# COMP6247 : Recursive Least Square Assignment

Thanakorn Panyapiang
(31446612, tp2n19@soton.ac.uk)

## 1 Recursive Least Square Derivation

I have studied the derivation of the Recursive least sqaure algorithm.

## 2 Gradient Descent

The solution estimated using the pseudo inverse, gradient descent, and stochastic gradient descent is shown in Fig 1, 2, and 3 respectively. The global trend of the learning curve in both gradient descent solutions is similar, as the squared error reduces over time and the decrease is significant at the first few iterations before stabilize later. However, it can be noticed clearly that while the curve is completely smooth, the learning curve fluctuates on the stochastic solution. This behavior is the result of adjusting the model using a single data point's gradient because its direction could be different from the total gradient.

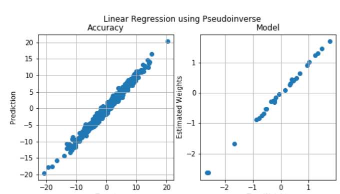



**Figure 1:** Pseudoinverse solution

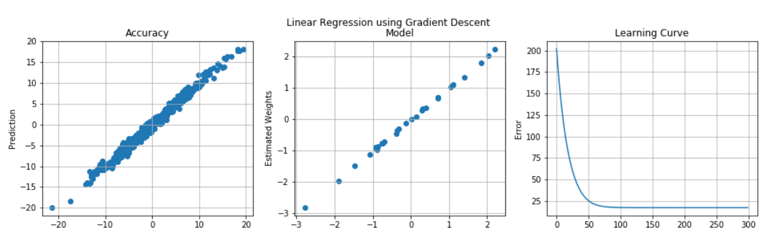



**Figure 2:** Gradient descent solution

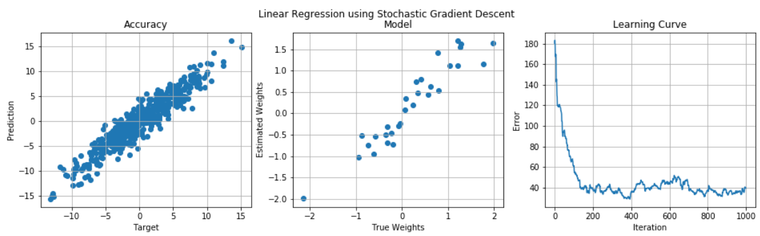
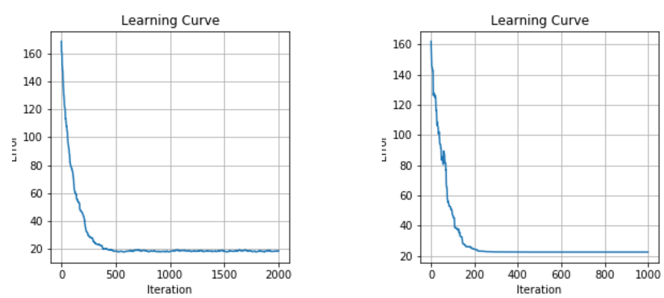



**Figure 3:** Stochastic gradient descent solution

There are several options for solving stochastic gradient descent issue. The easiest one would be to have a small learning rate so that each model adjustment is small as shown in Fig 4a, but it may decrease the speed of convergence.

The more efficient approach for solving this problem is *learning rate decay*. The principle of the technique is to reduce the learning rate over time. Learning rate decay allows the model to be updated quickly at the beginning so it can reach the global minimum. Once it is close to the global minimum, the model adjustment will be smaller to prevent overshoot. Fig 4b shows the learning curve of stochastic gradient descent with the same parameter as Fig 3 after applying the learning rate decay. It can be observed that the learning curve is still noisy at the beginning, but the model start to converge once the learning rate becomes low after around 200 iterations.




**(a)** Small learning rate **(b)** Learning rate decay

**Figure 4:** Improving SGD convergence

# 3 Recursive Least Square

The solution obtained from Recursive Least Square algorithm is show in Fig 5. The learning curve obsurved from the algorithm is very similar to stochastic gradient descent. However, the speed of convergence is noticeably faster as the model starts to stabilize after about 100 iterations compare to stochastic gradient descent which converges after 200 iterations.

Regarding data dimensionality, it can be observed that, as the dimention of data increase, the number of iteration which the algorithm uses before converge is higher. The result of this experiment is shown in Table 1.

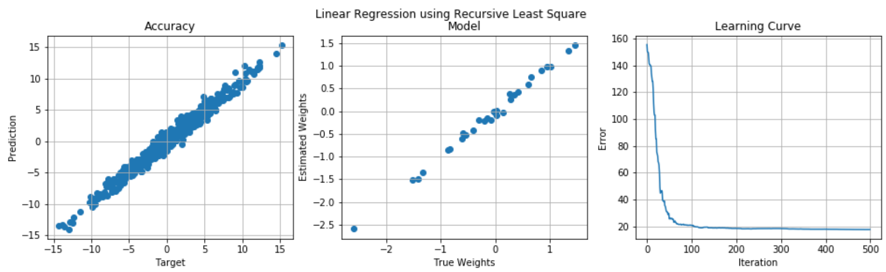



**Figure 5:** Recursive least sqaure solution

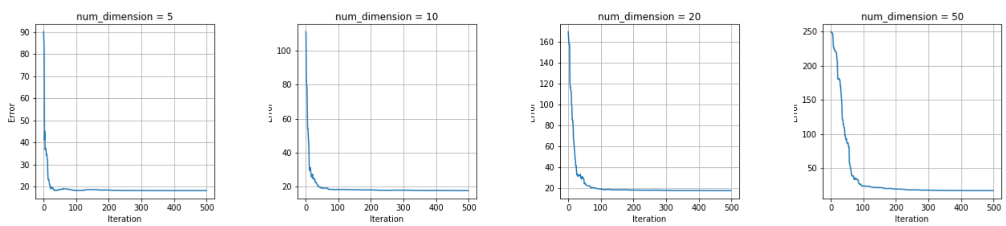
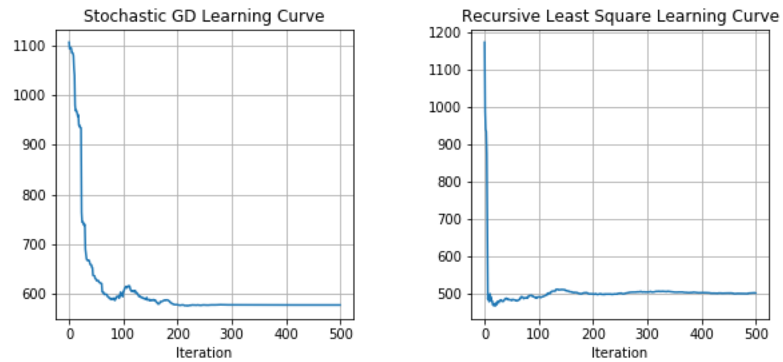



**Table 1:** Speed of convergence compare to number of dimensions

# 4 Recursive Least Square on UCI dataset

The data used in this section is the *Auto MPG* dataset(`https://archive.ics.uci.edu/ml/datasets/auto+mpg`) which contains 536 samples with 4 continuous features.

With the same number of iterations, it turns out that *Recursive least sqaure* algorithm performs slightly better in terms of total sqaured error. Regarding the speed of convergence, Recursive least square converges much faster at the beginning and become stable since around 400 iterations.




In terms of accuracy, both models perform similarly on the validation set(30 percent data from the whole dataset) as shown in the figure below.