Formulation of Gradient Descent for Linear Regression

Timur Çakmakoğlu

October 2024

1 Gradient Descent Formulation

The estimator of the ith sample with n variables is defined by the following function,

$$h_{\Theta}(X_i) = \Theta \cdot X_i = \theta_0 + \theta_1 x_{i1} + \theta_2 x_{i2} + ... + \theta_n x_{in} \tag{1}$$

$$X_i = \begin{bmatrix} 1 & x_i \end{bmatrix} = \begin{bmatrix} 1 & x_1 & x_2 & ... & x_n \end{bmatrix}$$

To determine the optimal coefficients for the estimator, we need a metric to quantify the error or success of the predictions. For this purpose, we define a cumulative loss function across all samples,

$$L(\Theta) = \frac{1}{m} \sum_{i=1}^{m} (h(X_i) - y_i)^2 \tag{2}$$

where y_i is the observed value of ith sample. This function, also known as mean squared error of the estimator, increases cumulatively as each estimation diverges further from the observed value. Notice that Θ is a variable in the loss function. This allows us to formulate an optimization problem as follows, to find the coefficient values that minimize the prediction error,

$$\min_{\Theta} L(\Theta)$$

This problem can be solved by iteratively updating the coefficients in the direction of the gradient of the loss function, thus reducing the error rate with each iteration:

$$\theta_j := \theta_j - \alpha * \frac{\partial L(\Theta)}{\partial \theta_j} \quad j \in [0, n] \tag{3}$$

where α is the hyperparameter known as the learning rate, which controls the magnitude of each update step. If α is set too low, the updates can become inefficiently small, whereas setting α too high may cause coefficients to diverge rather than converging to a local minimum. Given the estimator in Eq.1 the update rule for each coefficient is defined as follows,

$$\theta_j := \theta_j - \alpha * (\frac{2}{m}(\sum_{i=1}^{m}(h(X_i) - y_i) * X_{ij})) \quad j \in [0, n]$$

2 Normalization

It is generally a good practice to normalize data when features and observed values have with significantly different ranges. One of the most common method is standard scaling, which transforms each feature to have a mean of 0 and a standard deviation of 1. This is achieved by converting each data point to a z-score,

$$z_{ij} = \frac{x_{ij} - \bar{x}_j}{s_j} \tag{4}$$

The gradient descent algorithm is then applied to the scaled features and observed values using the following estimator,

$$h_{\Theta,Z}(X_i) = \Theta \cdot Z_i = \theta_0 + \theta_1 z_{i1} + \theta_2 z_{i2} + ... + \theta_n z_{in}$$

After obtaining the optimal Θ using gradient descent, the equation and coefficients should be converted back to the original scale. To do this, the z-scores must be transformed using Eq.4,

$$\begin{aligned} h_{\Theta,Z}(X_i) &= \theta_0 + \sum_{j=1}^{n} \theta_j \frac{x_{ij} - \bar{x}_j}{s_j} \\ &= \theta_0 + \sum_{j=1}^{n} \frac{\theta_j}{s_j} x_{ij} - \frac{\theta_j}{s_j} \bar{x}_j \\ &= (\theta_0 - \sum_{j=1}^{n} \frac{\theta_j}{s_j} \bar{x}_j) + (\sum_{j=1}^{n} \frac{\theta_j}{s_j} x_{ij}) \end{aligned}$$

This rearrangement shows how the intercept and coefficients should be transformed, respectively. However, since this estimator is optimized according to the scaled target values, the equation must be unscaled as follows,

$$\begin{aligned}
y_{scl} &= (\theta_0 - \sum_{j=1}^{n} \frac{\theta_j}{s_j} \bar{x}_j) + (\sum_{j=1}^{n} \frac{\theta_j}{s_j} x_{ij}) \\
\frac{y - \bar{y}}{s_y} &= (\theta_0 - \sum_{j=1}^{n} \frac{\theta_j}{s_j} \bar{x}_j) + (\sum_{j=1}^{n} \frac{\theta_j}{s_j} x_{ij}) \\
y &= [(\theta_0 - \sum_{j=1}^{n} \frac{\theta_j}{s_j} \bar{x}_j) s_y + \bar{y}] + (\sum_{j=1}^{n} \frac{\theta_j}{s_j} x_{ij}) s_y
\end{aligned}$$

In conclusion, the intercept and coefficients in the original scale are derived as,

$$\begin{aligned}
\theta_0 &:= s_y(\theta_0 - \sum_{j=1}^{n} \frac{\theta_j}{s_j} \bar{x}_j) + \bar{y} \\
\theta_j &:= s_y \frac{\theta_j}{s_j} & j \in [1, n]
\end{aligned}$$