

XGBoost Extreme Gradient Boosting

Introduction

video [↗](#)

Performance

- Regularized learning objective
 - have in build regularization term in built
- Handling missing values
- Sparsity aware split finding
- Tree pruning
- Efficient split finding(weight quantile sketch + approximate tree learning)

parallel processing

(Boosting is a sequential process) so how they implemented parallel processing in it??

Ans -> all models are trained sequentially but parallel processing is implemented while building a model

Example ->

1 model is made using parallel processing, then 1 model mistakes pass on to another 2 model, then

2 model is made using parallel processing,

same goes on

optimized data structure

other algo store data row wise

Feature 1	Feature 2	Output Feature
1	2	3
2	3	4
3	4	5
4	5	6

Store data column wise, Feature1 = {1,...} or Feature2 = {1,...}

new data type used -> Column block

Cache awareness

out of core computing

Hyper parameter tree_method = 'hist'

Distributed computing

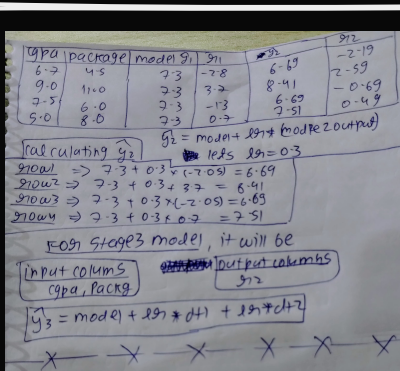
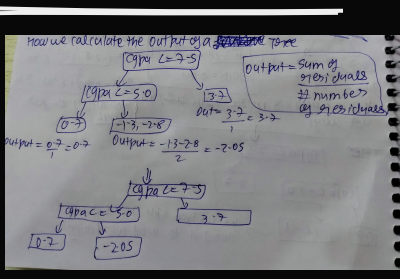
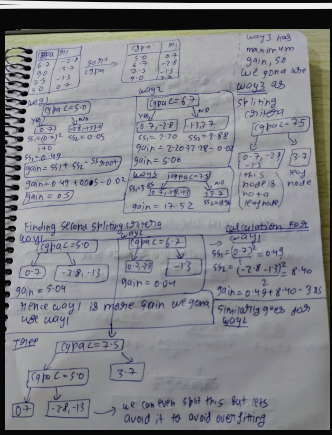
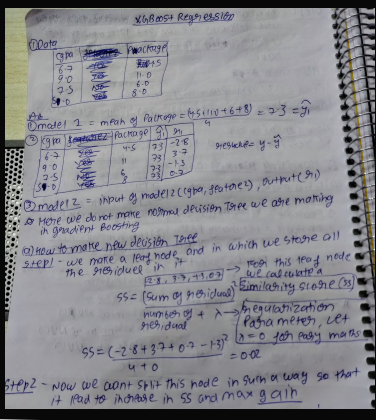
Gpu support Hyper parameter tree_method = 'gpu_hist'

cross platform
(Now most of algo are also cross platform)

Flexibility

- Support multiple programming languages
- Integration with other libraries and tools
- Support all kinds of ML problems

Regression



Classification

Theory

MATHs behind xgboost