

What are Hyperparameters ? and How to tune the Hyperparameters in a Deep Neural Network?



Pranoy Radhakrishnan · Follow

Published in Towards Data Science · 3 min read · Aug 10, 2017



1.1K



3



What are **hyperparameters**?

Hyperparameters are the **variables** which **determines** the **network**

Open in app ↗

Sign up

Sign in

Medium



Search



Write



Hyperparameters are **set before training**(before optimizing the weights and bias).

Hyperparameters related to Network structure

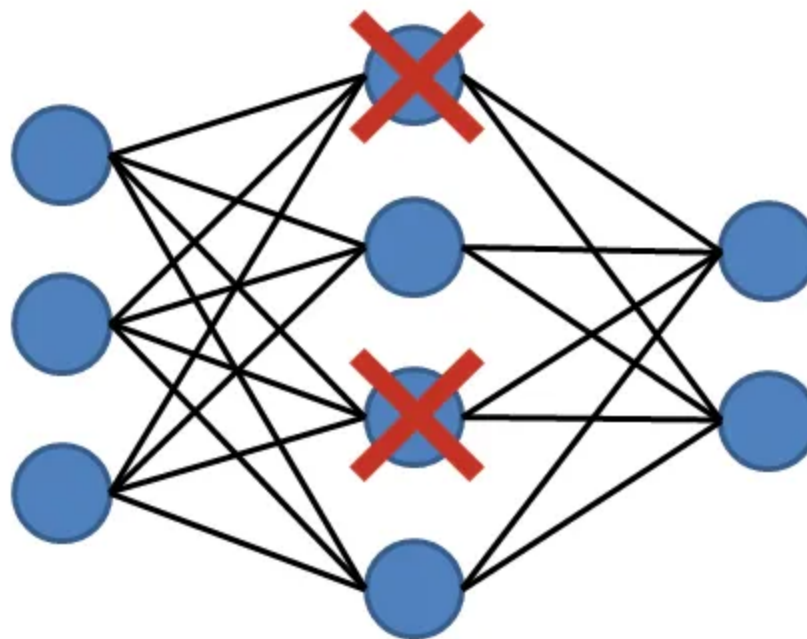
Number of Hidden Layers and units

Hidden layers are the layers between input layer and output layer.

“Very simple. Just keep adding layers until the test error does not improve anymore.”

Many hidden units within a layer with regularization techniques can increase accuracy. Smaller number of units may cause **underfitting**.

Dropout



Random neurons are cancelled

Dropout is regularization technique to avoid overfitting (increase the validation accuracy) thus increasing the generalizing power.

- Generally, use a small dropout value of 20%-50% of neurons with 20% providing a good starting point. A probability too low has minimal effect and a value too high results in under-learning by the network.

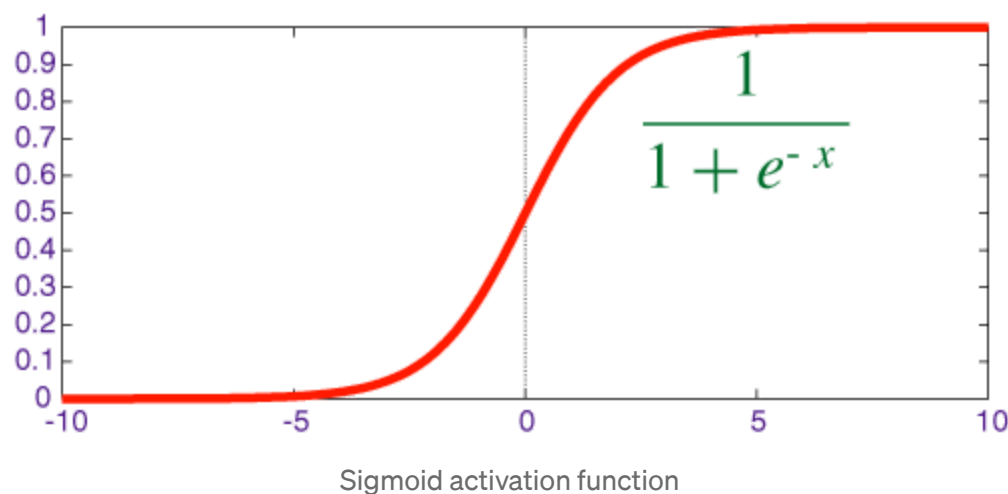
- Use a larger network. You are likely to get better performance when dropout is used on a larger network, giving the model more of an opportunity to learn independent representations.

Network Weight Initialization

Ideally, it may be better to use different weight initialization schemes according to the activation function used on each layer.

Mostly **uniform distribution** is used.

Activation function



Activation functions are used to **introduce nonlinearity** to models, which allows deep learning models to learn nonlinear prediction boundaries.

Generally, the **rectifier activation function** is the most popular.

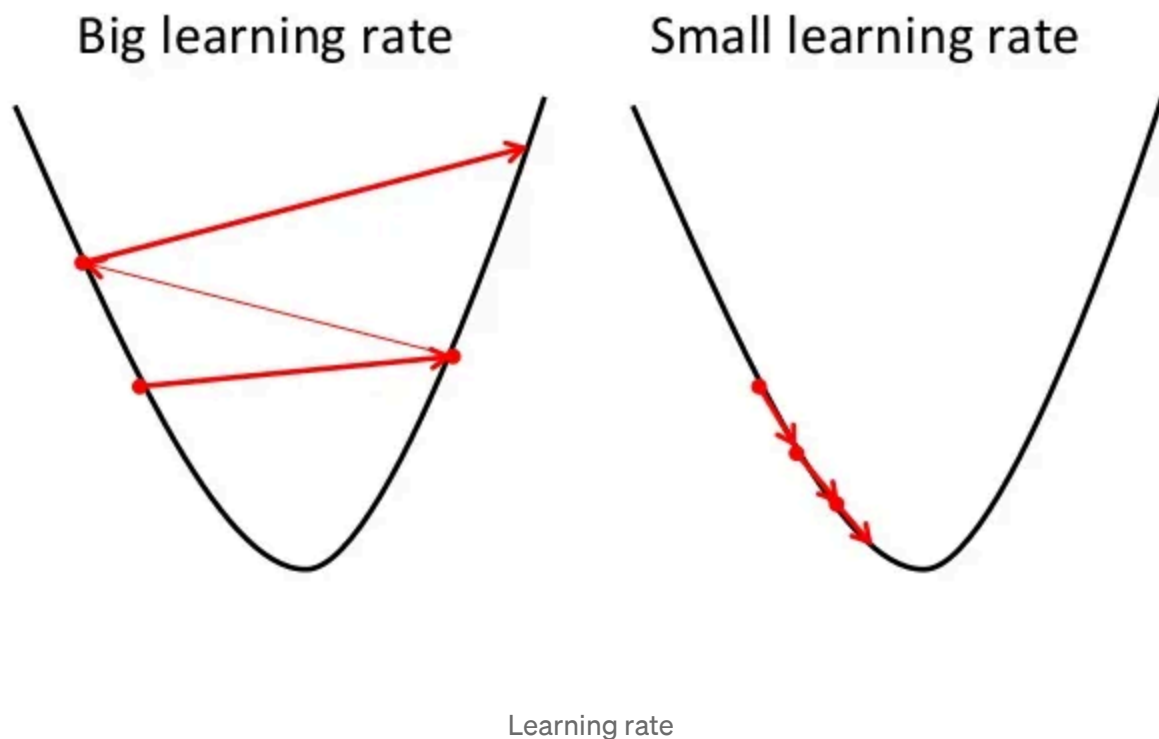
Sigmoid is used in the output layer while making **binary predictions**.

Softmax is used in the output layer while making **multi-class predictions**.

Hyperparameters related to Training Algorithm

Learning Rate

Gradient Descent



The learning rate defines how quickly a network updates its parameters.

Low learning rate slows down the learning process but converges smoothly.

Larger learning rate speeds up the learning but may not converge.

Usually a **decaying Learning rate** is preferred.

Momentum

Momentum helps to know the direction of the next step with the knowledge of the previous steps. It helps to prevent oscillations. A typical choice of momentum is between 0.5 to 0.9.

Number of epochs

Number of epochs is the number of times the whole training data is shown to the network while training.

Increase the number of epochs until the validation accuracy starts decreasing even when training accuracy is increasing(overfitting).

Batch size

Mini batch size is the number of sub samples given to the network after which parameter update happens.

A good default for batch size might be 32. Also try 32, 64, 128, 256, and so on.

Methods used to find out Hyperparameters

1. *Manual Search*
2. *Grid Search* (<http://machinelearningmastery.com/grid-search-hyperparameters-deep-learning-models-python-keras/>)
3. *Random Search*
4. *Bayesian Optimization*

[Machine Learning](#)[Deep Lear](#)[Artificial Intelligence](#)[Hyperparameters](#)

Published in Towards Data Science

790K Followers · Last published 4 hours ago

[Follow](#)