XGBoost Report

Donal Loitam

July 30, 2022

Contents

1 Introduction

- XGBoost is an advanced implementation of gradient boosting along with some regularization factors.
- XGBoost falls under the category of Boosting techniques in Ensemble Learning.

2 Key Points of the Algorithm

XGBoost with an example :
Let's look at how XGboost works with an example. Here I'll try to predict a child's IQ based on age. For any basic assumption in such statistical data, we can take the average IQ and find how much variance(loss) is present.

$$\text{residual} = \text{original value} - \text{predicted value} \tag{1}$$

CHILD's AGE	CHILD's IQ	RESIDUALS
10	20	-10
15	34	4
16	38	8

- So the average of 20, 34, and 38 is 30.67 for simplicity let's take it as 30. If we plot a graph keeping y-axis as IQ and x-axis as Age and then we can see the variance in points from the average mark.

  At first, our base model(M_0) will give a prediction 30

  We know this model(M_0) suffers a loss which will have some optimisation in the next model(M_1)

  Model M1 will have input as age(independent features) and target as the loss suffered(residuals) in M0. Until now it is the same as the gradient boosting technique.
  For XGboost some new terms are introduced,

$$\lambda \rightarrow \text{regularization parameter} \tag{2}$$
$$\gamma \rightarrow \text{auto tree pruning parameter} \tag{3}$$
$$\eta \rightarrow \text{how much model will converge} \tag{4}$$
$$\tag{5}$$

- Calculating the similarity score :
  This step can be written as :-

$$\text{similarity} = \frac{(\sum residual)^2}{N + \lambda} \tag{6}$$

  where, N is the number of residuals.
  Note: We are first summing up the residuals and then squaring the $\sum(residuals)$

  At first let's put $\lambda = 0$, then Similarity Score $= \frac{(-10 + 4 + 8)^2}{3 + 0} = 4/3 = 1.33$

  Let's make the decision tree using these residuals and similarity scores. I've set the tree splitting criteria as Age >10.

  Again for these two leaves, we calculate the similarity scores which is 100 and 72.

- After this, we calculate the **Gain** of splitting the residuals into 2 groups(to evaluate different thresholds)

$$\text{Gain} = \text{Similarity}_{parent} - \text{similarity}_{leftchild} - \text{similarity}_{rightchild} \tag{7}$$
$$\therefore \text{Gain} = 100 + 72 - 1.3 = 170.7 \tag{8}$$

- **Pruning trees:** Now we set our γ, which is a value provided to the model at starting and its used during splitting.

  If $Gain > \gamma$ then split will happen otherwise not. Let's assume that γ for this problem is 130 then since the gain is greater than 130, further split will occur. By this method, auto tree pruning will be achieved.

  The greater the γ value more pruning will be done.

  Note: For regularization and preventing overfitting, we must increase the λ which was initially set to 0. But this should be done carefully as greater the λ value lesser the Similarity score, lesser the gain and more the pruning.

- **The learning rate** η

$$\text{New prediction} = \text{Previous Prediction} + \text{Learning rate} * \text{Output} \tag{9}$$

XGboost calls the learning rate as eta and its value is set to 0.3

For the 2nd reading(Age=15) new prediction = 30 + (0.3 * 6) = 31.8

The outcome is 6 is calculated from the average residuals 4 and 8.

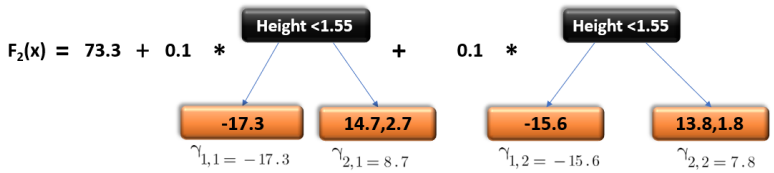
New Residual = 34 – 31.8 = 2.2

Age	IQ	Residual
10	20	-7
15	34	2.2
16	38	6.2

This way model M1 will be trained and residuals will keep on decreasing, which means the loss will be optimized in further models.

3 Some Questions

1. How do we predict the output for a new dataset?
Ans. I am taking a hypothetical example here just to make you understand how this predicts for a new dataset:



If a new data point says height = 1.40 comes, it'll go through all the trees and then will give the prediction. Here we have only 2 trees hence the datapoint will go through these 2 trees and the final output will be $F_2(x)$.

2.
Ans.

3.
Ans :
4.
Ans.
5. How can you evaluate the performance of a dimensionality reduction algorithm on your dataset?
Ans.