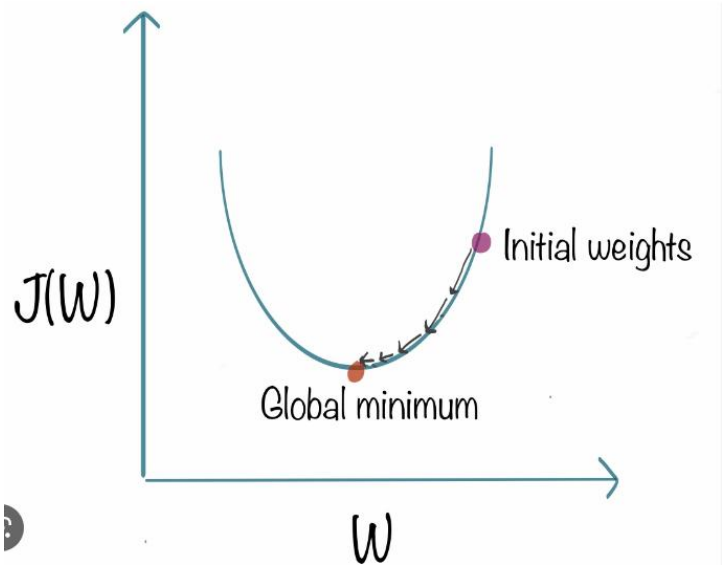
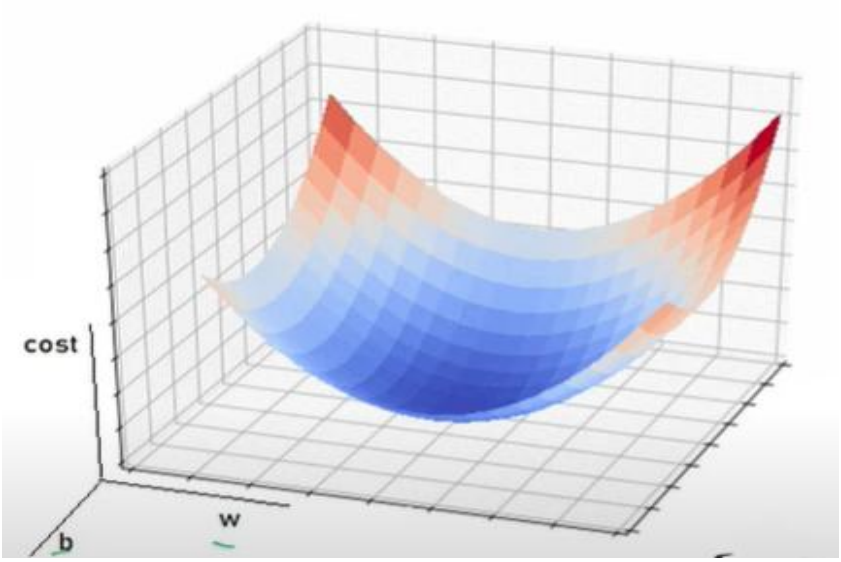
Optimizers in Deep Learning

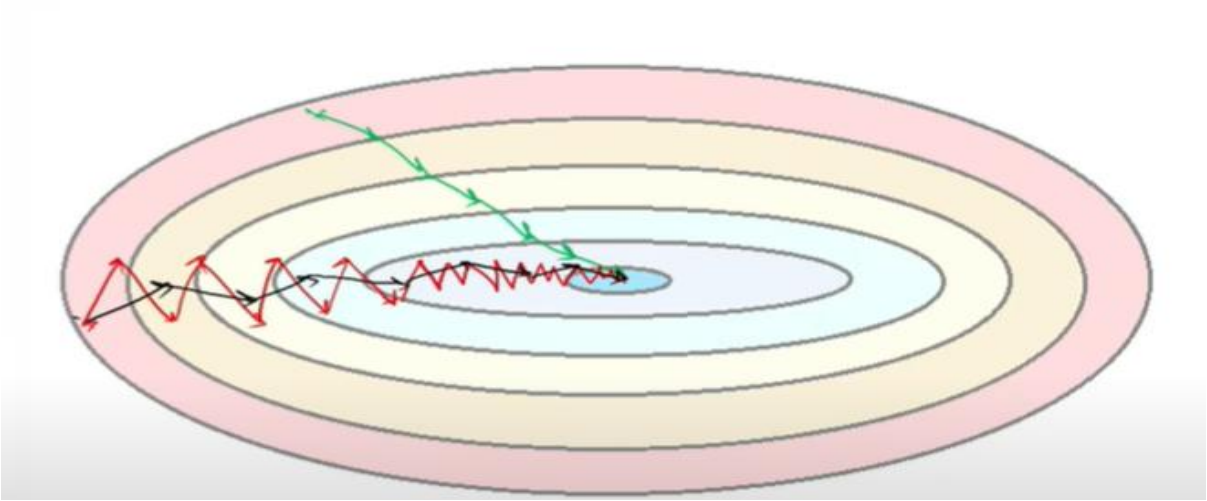
- **Optimizers** are algorithms or methods used to minimize an error function (*loss function*) or to maximize the efficiency of production.
- Optimizers are mathematical functions which are dependent on model's learnable parameters i.e Weights & Biases.
- Optimizers help to know how to change weights and learning rate of neural network to reduce the losses.



3D view of optimizers



2D view of optimizers



Gradient Descent

- It is dependent on the derivatives of the loss function for finding minima.
- It uses the data of the **entire training set** to calculate the gradient of the cost function to the parameters which requires large amount of memory and slows down the process.

$$W_{new} = W_{old} - \alpha * \frac{\partial(Loss)}{\partial(W_{old})}$$

Advantages of Gradient Descent

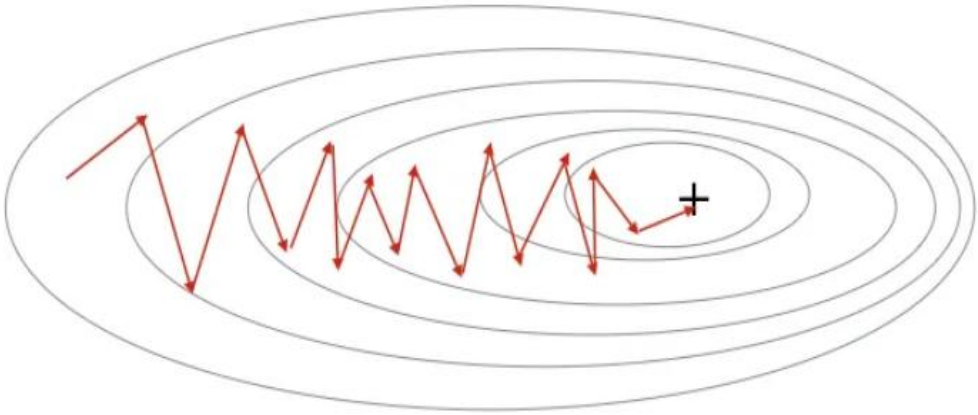
1. Easy to understand
2. Easy to implement

Disadvantages of Gradient Descent

1. Because this method calculates the gradient for the entire data set in one update, the calculation is very slow.
2. It requires large memory and it is computationally expensive.

Stochastic Gradient Descent

- It is a variant of Gradient Descent. It updates the **model parameters one by one**. If the model has 10K dataset SGD will update the model parameters 10k times.



Stochastic Gradient Descent

Advantages of Stochastic Gradient Descent

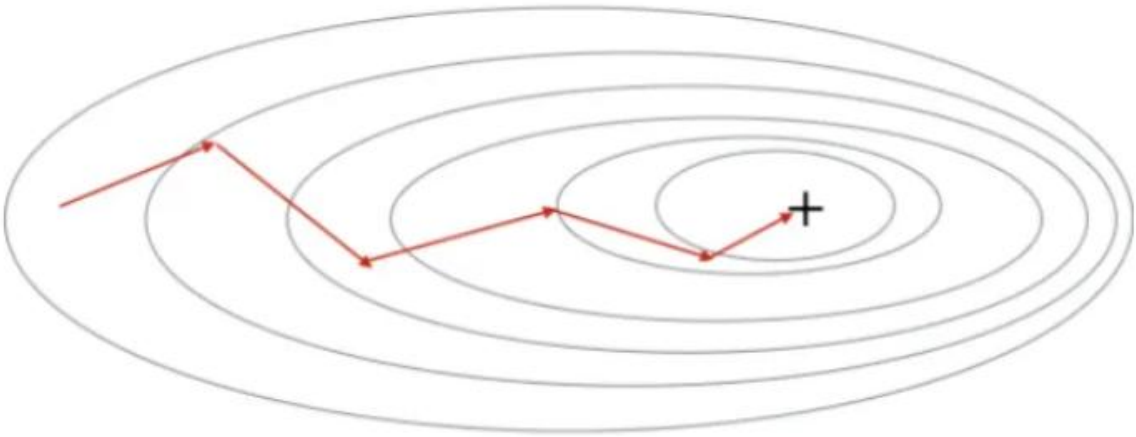
1. Frequent updates of model parameter
2. Requires less Memory.
3. Allows the use of large data sets as it has to update only one example at a time.

Disadvantages of Stochastic Gradient Descent

1. The frequent can also result in noisy gradients which may cause the error to increase instead of decreasing it.
2. High Variance.
3. Frequent updates are computationally expensive.

Mini-Batch Gradient Descent

- It simply **splits the training dataset into small batches** and performs an update for each of those batches.
- This creates a balance between the robustness of stochastic gradient descent and the efficiency of batch gradient descent.
- It can reduce the variance when the parameters are updated, and the convergence is more stable.



Mini Batch Gradient Descent

Advantages of Mini Batch Gradient Descent:

1. It leads to more stable convergence.
2. more efficient gradient calculations.
3. Requires less amount of memory.

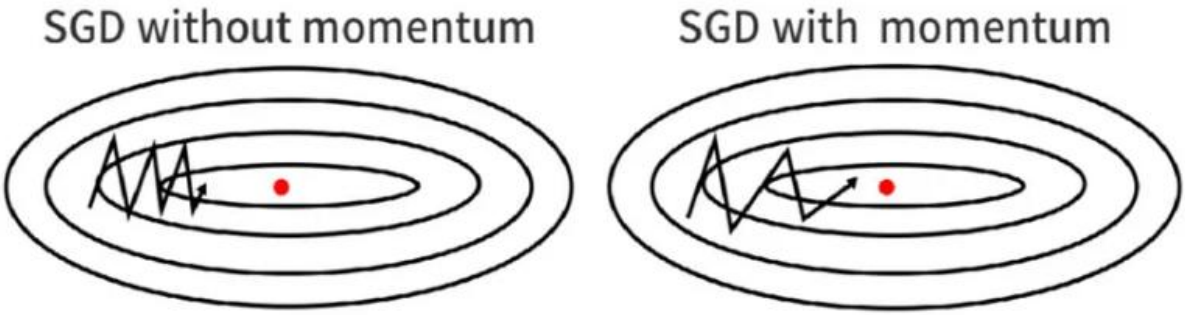
Disadvantages of Mini Batch Gradient Descent

1. Mini-batch gradient descent does not guarantee good convergence,
2. If the learning rate is too small, the convergence rate will be slow. If it is too large, the loss function will oscillate or even deviate at the minimum value.

SGD with Momentum

SGD with Momentum is a stochastic optimization method that adds a momentum term to regular stochastic gradient descent.

Momentum simulates the inertia of an object when it is moving,



SGD with Momentum

Advantages of SGD with momentum

1. Momentum helps to reduce the noise.
2. Exponential Weighted Average is used to smoothen the curve.

Disadvantage of SGD with momentum

1. Extra hyperparameter is added.

Nesterov Accelerated Gradient (NAG)

- Its an optimizer that is an **upgraded version of momentum optimizers** and mostly it performs well than momentum optimizers.
- Look before jump
- Provide momentum on momentum
- It has faster convergence compare to previous one.

AdaGrad(Adaptive Gradient Descent)

The intuition behind AdaGrad is can we use **different Learning Rates** for each and every neuron for each and every hidden layer based on different iterations.

Advantages of AdaGrad

1. Learning Rate changes adaptively with iterations.
2. It is able to train sparse data as well.

Disadvantage of AdaGrad

1. If the neural network is deep the learning rate becomes very small number which will cause dead neuron problem.

Sparse data is **a type of data that does not contain the actual values of features**; it is a dataset containing a high amount of zero or null values

RMS-Prop (Root Mean Square Propagation)

- RMS-Prop basically **combines momentum with AdaGrad.**
- RMS-Prop is a special version of Adagrad in which the learning rate is an exponential average of the gradients instead of the cumulative sum of squared gradients.

Advantages of RMS-Prop

1. In RMS-Prop learning rate gets adjusted automatically and it chooses a different learning rate for each parameter.

Disadvantages of RMS-Prop

1. Slow Learning

Adam (Adaptive Moment Estimation)

- Adam optimizer is one of the most popular and famous gradient descent optimization algorithms.
- It is a method that computes adaptive learning rates for each parameter.
- It's a **combination of momentum & RMSProp**.
- Runs faster
- If it is far away from local minima it takes larger step but as it comes nearer it takes small step.

Advantages of Adam

1. Easy to implement
2. Computationally efficient.
3. Little memory requirements.

What is the best Optimization Algorithm for Deep Learning?

- In general, a normal gradient descent algorithm is more than adequate for simpler tasks.
- If you are not satisfied with the accuracy of your model you can try out RMSprop or add a momentum term to your gradient descent algorithms means use ADAM.
- **Adam is the best optimizer. If one wants to train the neural network in less time and more efficiently then Adam is the optimizer.**