partial z_2^{(3)}}{\partial \mathrm{a}_1^{(2)}}\right)\left(\frac{\partial \mathrm{a}_1^{(2)}}{\partial z_1^{(2)}}\right)\left(\frac{\partial z_1^{(2)}}{\partial \theta_{12}^{(1)}}\right)$$

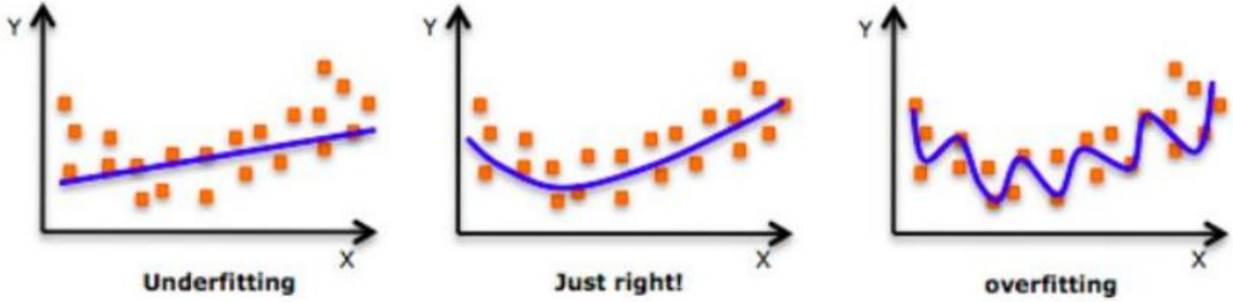
$$\frac{\partial J(\theta)}{\partial \theta_{21}^{(1)}} = \left(\frac{\partial J(\theta)}{\partial \mathrm{a}_1^{(3)}}\right)\left(\frac{\partial \mathrm{a}_1^{(3)}}{\partial z_1^{(3)}}\right)\left(\frac{\partial z_1^{(3)}}{\partial \mathrm{a}_2^{(2)}}\right)\left(\frac{\partial \mathrm{a}_2^{(2)}}{\partial z_2^{(2)}}\right)\left(\frac{\partial z_2^{(2)}}{\partial \theta_{21}^{(1)}}\right) + \left(\frac{\partial J(\theta)}{\partial \mathrm{a}_2^{(3)}}\right)\left(\frac{\partial \mathrm{a}_2^{(3)}}{\partial z_2^{(3)}}\right)\left(\frac{\partial z_2^{(3)}}{\partial \mathrm{a}_2^{(2)}}\right)\left(\frac{\partial \mathrm{a}_2^{(2)}}{\partial z_2^{(2)}}\right)\left(\frac{\partial z_2^{(2)}}{\partial \theta_{21}^{(1)}}\right)$$

$$\frac{\partial J(\theta)}{\partial \theta_{22}^{(1)}} = \left(\frac{\partial J(\theta)}{\partial \mathrm{a}_1^{(3)}}\right)\left(\frac{\partial \mathrm{a}_1^{(3)}}{\partial z_1^{(3)}}\right)\left(\frac{\partial z_1^{(3)}}{\partial \mathrm{a}_2^{(2)}}\right)\left(\frac{\partial \mathrm{a}_2^{(2)}}{\partial z_2^{(2)}}\right)\left(\frac{\partial z_2^{(2)}}{\partial \theta_{22}^{(1)}}\right) + \left(\frac{\partial J(\theta)}{\partial \mathrm{a}_2^{(3)}}\right)\left(\frac{\partial \mathrm{a}_2^{(3)}}{\partial z_2^{(3)}}\right)\left(\frac{\partial z_2^{(3)}}{\partial \mathrm{a}_2^{(2)}}\right)\left(\frac{\partial \mathrm{a}_2^{(2)}}{\partial z_2^{(2)}}\right)\left(\frac{\partial z_2^{(2)}}{\partial \theta_{22}^{(1)}}\right)$$

Discussion on Assignment Network

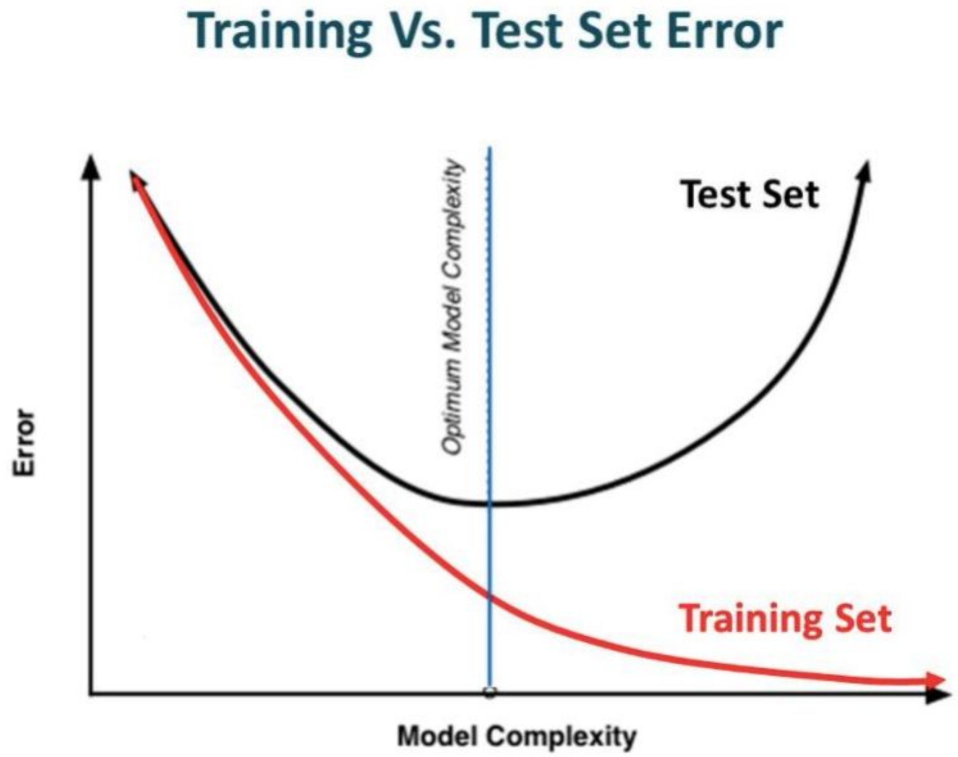
Regularization

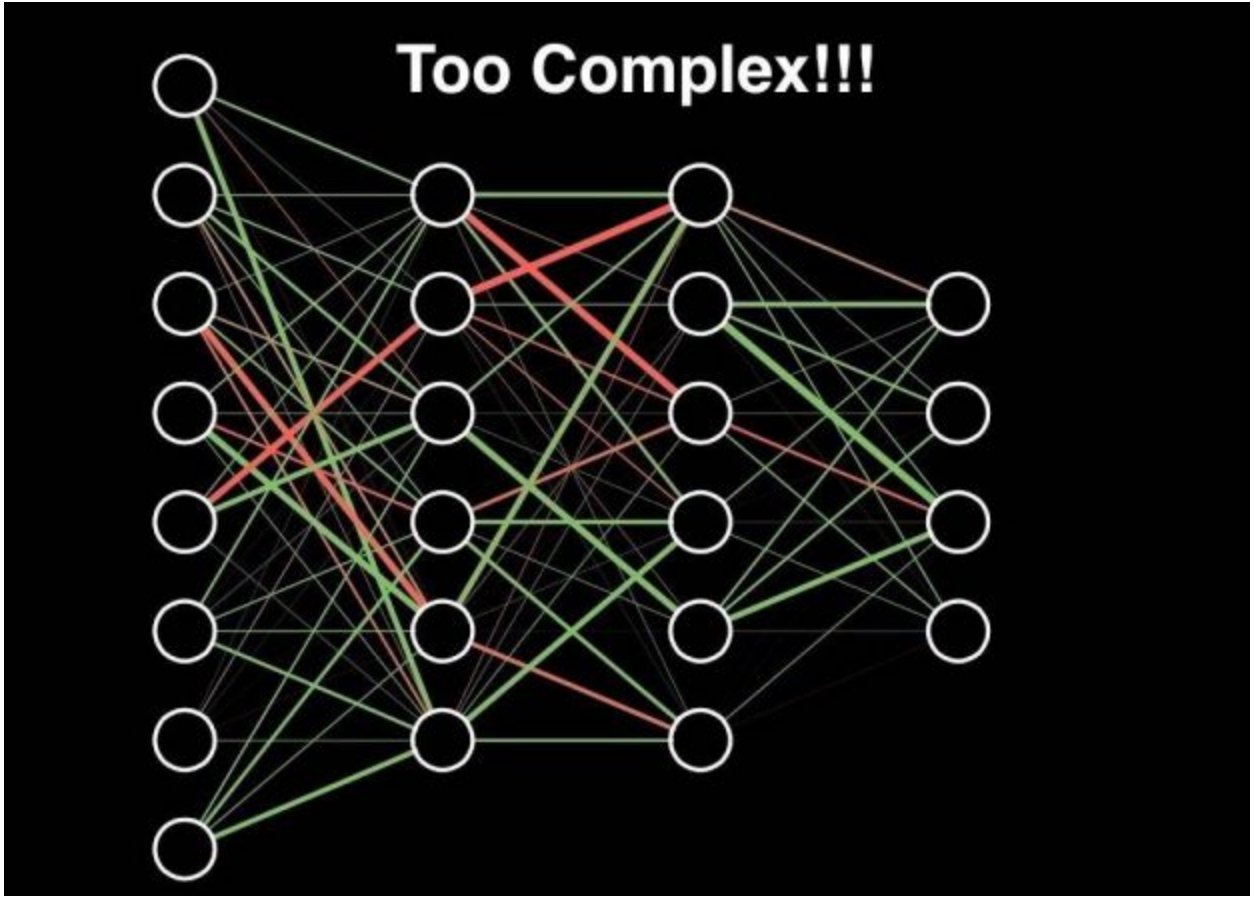
Regularization



As move towards right, poor performance on unseen data

Regularization



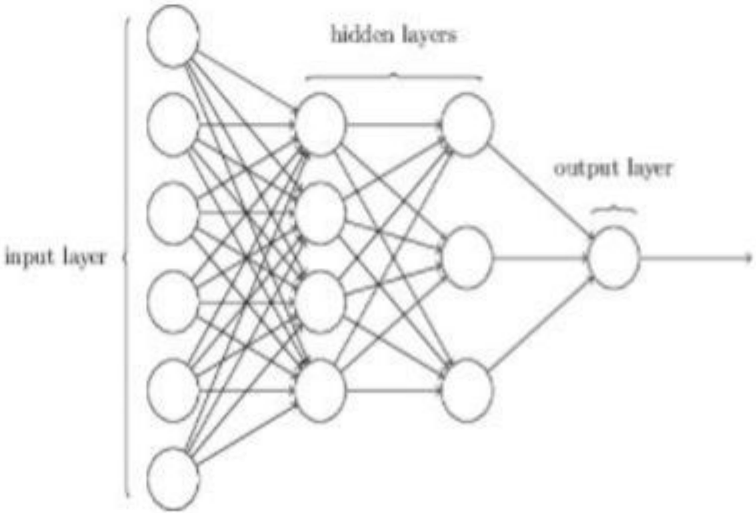

Too Complex!!!

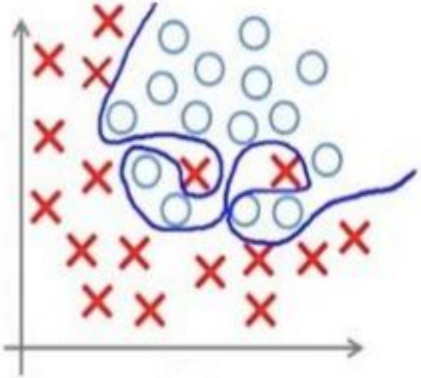
What is Regularization ?

Regularization is a technique which makes slight modifications to the learning algorithm such that the model generalizes better

This in turn improves the model’s performance on the unseen data as well

Regularization





Over-fitting

Regularization

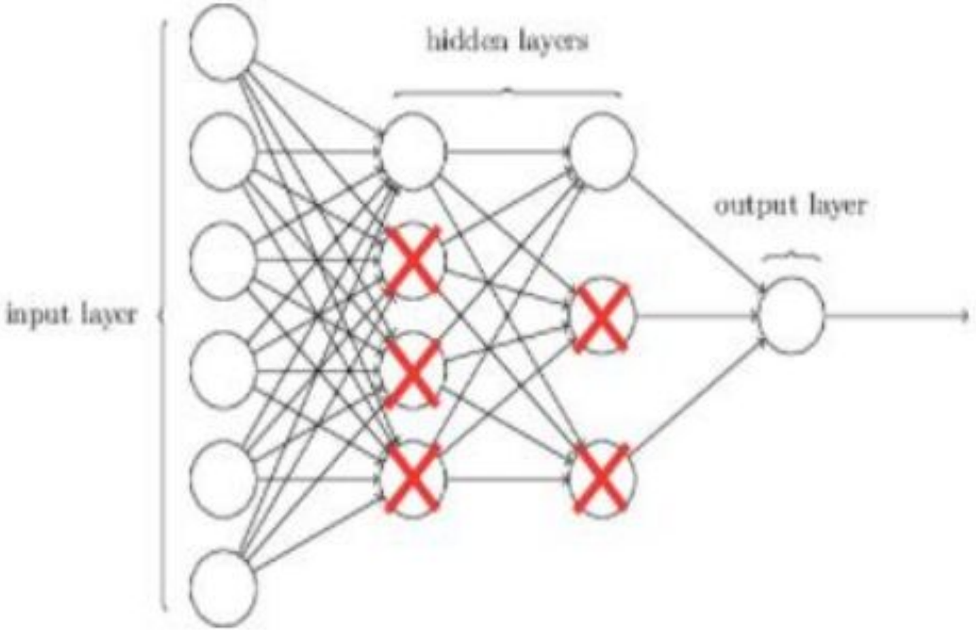
In machine learning, regularization penalizes the coefficients

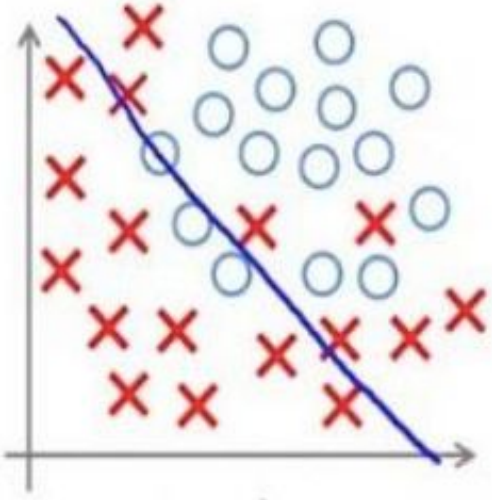
In deep learning, it actually penalizes the weight matrices of the nodes

Assume that our regularization coefficient is so high that some of the weight matrices are nearly equal to zero

This will result in a much simpler linear network and slight underfitting of the training data.

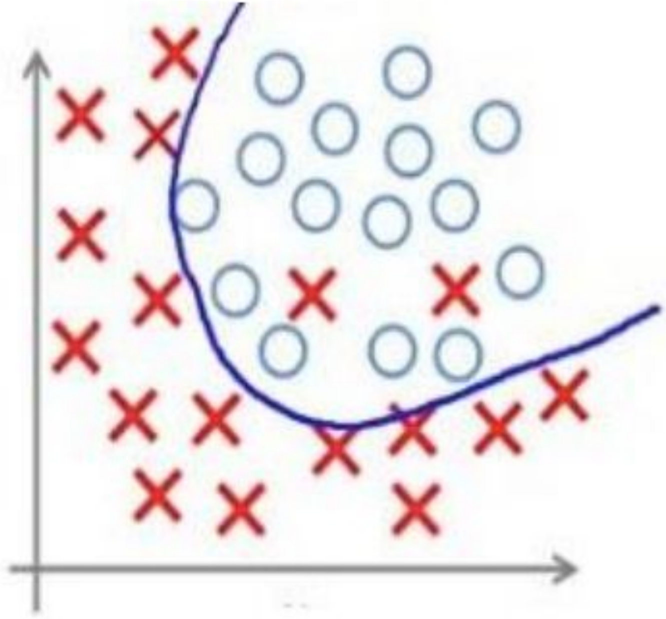
Regularization





Under-fitting

Regularization



Appropriate-fitting

Such a large value of the regularization coefficient is not that useful

We need to optimize the value of regularization coefficient in order to obtain a well-fitted model as shown in the image below

Regularization

- L1 and L2 regularization
- DropOut
- Data Augmentation
- Early Stopping

Reading Assignment

Read Neural Network L2 Regularization Using Python -- Visual Studio Magazine.pdf

Read Neural Network L1 Regularization Using Python -- Visual Studio Magazine.pdf

L1 L2 Regularization

L1 Regularization

$$\text{Modified loss function} = \text{Loss function} + \lambda \sum_{i=1}^{n} |W_i|$$

L2 Regularization

$$\text{Modified loss function} = \text{Loss function} + \lambda \sum_{i=1}^{n} W_i^{2}$$

L1

$$E = \underbrace{\frac{1}{2} * \sum (t_k - o_k)^2}_{\text{squared error}} + \underbrace{\lambda * \sum |w_i|}_{\text{L1 weight penalty}}$$

$$\Delta w_{jk} = -1 * \underbrace{\eta}_{\text{learning rate}} * \overbrace{[\, x_j * \underbrace{(o_k - t_k) * o_k * (1 - o_k) \,] \pm \lambda \,]}_{\text{signal}}}^{\frac{\partial E}{\partial w_{jk}} \ \text{gradient}}$$

$$w_{jk} = w_{jk} + \Delta w_{jk}$$

L2

$$E = \underbrace{\frac{1}{2} * \sum (t_k - o_k)^2}_{\text{plain error}} + \underbrace{\frac{\lambda}{2} * \sum {w_i}^2}_{\text{weight penalty}}$$

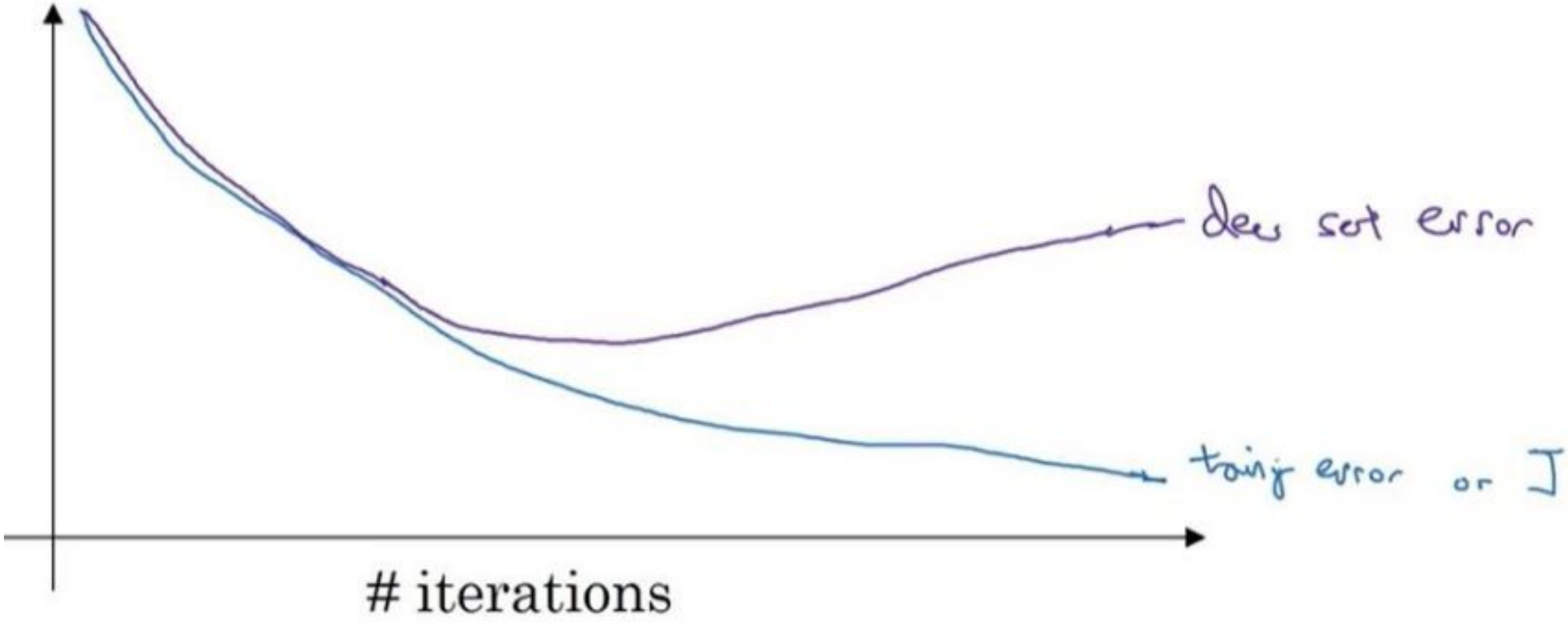
elegant math

simple math

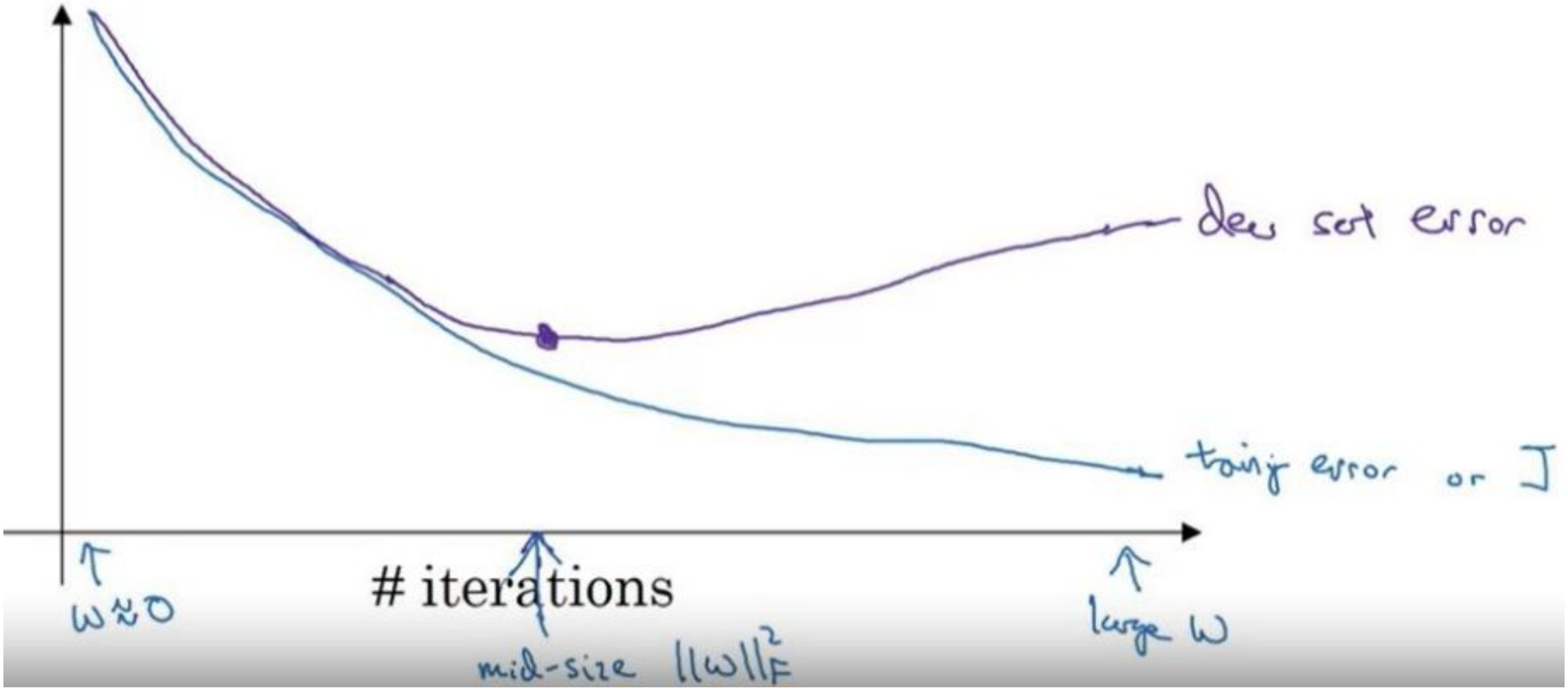
$$\Delta w_{jk} = \underbrace{\eta}_{\text{learning rate}} * \overbrace{\left[x_j * \underbrace{(o_k - t_k) * o_k * (1 - o_k) \right] + [\lambda * w_{jk}]}_{\text{signal}}}^{\frac{\partial E}{\partial w_{jk}} \ \text{gradient}}$$

Data Augmentation

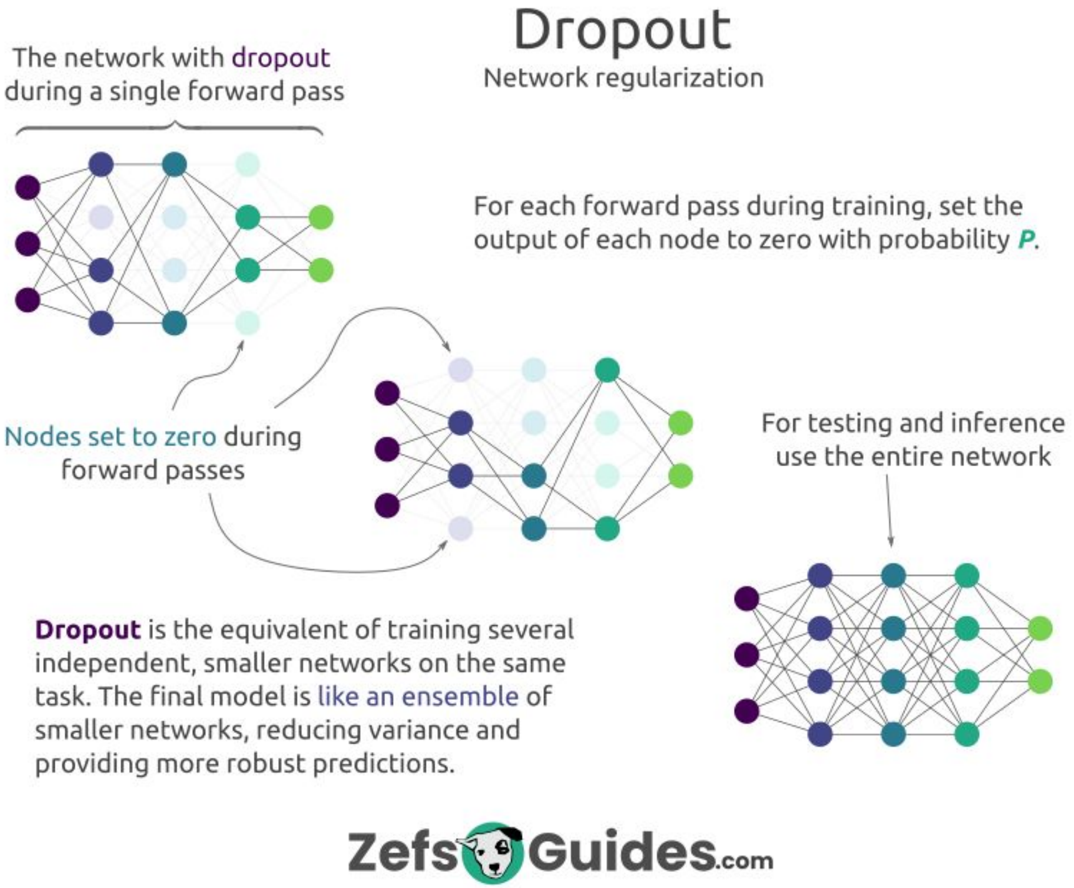
Early Stopping



Early Stopping



Drop Out


Dropout
Network regularization
The network with dropout during a single forward pass
For each forward pass during training, set the output of each node to zero with probability *P*.
Nodes set to zero during forward passes
For testing and inference use the entire network
Dropout is the equivalent of training several independent, smaller networks on the same task. The final model is like an ensemble of smaller networks, reducing variance and providing more robust predictions.
ZefsGuides.com

References

https://www.jeremyjordan.me/neural-networks-training/

https://www.reddit.com/r/learnmachinelearning/comments/x89qsi/dropout_in_neural_networks_what_it_is_and_how_it/?rdt=55837

https://www.analyticsvidhya.com/blog/2018/04/fundamentals-deep-learning-regularization-techniques/