Loss Function in the Extreme Gradient Boost

Ziren Xiao 675485 zirenx@student.unimelb.edu.au

Agenda

- Background & Motivation
- Problems & Challenges
- The New Loss Function
- Measuring Methods & Expectation
- Impacts

Background & Motivation

- What is eXtreme Gradient Boost (xgboost)?
- What is the loss function?
- Why do we need a good loss function in the xgboost?
- How to customize loss function in the xgboost?
  - Objective function in the xgboost
    - $Obj^t(q) = -\frac{1}{2}\sum_{j=1}^{T}\frac{(\sum_{i \in I_j} g_i)^2}{\sum_{i \in I_j} h_i + \gamma} + \delta T$

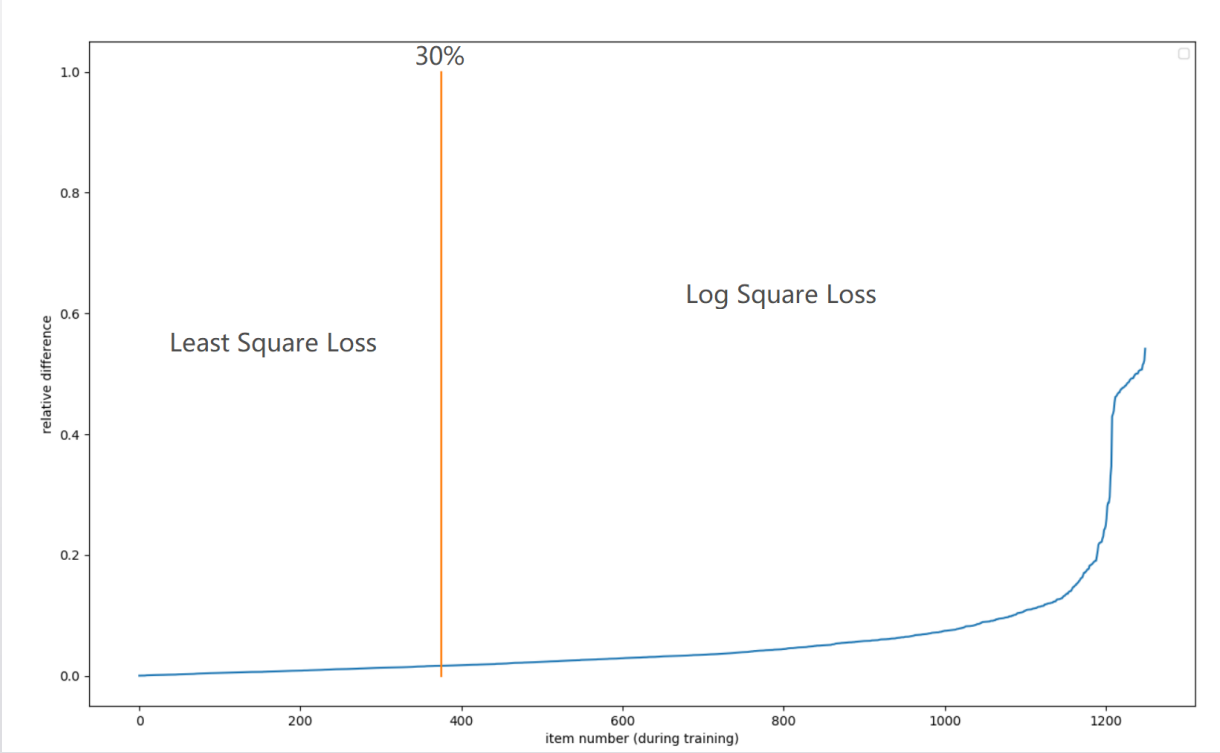
    where g and h are the first and the second derivative
  - Customization
    - Define g and h
- Goal: Find a loss function addressing the outlier problem

Current problems & Challenges

- Bad performance on outliers, good on normal points (e.g. Least Square Loss)
- Bad performance on normal points, but good on outliers (e.g. Log Square Loss)
- Loss functions must be second-order differentiable
  - Huber Loss
  - Absolute Loss
- Loss functions must produce positive numbers
  - Affect performance on training

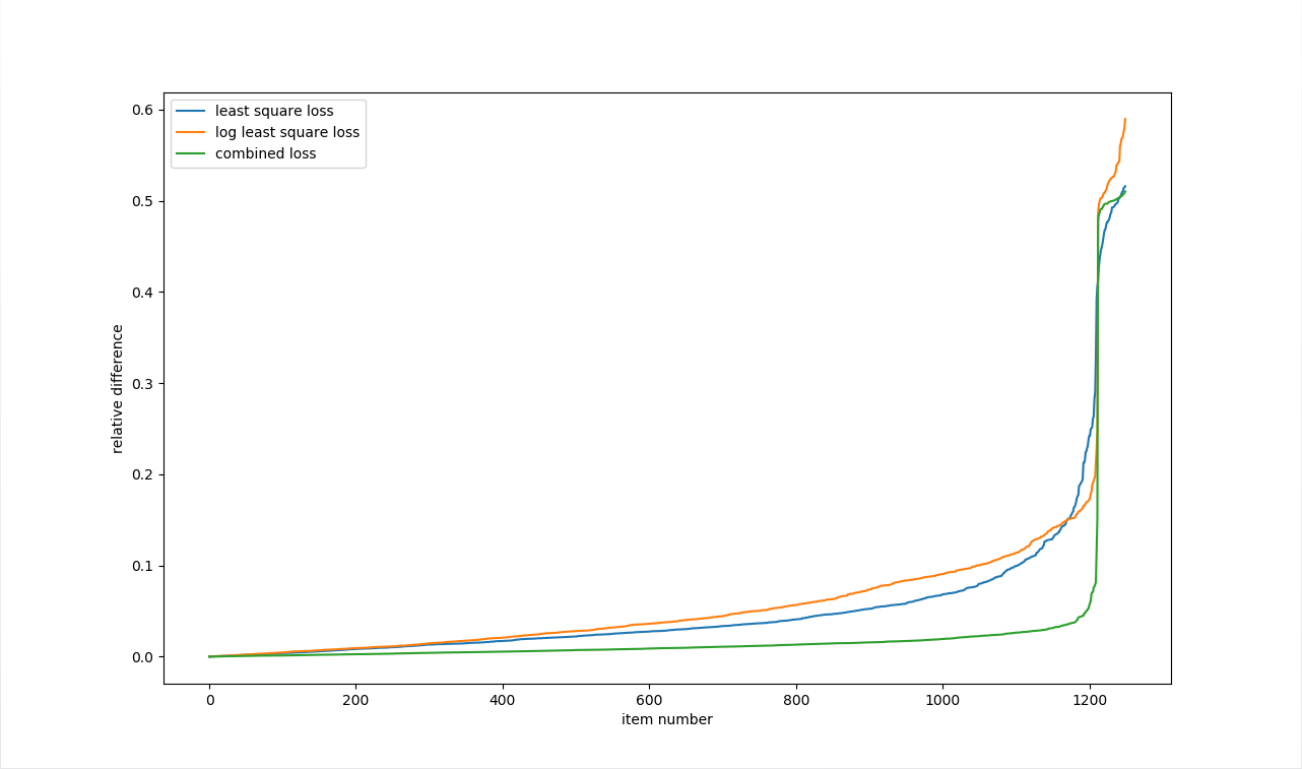
The New Loss Function

$$f(real, pred) = \begin{cases} Least\ Sqaure\ Loss\ (real, pred), & relative\ difference < 30\% \\ Log\ Sqaure\ Loss\ (real, pred), & otherwise \end{cases}$$



Expected outcomes

- Measure: Absolute relative difference
  - Relative Difference = $\left|\frac{\text{real} - prediction}{real}\right|$
- Dataset:
  - Generated dataset f = a0 + a1 + a2 + a3
    - Apply outliers
      - Probability p = 2% ~ 5% to set f = 2 * f
  - Real dataset
    - Kaggle (https://www.kaggle.com/)
- Expectation
  - better performance on both normal points and outliers



Expectation - example

Lower relative difference on Combined Loss

Impacts

- Using a single loss function to
  - Minimize error on normal points
  - Minimize error on outliers
  - Therefore increase the overall accuracy of predictions
- Provide a way to do further improvements
  - Replace better loss functions on both parts
- Drawback
  - More computation time during training