Gradient Descent

Introduction

In machine learning we always want to optimize our predictions to get the best result. For example, in linear regression we optimize the intercept and the slope to get the best fitting regression line for our data. In linear regression the better fitting line we have, the better prediction we get. One of the ways to optimize our prediction is to use something called gradient descent.

Discussion

When using gradient descent to optimize a parameter, it looks at the cost-function in regards to the parameter. Gradient descent uses random initialization which means that it inputs random numbers into the parameter that we want to optimize. The gradient descent then measures the gradients on the cost-function. By measuring the gradients it can find the global minimum which is the point on the cost-function, where the parameter is best optimized for correct predictions.
Gradient descent finds the global minimum by taking small steps where each step it tries to decrease the cost-function.

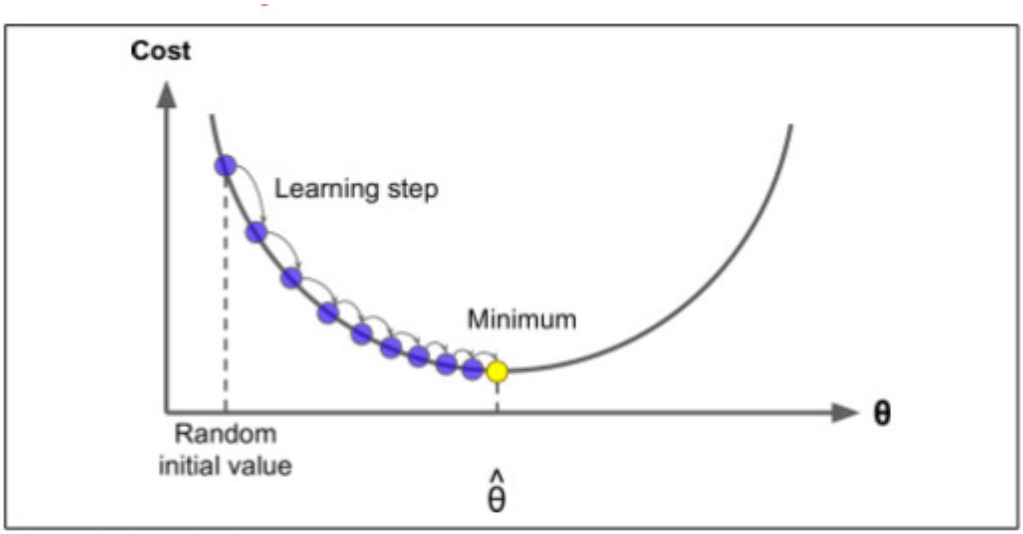

Figure 4-3. Gradient Descent

These learning steps that gradient descent uses can be altered by using a parameter called the learning rate. Learning rate decides how large the steps are for the gradient descent. With a small learning rate it would take more iterations to reach the minimum. If the learning rate is too large the gradient descent could miss the minimum by jumping over it.

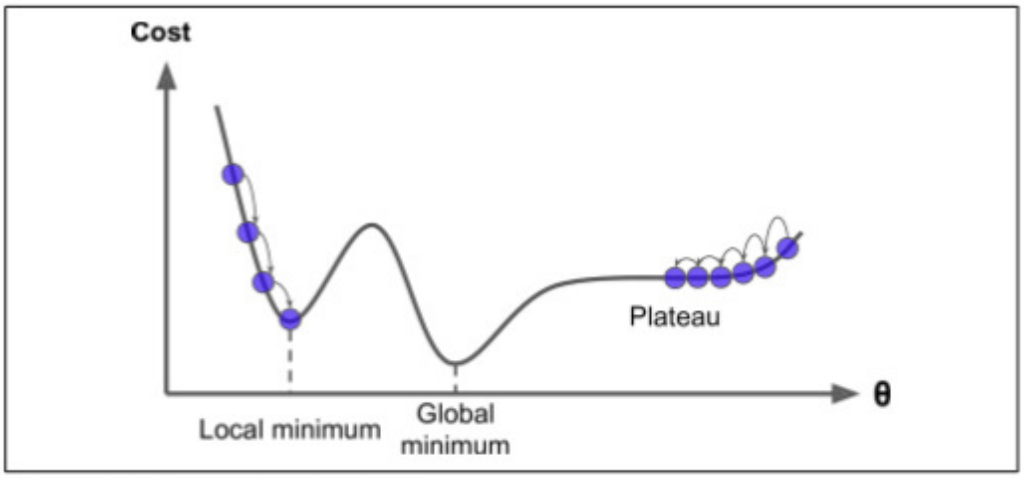

Figure 4-6. Gradient Descent pitfalls

Since there can be different irregularities in the cost-function it is not always certain that gradient descent reaches the global minimum, and could stop at a local minimum. The irregularities can also cause the gradient to become close to zero while still not being the minimum. This would mean that it would take a lot of iterations for the gradient descent to reach the global minimum.

References

edition.[st]Hands-on Machine Learning with Scikit-Learn & TensorFlow by Aurélien Géron, 1