Machine Learning Notes

**Basic definitions & use cases**

* Supervised Learning- Right answers are given and we produce more of these right answers
* Regression problem – we predict a continuous valued problem
* Classification problem- Discrete valued problem (0 or 1 or could be more)
* Unsupervised learning
  + When the training data is not labelled
  + Uses clustering algorithm
    - Applications-
      * Organize computer clusters
      * Social network analysis
      * Market segmentation
      * Astronomical data analysis

**Different machine learning algorithms**

* Linear Regression
  + Univariate linear regression- Linear regression with one variable
  + Multivariate linear regression- linear regression with multiple variables
    - Feature scaling- When applying gradient descent while doing multivariate linear regression we should scale our features and ensure that they fall in approx. in the -1 <= x <= 1 range. This can help us find the optimal solution in shorter time.
    - Mean Normalization- Change a feature’s value’s such that the mean is zero. Formula used is- x - mean/ max(x) - min(x)
  + Ways to find the solution of a linear regression problem-
    - Gradient descent- Continually decrease the parameters of the regression equation by a small factor
    - Normal Equation- Solve for optimal value of parameters analytically
* Classification
  + Also known as logistic regression where the variable to be predicted- Y, takes on discrete values
  + Hypothesis function is this case is the [sigmoid function](https://en.wikipedia.org/wiki/Sigmoid_function) or the logistic function, which gives us 0 <= h(x) <= 1
* Regularization-
  + Method to prevent overfitting in regression
  + Done by penalizing parameters-->smaller parameter values-->simpler hypothesis-->less prone to overfitting
* Neural Networks
  + Some problems have many features
  + These problems can’t just be solved using logistic regression by including quadratic features or other higher order features, because of the infeasibility of the problem.

1. What is a training and testing data-set?

When we use supervised learning in machine learning, which is using labelled data for inferring a function from the data, we divide our learning data into two sets – training data and testing data.

Training data – The training data as the name suggests is used to train a model. To elaborate we lay down one or more features describing one or more contextual properties of our data and then feed the data and these feature values to our model. We then try to deduce a model function that fits our data as closely as possible.

Testing data- The testing data is used to test how well our model was trained using our training data and how well it can perform on data-points outside of the training data set. We do this by feeding the test data-points to the model function deduced after training and comparing the actual results with the expected results.

2. What is meant by model variance and model bias? Which one would we prefer and why?

Both model bias and variance are sources of error that prevent our ML model from performing optimally.

If we could repeat the model building process, using different training data sets and then test these models on one test data set, then there will be a range of predictions, due to the randomness in our training data sets. Bias measures how far-off these predictions are from the actual value.

Same way, if we could repeat the model building process, using different training data sets, also assuming that this time our training data set has an outlier, and then test these models on one test data set, then variance is how much predictions for the outlier vary across different model realizations.

We want to choose a model function that minimizes both bias and variance.

3. What is overfitting and how does it relate to model complexity?

Overfitting is when our model function is so closely fit to the training data that it loses its generality. If we overfit data, we’ll get drastically different model functions for slightly different data sets. Overfitting increases with increasing model complexity.

4. What is cross-validation?

Cross-validation is a practice in which we add an extra model validation layer in our machine learning process. In cross-validation we train our model on several subsets of the data and then evaluate it on the complementary data. This practice prevents overfitting of our model to any one given dataset.