Regularization Term

Both L1 and L2 can add a penalty to the cost depending upon the model complexity, so at the place of computing the cost by using a loss function, there will be an auxiliary component, known as regularization terms, added in order to panelizing complex models.

By adding regularization terms, the value of weights matrices reduces by assuming that a neural network having less weights makes simpler models. And hence, it reduces the overfitting to a certain level.

Penalty Terms

Through biasing data points towards specific values such as very small values to zero, Regularization achieves this biasing by adding a tuning parameter to strengthen those data points. Such as;

1. **L1 regularization:** It adds an L1 penalty that is equal to the absolute value of the magnitude of coefficient, or simply restricting the size of coefficients. For example, Lasso regression implements this method.

2. **L2 Regularization:** It adds an L2 penalty which is equal to the square of the magnitude of coefficients. For example, Ridge regression and SVM implement this method.

3. **Elastic Net:** When L1 and L2 regularization combine together, it becomes the elastic net method, it adds a hyperparameter.

L1 Regularization

L1 regularization is the preferred choice when having a high number of features as it provides sparse solutions. Even, we obtain the computational advantage because features with zero coefficients can be avoided.

The regression model that uses L1 regularization technique is called Lasso Regression.

Mathematical Formula for L1 regularization

For instance, we define the simple linear regression model Y with an independent variable to understand how L1 regularization works.

For this model, W and b represents **“weight”** and **“bias”** respectively, such as

$W= w_1, w_2, w_3, ... w_n$

And,

$b=b_1, b_2, b_3, ... b_n$

And Ŷ is the predicted result such that

$\hat{Y}= w_1 x_1 +w_2 x_2 +......+w_n x_n, + b$

The below function calculates an error without the regularization function

Loss= Error (Y, Ŷ)

And function that can calculate the error with L1 regularization function,

Display the formula of loss function with L1 regularization

Where **λ** is called the regularization parameter and **λ**> 0 is manually tuned. Also, **λ**=0 then the above loss function acts as Ordinary Least Square where the high range value push the coefficients (weights) 0 and hence make it underfits.

Now |w| is only differentiable everywhere except when w=0 as shown below;

Differentiating weight (W)

Substituting the formula of Gradient Descent optimizer for calculating new weights;

Substituting gradient descent optimizer formula to calculate new weight

Putting the L1 formula in the above equation;

Displaying the new weight formula equation with regularization parameters.

From the above formula, we can say that;

- When w is positive, the regularization parameter ($\lambda > 0$) will make w to be least positive, by deducting λ from w.

- When w is negative, the regularization parameter ($\lambda < 0$) will make w to be a little negative, by summing λ to w.

L2 regularization

L2 regularization can deal with the multicollinearity (independent variables are highly correlated) problems through constricting the coefficient and by keeping all the variables.

L2 regression can be used to estimate the significance of predictors and based on that it can penalize the insignificant predictors.

A regression model that uses L2 regularization techniques is called Ridge Regression.

Mathematical Formula for L2 regularization

For instance, we define the simple linear regression model Y with an independent variable to understand how L2 regularization works.

For this model, W and b represents **“weight”** and **“bias”** respectively, such as

$$\mathbf{W= w_1, w_2, w_3, \dots\dots\dots\ w_n}$$

And,

$$\mathbf{b=b1, b2, b3, \dots\dots\dots\ bn}$$

And Ŷ is the predicted result such that

$$\mathbf{\hat{Y}= w1\ x1 +w2\ x2 +\dots\dots+wn\ xn, + b}$$

The below function calculates an error without the regularization function

$$\mathbf{Loss= Error\ (Y, \hat{Y})}$$

And function that can calculate the error with L2 regularization function,

Displaying the formula of loss function with L2 regularization term.

Here, **λ** is known as Regularization parameter, also if the lambda is zero, this again would act as OLS, and if lambda is extremely large, it leads to adding huge weights and yield as underfitting.

Substituting the formula of Gradient Descent optimizer for calculating new weights;

Displaying the formula of new weight with L2 regularization term.

Putting the L2 formula in the above equation;

Showing the new weight equations with L2 regularization term.

L2 vs L1 Regularization

It is often observed that people get confused in selecting the suitable regularization approach to avoid overfitting while training a machine learning model.

Among many regularization techniques, such as L2 and L1 regularization, dropout, data augmentation, and early stopping, we will learn here intuitive differences between L1 and L2 regularization.

1. Where L1 regularization attempts to estimate the median of data, L2 regularization makes estimation for the mean of the data in order to evade overfitting.

2. Through including the absolute value of weight parameters, L1 regularization can add the penalty term in cost function. On the other hand, L2 regularization appends the squared value of weights in the cost function.

3. As defined, sparsity is the characteristic of holding highly significant coefficients, either very close to zero or not very close to zero, where in general coefficients approaching zero would be eliminated later.

   And the feature selection is the in-depth of sparsity, i.e. in place of confining coefficients nearby to zero, feature selection is brought them exactly to zero, and hence expel certain features from the data model.

   In this context, L1 regularization can be helpful in feature selection by eradicating the unimportant features, whereas, L2 regularization is not recommended for feature selection.

4. L2 has a solution in closed form as it's a square of a weight, on the other side, L1 doesn't have a closed form solution since it includes an absolute value and it is a non-differentiable function.

   Due to this reason, L1 regularization is relatively more expensive in computation, it can't be solved in the context of matrix measurement and heavily relies on approximations.

   L2 regularization is likely to be more accurate in all the circumstances, however, at a much higher level of computational costs.

Conclusion

In order to prevent overfitting, regularization is the most-approaches mathematical technique, it achieves this by panelizing the complex ML models via adding regularization terms to the loss function/cost function of the model.

- L1 regularization gives output in binary weights from 0 to 1 for the model’s features and is adopted for decreasing the number of features in a huge dimensional dataset.

- L2 regularization disperse the error terms in all the weights that leads to more accurate customized final models.