Hala Raslan

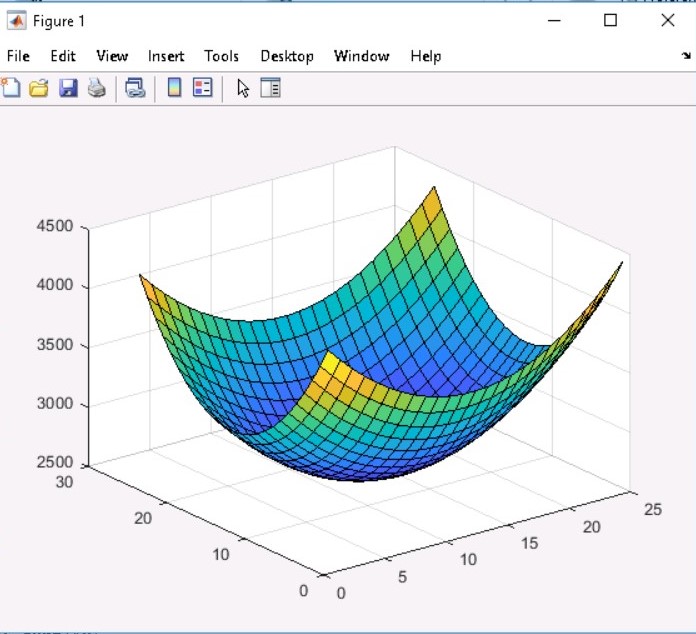
1) x1=[-12:1:12];

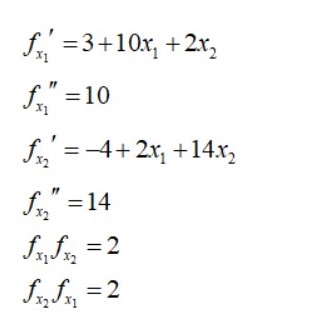
x2=[-12:1:12];

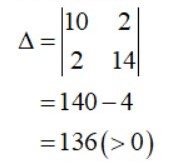
[X1,X2]=meshgrid(x1,x2);

Fx=3\*X1-4\*X2+5\*X1.^2+2\*(X1\*X2)+7\*X2.^2;

surf(Fx)







Therefore, it is a convex function.

1. Yes

2) Which Linear Regression training algorithm can you use if you have a training set with millions of features?

**It would be ideal to use stochastic, mini-batch, and batch gradient descent. With batch gradient descent, it is conditional depending if you have enough memory. We want to avoid using normal equation or SVD since it can get slow with higher number of features.**

3) Suppose the features in your training set have very different scales. (a) Which algorithms might suffer from this, and how? (b) What can you do about it?

**(a) Different scales can lead to an elongated bowl versus a bowl. This means that it will take gradient descent algorithms a longer time to converge. Take for example, batch gradient. Batch gradient will be painfully slow.**

**(b) You can scale the data before training. For example, for every data value, the mean will be subtracted, and that result will be divided by standard deviation. Another option is to take the mean-max value. We can range them from 0 to 1 or -1 to +1 using minimum and maximum value per column.**

4) Do all Gradient Descent algorithms lead to the same model, provided you let them run long enough? Explain.

**Let’s assume the cost function is convex with a normal (not too high) learning rate. This will lead to a production of similar models. An exception to this is when the learning rate is too high. If you don’t lower the learning rate, stochastic and minibatch gradient descent algorithms will have a hard time to converge thus producing different models (it can diverge).**

5) Suppose you use Batch Gradient Descent and you plot the validation (or, test) error at every epoch. (a) If you notice that the validation error consistently goes up, what is likely going on? What if the training error is (b) either going up or (c) going down at the same time? (d) How can you fix this?

**If the validation error goes up, then that could mean the learning rate is very high and the batch gradient descent algorithm is diverging. Training error determines how well it has learned the training data. If the training error is going up, then that could mean your model is diverging and you should lower the learning rate. If the training error is going down, then this could mean you are overfitting. You can fix overfitting by stopping the training process a bit earlier.**

6) (a) Which Gradient Descent algorithm (among those we discussed) will reach the vicinity of the optimal solution the fastest? (b) Which will actually converge? (c) How can you make the others converge as well? Explain.

**(a) Stochastic Gradient Descent – since it picks a random instance in training set at every step and computes gradients based only on that single instance.**

**(b) Batch Gradient Descent- uses all the samples so it should smoothly converge.**

**(c) Stochastic and minibatch gradient descent will need to have their learning rate lowered to avoid bouncing around.**