FLIP ROBO

Machine learning worksheet 1

1. The computational complexity of linear regression is
   $O(n)$
2. Which of the following can be used to fit nonlinear data?
   Lasso Regression
3. Which of the following can be used to optimize the cost function of linear regression?
   Gradient descent
4. Which of the following method does not have closed from solution for its coefficient?
   Lasso
5. Which gradient descent algorithm always give optimal solution?

   Mini-batch Gradient Descent

6. Generalization error measures how well a model performs on training data

   False

7. The cost function of linear regression can be given as -----------

   I do not matter weather half is there or not

8. Which of the following will have symmetric relation between dependent variable and independent variable?

   Correlation

9. Which of the following is true about normal equation used to compute the coefficient of the linear regression?
   We don't have to choose the learning rate
   It becomes slow when number of features are very large
   We need to iterate

10. Which of the following statement are true if we generated data with the help of polynomial features with 5 degrees of freedom which perfectly fits the data?

Linear regression will have low bias and high variance

Polynomial with degree 5 will have low bias and high variance

11. Which of the following sentence is false regarding regression?

It discovers causal relationship is false here

No interference can be made from regression line is false here

12. Which linear regression training algorithm can we use if we have a training set with millions of features?

If we have a training set with millions of features first, we need to check whether there are any irrelevant features present, and we will need to make use of feature selection methods to ensure that only the one's that can provide a better model are being selected. There are various algorithms used to ensure that we pick out the most impacting feature since we have millions of features, we can afford to remove the feature that have a very low to zero contribution while building a machine learning model. Few of the algorithms that can be used are Lasso regression where with the help of lasso we can shrink the features by making the non-contribution to a zero. Variable inflation factor helps to detect Multicollinearity visible through a heat map and can check if a feature can be neglected. We need to be cautious to make multiple tests and check through various proofs before deciding on removing features from a particular data set to avoid the curse of dimensionality. We also have the wrapper method that allows us to simply take up on the subset/sample of the whole population train the model using less time and computational resource eventually allowing us to work on larger data sets without hampering productivity. Then we batch gradient descent, stochastic gradient descent and mini-batch gradient descent methods than allows us to again use chunks of data set and work upon them ensuring we can cover up the entire data set containing millions of features in it.

13. <u>Which algorithms will not suffer or might suffer, if the features in training set have very different scales?</u>

Feature scaling is required to ensure that the model which we are building do not get biased due to the difference in magnitude created by the difference in unit for features present in a particular data set. Also, in linear regression if we use feature scaling techniques then the convergence for gradient descent will happen sooner as compared to times when we do not perform any scaling technique to make all the feature data in some uniform pattern. Therefore, algorithms like KNN in supervised machine learning and K Means in unsupervised machine learning that make use of the Euclidean distance can suffer impact of unscaled feature datasets. However, there are algorithms that do not require feature scaling as well or rather even if performed do not give any kind of impact over the outcome and they are Decision tree, Random Forest algorithm, XG Boost that are mostly into ensemble techniques. Also, the two most common scaling methods used are normalization and standardization. In normalization we can scale down our features between zero to one whereas in standardization we scale down the features using the standard normal distribution using the mean as zero and standard deviation as one. So, in conclusion feature scaling is necessary for gradient descent and distance related techniques but is not always needed to be performed in scenarios where tree formation decision making is performed, or any kind of ensemble approach were performing or not performing a feature scaling does not impact the output so it will be quite logical to skip on scaling to save time.