***Student: Victor Pugliese***

***R#: 11492336***

***Course: Independent Study on Python Machine Learning for Petroleum Engineering Application (PETR 5000)***

***Self-Homework #4***

1. ***What Linear Regression training algorithm can you use if you have a training set with millions of features?***

It is suggested using the Stochastic Gradient Descent, the Mini-batch Gradient Descent or the Batch Gradient Descent, because the Normal Equation has to invert a 1Mx1M matrix, which consumes a lot of computational power.

1. ***Suppose the features in your training set have very different scales. What algorithms might suffer from this, and how? What can you do about it?***

If the training set has features with different scales, the Gradient Descent algorithm could suffer from this. It would take more time to get the minimum. We can scale the data before performing the training procedure.

1. ***Can Gradient Descent get stuck in a local minimum when training a Logistic Regression model?***

Since the cost function is convex, the Linear Regression model will not get stuck in a local minimum.

1. ***Do all Gradient Descent algorithms lead to the same model provided you let them run long enough?***

The model parameters will be the same if the learning rate is small. If the learning model is too high, the Stochastic GD and Mini-Batch GD will be jumping around the absolute minimum.

1. ***Suppose you use Batch Gradient Descent and you plot the validation error at every epoch. If you notice that the validation error consistently goes up, what is likely going on? How can you fix this?***

If the validation error consistently goes up, the learning rate could be too large, or the model is overfitting. We can fix this using a smaller learning rate or using a simplified model.

1. ***Is it a good idea to stop Mini-batch Gradient Descent immediately when the validation error goes up?***

The Mini-batch gradient Descent use random instances to fit the model, because of that the error could go up during different iterations. It is not a good idea stop the algorithm when this happen

1. ***Which Gradient Descent algorithm (among those we discussed) will reach the vicinity of the optimal solution the fastest? Which will converge? How can you make the others converge as well?***

The Stochastic GD algorithm will reach the vicinity of the optimal solution faster. The Batch GD will converge. The others will converge if the learning rate is decreased as we get closer to the vicinity of the optimal solution.

1. ***Suppose you are using Polynomial Regression. You plot the learning curves and you notice that there is a large gap between the training error and the validation error. What is happening? What are three ways to solve this?***

The model is overfitting. We could reduce the degree of the polynomial or increase the number of instances in the training set to solve this. Also, we could use a regularized linear model.

1. ***Suppose you are using Ridge Regression and you notice that the training error and the validation error are almost equal and fairly high. Would you say that the model suffers from high bias or high variance? Should you increase the regularization hyperparameter α or reduce it?***

The model suffers from high bias (underfitting). We should reduce the regularization hyperparameter.

1. ***Why would you want to use:***

* ***Ridge Regression instead of Linear Regression?***

In case you have a overfitting model, the regularization lead to less extreme prediction.

* ***Lasso instead of Ridge Regression?***

When you suspect that a few features are not important, you could use Lasso Regression.

* ***Elastic Net instead of Lasso?***

Elastic Net is preferred over Lasso because Lasso may behave erratically when the number of features is greater than the number of training instances or when several features are strongly correlated.

1. ***Suppose you want to classify pictures as outdoor/indoor and daytime/nighttime. Should you implement two Logistic Regression classifiers or one Softmax Regression classifier?***

We need to implement 2 Logistic Regression classifier because Softmax Regression should be used only with mutually exclusive classes.

1. ***Implement Batch Gradient Descent with early stopping for Softmax Regression (without using Scikit-Learn).***

See file: HML\_Chap04\_Exercise\_12.py