Gradient Boosting

Jan-Philipp Kolb

4 September 2018

Gradient boosting

Gradient boosting is a machine learning technique for regression and classification problems, which produces a prediction model in the form of an ensemble of weak prediction models, typically decision trees. It builds the model in a stage-wise fashion like other boosting methods do, and it generalizes them by allowing optimization of an arbitrary differentiable loss function.

The idea of gradient boosting originated in the observation by Leo Breiman that boosting can be interpreted as an optimization algorithm on a suitable cost function.

Breiman, L. (1997). "Arcing The Edge". Technical Report 486. Statistics Department, University of California, Berkeley.

Explicit regression gradient boosting algorithms were subsequently developed by Jerome H. Friedman,[2][3] simultaneously with the more general functional gradient boosting perspective of Llew Mason, Jonathan Baxter, Peter Bartlett and Marcus Frean.[4][5]

The latter two papers introduced the view of boosting algorithms as

Advantages of gradient boosting

- Often provides predictive accuracy that cannot be beat.
- Lots of flexibility - can optimize on different loss functions and provides several hyperparameter tuning options that make the function fit very flexible.
- No data pre-processing required - often works great with categorical and numerical values as is.
- Handles missing data - imputation not required.

Disadvantages of gradient boosting

- GBMs will continue improving to minimize all errors. This can overemphasize outliers and cause overfitting. Must use cross-validation to neutralize.
- Computationally expensive - GBMs often require many trees ($>$1000) which can be time and memory exhaustive.
- The high flexibility results in many parameters that interact and influence heavily the behavior of the approach (number of iterations, tree depth, regularization parameters, etc.). This requires a large grid search during tuning.
- Less interpretable although this is easily addressed with various tools (variable importance, partial dependence plots, LIME, etc.).

Two types of errors for tree methods

Bias related errors

- Adaptive boosting
- Gradient boosting

Variance related errors

- Bagging
- Random forest

Gradient Boosting for Linear Regression - why does it not work?

While learning about Gradient Boosting, I haven't heard about any constraints regarding the properties of a "weak classifier" that the method uses to build and ensemble model. However, I could not imagine an application of a GB that uses linear regression, and in fact when I've performed some tests - it doesn't work. I was testing the most standard approach with a gradient of sum of squared residuals and adding the subsequent models together.

The obvious problem is that the residuals from the first model are populated in such manner that there is really no regression line to fit anymore. My another observation is that a sum of subsequent linear regression models can be represented as a single regression model as well (adding all intercepts and corresponding coefficients) so I cannot imagine how that could ever improve the model. The last observation is that a linear regression (the most typical approach) is using sum of squared residuals as a loss function - the same one that GB is using.

Links

- **Gradient Boosting Machines**
- How to Visualize Gradient Boosting Decision Trees With XGBoost in Python