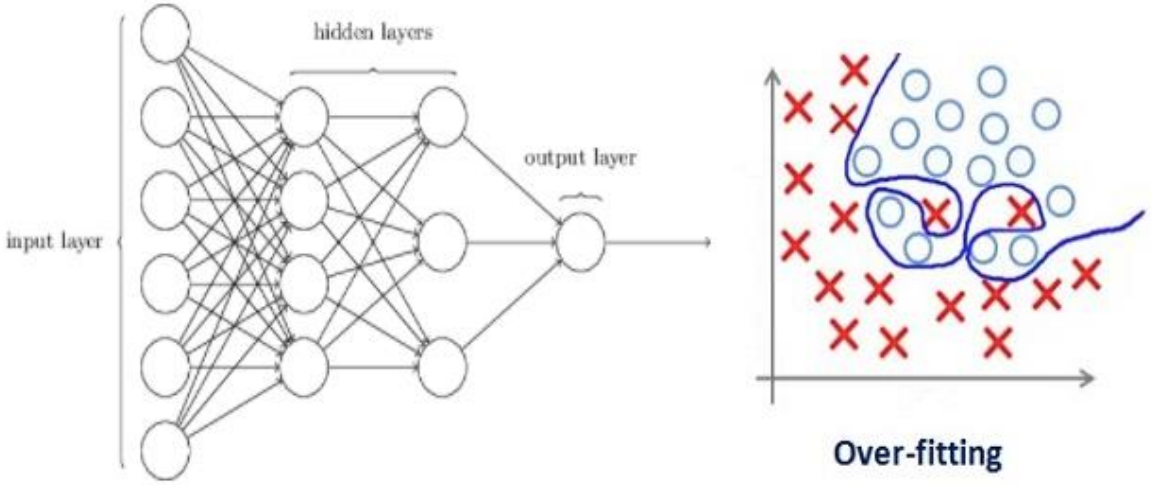
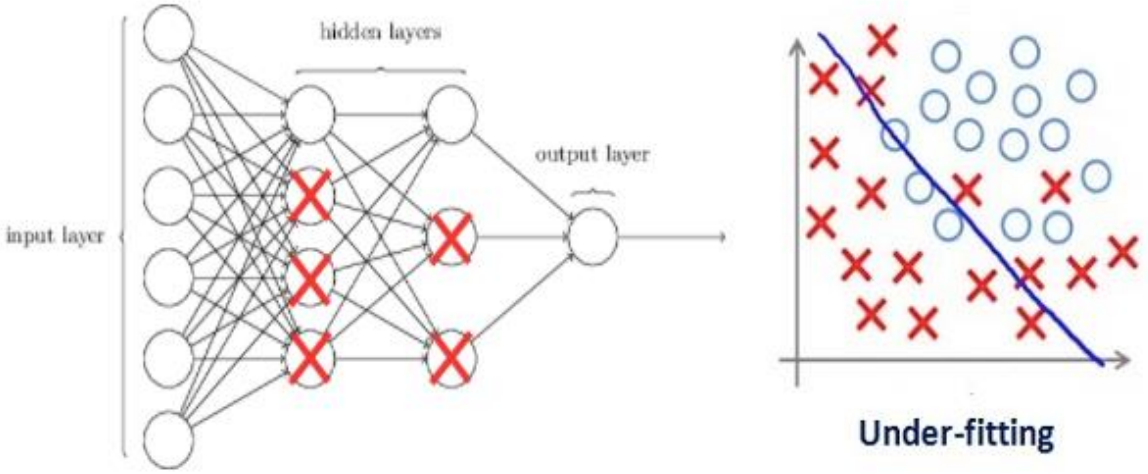
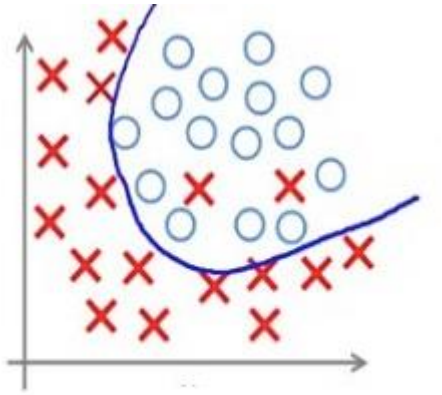
Regularization: As we all know if our model learns too much then overfit will occur. It is bad for the model because our model won't perform well outside of the training data.



Underfitting: When a model learns too less when our data is less or, our when our model is linear.



To get around this issue we use regularization. After using regularization, we will get a better model which will produce better and correct outcome.



Appropriate-fitting

L1 & L2 Regularization:

In L1, we penalize absolute value of the weights. Weights may have reduced to zero here. It is useful when we want to compress our model.

$$Cost\,function = Loss + \frac{\lambda}{2m} * \sum ||w||$$

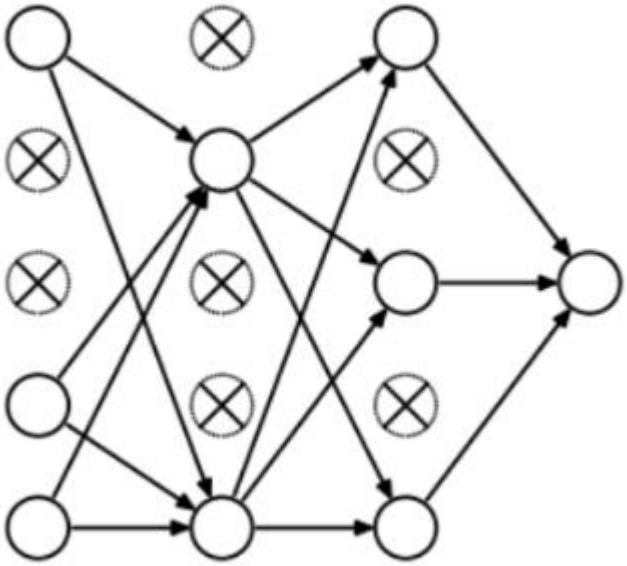
In L2 formula,

$$Cost\,function = Loss + \frac{\lambda}{2m} * \sum ||w||^2$$

Lambda is the regularization parameter which is hyperparameter. Its value is optimized to get better result. L2

regularization forces the weights to go near to zero. But not exactly zero.

Dropout: In deep learning dropout is very sophisticated. It gives better outcome than others. In every iteration it randomly selects some nodes and discards them with their connections. It is kind of ensemble technique because of the randomness it uses to dropout neurons.



Also need to mention, dropout number is a hyperparameter. User has complete control over it.