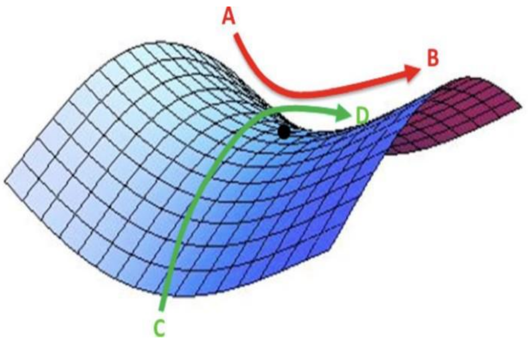
# DEEP LEARNING

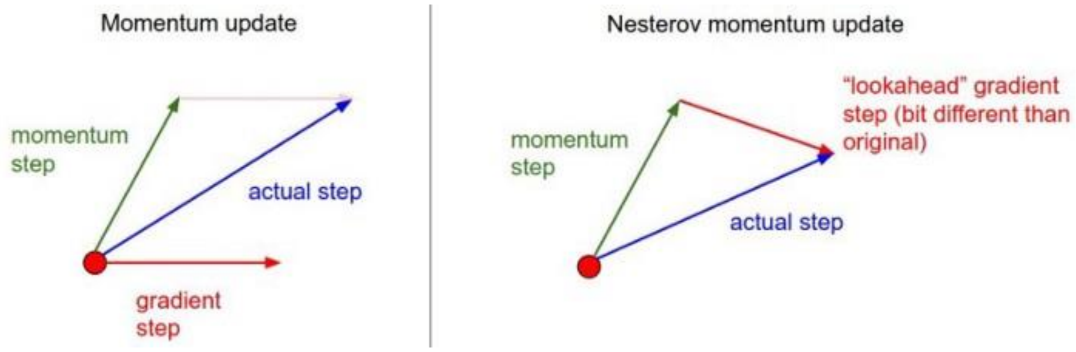
## WORKSHEET-5 ANSWER KEY

1. (D) All of the above
2. (A) Sigmoids do not saturate and hence have faster convergence
3. (D) None of the above
4. (A) True. the output of the tanh is between -1 and 1, it thus centers the data which makes the learning simpler for the next layer.
5. (B) Xavier Initialisation
6. (A) learning rate shrinks and becomes infinitesimally small
7. (B) momentum must be high and learning rate must be low. Because, higher learning rates will never reach the global minima, high momentum and high learning rate will create a problem of exploding gradients whereas, low momentum and low learning rates will raise a problem of vanishing gradient by getting stuck at local minima.
8. (C) when it has many saddle points and flat areas.
9. (A) ADAM
   (C) NADAM
   (D) RMS Prop.
10. (C) when it reaches global minimum
    (D) when it reaches some local minima which is similar to global minima (i.e. which has very less error distance with global minima)
11. **Convex Optimization:**
    It is an optimizing technique which involves a function in which there is only one optimum, corresponding to the global optimum (maximum or minimum). There is no concept of local optima for convex optimization problems, making them relatively easy to solve.
    **Non-Convex Optimization:**
    It is an optimization technique which involves a function which has multiple optima, only one of which is the global optima. Depending on the loss surface, it can be very difficult to locate the global optima.
12. Saddle points are the stable points which have local minima in one direction but local maxima in other direction.



In the adjacent figure, the point shown in black is a saddle point, It has a local minimum along the direction AB but has a local Maximum along the direction CD. Since, it looks like a saddle which is put over a horse, it is name d as a saddle point. This is a non-convex function with a global minimum located within a long and narrow valley. Finding the valley is relatively easy, but it is difficult to converge to the global minimum due to the flat valley, which thus has small gradients so it is difficult for gradient-based optimization procedures to converge.

13.



The main difference is in classical momentum you first correct your velocity and then make a big step according to that velocity (and then repeat), but in Nesterov momentum, you first make a step into velocity direction and then make a correction to a velocity vector based on a new location (then repeat). Simply we can say that in Nesterov Momentum, the gradient step is adaptive.

14. Pre-Initialisation is common for convolutional networks used for examining images. The technique involves importing the weights of an already trained network (such as VGG16) and using these as the initial weights of the network to be trained.

    This technique is only really viable for networks which are to be used on similar data to that which the network was trained on. For example, VGG16 was developed for image analysis, if you are planning to analyse images but have few data samples in your data set, pre-initialization might be a tenable method to utilize. This is the underlying concept behind transfer learning, but the terms pre-initialization and transfer learning are not necessarily synonymous.

15. In neural networks, the output of the first layer feeds into the second layer, the output of the second layer feeds into the third, and so on. When the parameters of a layer change, the distribution of inputs to subsequent layers also changes. We define Internal Covariate Shift as the change in the distribution of network activations due to the change in network parameters during training.

    These shifts in input distributions can be problematic for neural networks, as it has a tendency to slow down learning, especially deep neural networks that could have a large number of layers.

    It is well established that networks converge faster if the inputs have been whitened (ie zero mean, unit variances) and are uncorrelated and internal covariate shift leads to just the opposite.

    **Batch normalization** is a method intended to mitigate internal covariate shift for neural networks.