

HW 2: Problem 7

Summary of Practical Machine Learning Advice:

If learning algorithm is not working as expected, instead of simply obtaining more data and or computing different features, we should first debug the issue and based on the analysis, take the right corrective steps.

Learning algorithms – Diagnostics:

High Variance: Training error will be much lower than test error.

High Bias: Training error will also be high.

Options to fix above two issues:

Fixes for high variance: Get more training data or use smaller set of features

Fixes for high bias : Use larger set of features or use different features.

Diagnostic options for algorithms which involve optimization problems e.g. Gradient descent and other similar algorithms:

We should always check if the objective function is converging at each iteration. Also, Bias vs variance test is good for such algorithms also.

We should make sure if we are using correct objective function. We should also check if we are using correct parameters for the algorithm. e.g. the regularization constant C in SVM etc.

Based on the analysis, various options to fix an optimization algorithm are :

Fix the algorithm by running more iterations or using a different optimization method.

Use different parameters specific to the algorithm.

We should always do error analysis of learning algorithm to find out various details of learning system such as how much error can be attributed to each component of the system, which components corrects most of the error. Based on such analysis, we can make smart choices to reduce computation required for the learning system.

Ablative analysis of an algorithm involves comparing the algorithm's performance against a baseline. This analysis helps in understanding how much each of the component in the algorithm contributed towards its performance.

Other than above listed methods, we should use problem specific techniques to debug the issues. There is no such general technique. We should try to devise them based on the problem.

In general, we should try to solve any machine learning problem in an iterative manner i.e. build a quick prototype, test it and then keep iterating to further improve it. This allows us to beat the time to market in a commercial setup.

Some common pitfalls to avoid:

- Don't over-theorize the problem. Certain components may not be useful. So we should first build the system and identify improvement areas.
- Starting with developing difficult parts first rather than developing simpler ones first.