E. Checking Gradient Descent for Convergence

Tuesday, 9 September 2025 2:46 PM

1. Introduction:

By learning to recognize what a well-running implementation of Gradient Descent look like, we'll also be better able to choose a good learning rate.

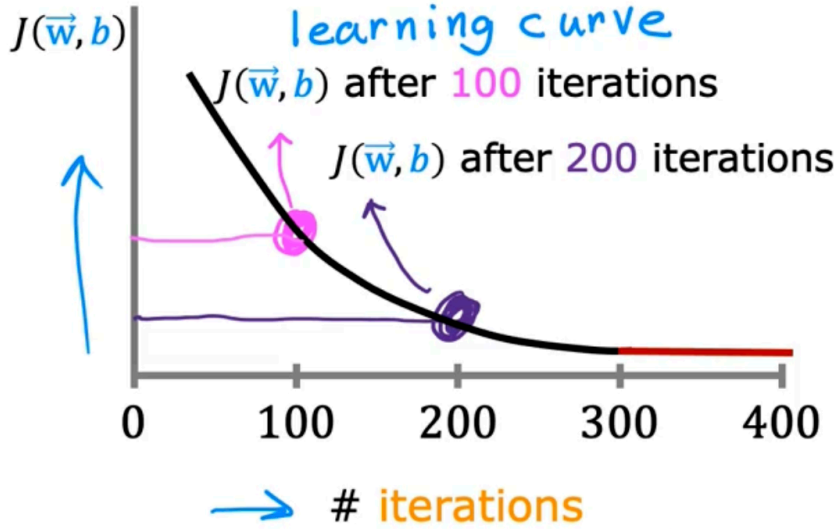
We know that Gradient Descent in Multiple LR is to find parameters w & b that minimize the cost function J. This is done iteratively by constantly updating w and b until we reach the global minimum.

$$\begin{cases} w_j = w_j - \alpha \dfrac{\partial}{\partial w_j} J(\vec{w}, b) \\ b = b - \alpha \dfrac{\partial}{\partial b} J(\vec{w}, b) \end{cases}$$

objective: $\min_{\vec{w},b} J(\vec{w}, b)$

Now, let's plot a graph with the cost function, J (calculated on the training set), on the y-axis vs the no. of iterations of gradient descent on the x-axis.

Remember that each iteration means the completion of each simultaneous update of the parameters w & b.



As the no. of iterations increase, the cost function decreases and we approach the global minimum. This curve is also called a Learning Curve.

Observations from the above graph:

a. Now, if the gradient descent is working properly, we'll see a reduction in the Cost J after every iteration.

b. If J ever increases after one iteration, that means either the learning rate (α) is chosen poorly, and it usually means the α is too large, or there could be a

bug in code.

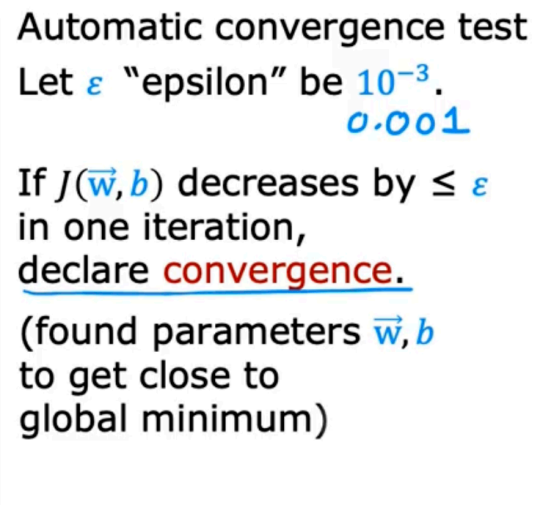
c. Also, we can see from the graph, that as we complete 300 Gradient Descent iterations, the cost J is levelling off and is no longer decreasing much.

d. By 400 iterations, it looks like the cost curve has flattened out. This means we have reached global min for parameters w and b.

Note that, there are different types of learning curves used in Machine Learning. Also the no. of iteration needed to converge varies b/w applications. It could be 30 iterations, 1000 iterations or maybe 100,000 iterations and so on.

Therefore, it is very hard to tell in advance, the no of iterations the gradient descent needs to converge, which is why we can create a graph like the Learning Curve.

Another method to decide if the model is done training is with an Automatic Convergence Test:



However, choosing the right epsilon may prove to be pretty difficult. So making observations from the Learning Curve can be a better option.
