Question 1:

Why is gradient descent important in machine learning?

Gradient Descent is used in machine learning to find parameter values which would minimize the error on any dataset. An error would be defined as the difference between a prediction and an actual output. It can be thought as a blindfolded person going down a valley to reach the lowest point. In this process the person has to apply a trade off between his step size (learning rate) and target point (lowest point). A bigger step size could result in missing the lowest point and vice versa.

How does plain vanilla gradient descent work?

Vanilla descent is the pure form of gradient descent. It works by taking small steps in the direction of the gradient. Vanilla descent helps in calculating the error for each of the sample in the data and then updates the same once the output has been calculated.

Two modifications to plain vanilla gradient descent.

Momentum

In momentum, past gradients are considered to smoothen out the update. Using the momentum, we would generally avoid the zig zag behaviour and also avoids getting stuck at the saddle points. It helps in computing an exponentially weighted average of the gradients which would then be used to update the weights. In effect, the momentum works faster than the pure vanilla gradient descent algorithm.

Nesterov’s Accelerated Gradient

Even though momentum can be good algorithm, it can sometimes overshoot. If the gradient keeps on getting flatter or is reverses direction ahead, there is a need to slow down. Nesterov’s Accelerated Gradient helps us in calculating the gradient someway ahead of where we are now. This method can be viewed as a correction factor for the momentum method.