- The validation sets and the test sets must belong to the same distribution. If they belong to different distribution then choosing the best accuracy algorithm is not possible.
- Highly correlated features should be removed from dataset to reduce redundancy.
- Due to large datasets, the % split of validation and test set have reduced (keeping the number same though).
- Asssuming ideal % error is 1
    - if train error = 15% –> **high bias**
    - if train error = 1%, val error = 15% –> **high variance**
    - if train error = 15%, val error = 30% –> **high bias and variance**
- Underfitted model has high bias, overfitted model has high variance
- High bias and high variance corresponds to models which underfit at some points and overfit at some.
- Low bias and low variance are the desirable charcateristics of ideal model

High bias (Underfitting):

- Add more layers (make the model complex)
- Train longer

High variance (Overfitting):

- Add more data
- Use regularisation

L1 regularisation for logistic regression:

- Cost function changes to $$ J(w,b)=-1/n \sum_{i=0}^n (y~log(y') + (1-y)log(1-y')) + lambda/2m * ||w|| $$
- Leads to sparsity in w

L2 regularisation for logistic regression:

- Cost function changes to $$ J(w,b)=-1/n \sum_{i=0}^n (y~log(y') + (1-y)log(1-y')) + lambda/2m * ||w||^2 $$
- Performs better than L1
- lambda here is the regularisation parameter

L2 regularisation in neural networks:

- Cost function changes to $$ J(w,b)=-1/n \sum_{i=0}^n (L(y,y')) + \sum_{l=1}^L lambda/2m * ||wl||^2 - equation 1$$
- Often known as weight decay
- here ||w||^2 is the matrix 2-norm

How does regularisation reduce overfitting?

- If the value of lambda in equation is too high then inorder to decrease the cost function, value of w*l* will be really low.
- Therefore, weights associated with many neurons are going to be close to zero.
- Also, many activation functions like tanh, sigmoid have linear behaviour for small values (near to zero).
- Weights ar going to be close to 0 due to large value of lambda leading to nearly zero inputto a neuron (weight*x).

- Therefore, the network will not be abe o learn the nonlinear behaviour which was leading to overfitting.

Dropout Regularisation:

- Regularisation technique applied to avoid overfitting int he network
- Dropout layer is not applied during test time as we dont want random output for a test point
- As a random hidden node can be crossed out in dropout regularisation, large weights are not assigned to a particular node.
- The weights of the hidden nodes are spread out prevening assigning large value weight to a node which can be crossed out randomly.

Other Regularisation techniques:

- **Data Augmentation** - If you are working with images then you can flip the existing image data horizontally / zoom in / crop them to create a larger dataset.
    - This doesnt add large value in terms of features to your dataset but is an easy way of getting more data.
- **Early Stopping** - Not used extensively rather prefer using L2 regularisation