

Introduction

An intro to XGBoost and gradient boosted decision trees.

In the **Gradient Boosting** section, you will be learning about [XGBoost](#), a library for highly efficient gradient boosted decision trees. It is one of the premier libraries used in data science for classification and regression.

A. XGBoost vs. scikit-learn

The previous 3 sections of this course used scikit-learn for a variety of tasks. In this section, we cover XGBoost, a state-of-the-art data science library for performing classification and regression. XGBoost makes use of [gradient boosted decision trees](#), which provides better performance than regular decision trees.

In addition to the performance boost, XGBoost implements an extremely efficient version of gradient boosted trees. The XGBoost models train much faster than scikit-learn models, while still providing the same ease of use.

For data science and machine learning competitions that use small to medium sized datasets (e.g. [Kaggle](#)), XGBoost is always among the top performing models.

B. Gradient boosted trees

The problem with regular decision trees is that they are often not complex enough to capture the intricacies of many large datasets. We could continuously increase the maximum depth of a decision tree to fit larger datasets, but decision trees with many nodes tend to overfit the data.

Instead, we make use of gradient boosting to *combine* many decision trees into a single model for classification or regression. Gradient boosting starts off with a single decision tree, then iteratively adds more decision trees to the overall model to correct the model's errors on the training dataset.

The XGBoost API handles the gradient boosting process for us, which produces a much better model than if we had used a single decision tree.