Machine Learning worksheet -1

Q.1- **B**

Q.2- **B**

Q.3- **B**

Q.4- **C**

Q.5- **C**

Q.6- **A**

Q.7- **A**

Q.8- **C**

Q.9- **A-B-C**

Q.10- **A-D**

Q.11- **C**

Q.12-Which Linear Regression training algorithm can we use ?

you could use batch gradient descent stochastic gradient descent or mini-batch gradient descen.

When we have large amount of data, you can use a variation of gradient descent called stochastic gradient descent. In this variation, the gradient descent procedure described above is run but the update to the coefficients is performed for each training instance, rather than at the end of the batch of instances. The first step of the procedure requires that the order of the training dataset is randomized. This is to mix up the order that updates are made to the coefficients. Because the coefficients are updated after every training instance, the updates will be noisy jumping all over the place, and so will the corresponding cost function. By mixing up the order for the updates to the coefficients, it harnesses this random walk and avoids it getting distracted or stuck.

The update procedure for the coefficients is the same as that above, except the cost is not summed over all training patterns, but instead calculated for one training pattern. The learning can be much faster with stochastic gradient descent for very large training datasets and often you only need a small number of passes through the dataset to reach a good or good enough set of coefficients, e.g. 1-to-10 passes through the dataset.

Optimization is a big part of machine learning.

Gradient descent is a simple optimization procedure that you can use with many machine learning algorithms.

Batch gradient descent refers to calculating the derivative from all training data before calculating an update.

Stochastic gradient descent refers to calculating the derivative from each training data instance and calculating the update immediately.

Q.13- Which algorithms will not suffer or might suffer, if the features in training set have very different scale?

If the features in our training set have very different scales, the cost function will have the shape of an elongated bowl, so the Gradient Descent algorithms will take a long time to converge. To solve this you should scale the data before training the model. Note that the Normal Equation will work just fine without scaling. A gradient measures how much the output of a function changes if you change the inputs a little bit

The normal equations method does not require normalizing the features, so it remains unaffected by features in the training set having very different scales. Feature scaling is required for the various gradient descent algorithms. Feature scaling will help gradient descent converge quicker.

Common examples of algorithms with coefficients that can be optimized using gradient descent are Linear Regression and Logistic Regression.