Machine Learning Techniques

DATASCI 420

Lesson 06-02 Gradient Boosted Decision Trees

Gradient Boosted Decision Trees

- An ensemble model
  - Ensemble of decision trees as base learners

- A powerful supervised machine learning model

- Applies to regression, classification, and ranking problems

- Won Yahoo Learning to Rank Challenge (Track 1)

AdaBoost (Adaptive Boosting)

- Boosting is a powerful technique for combining multiple “base” learners to produce a form of committee whose performance can be significantly better than that of the base learners.

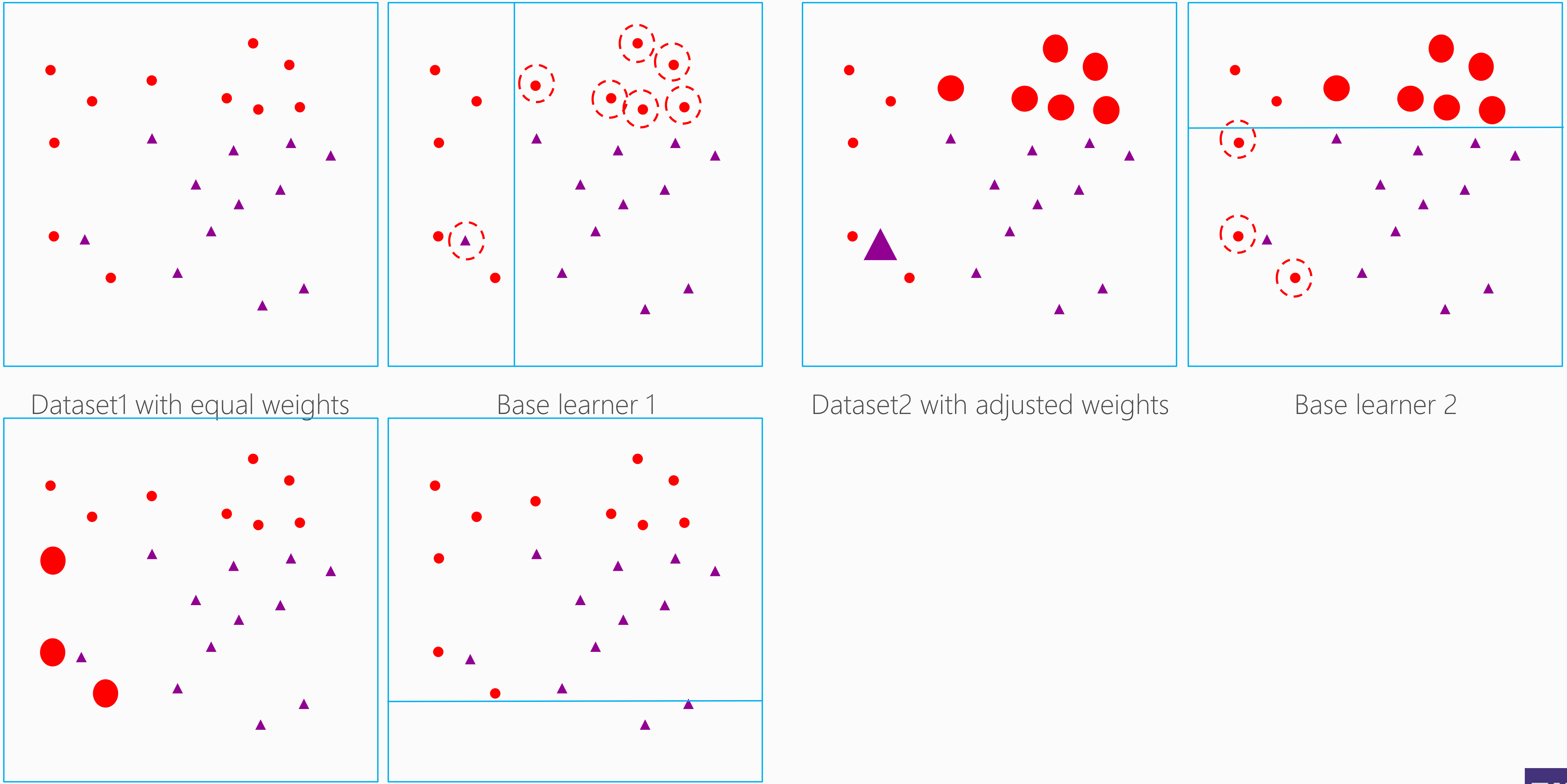
- Boosting and Bagging
  - In bagging, every base learner is trained on a random sample from the original dataset which is independent to the training set of other base learners.
  - In boosting, base learner i+1 is trained on a random sample which is dependent on the previous base learners.

- AdaBoost is a widely used form of boosting algorithm

How AdaBoost Works?

- Step 0: initialize the weight for each observation to be 1/n, n=# of observations
- Step 1: for m = 1, …, M
  - Train a classifier to minimize weighted classification error. When m = 1, the weight of each observation is initialized in Step 0.
  - Increase the weights of observations that are misclassified by the current classifier
- Step 2: the final prediction is the weighted average of all M classifiers

Example of AdaBoost



Dataset1 with equal weights

Base learner 1

Dataset2 with adjusted weights

Base learner 2

Dataset3 with equal weights

Base learner 3

Gradient Boosted Decision Trees

- AdaBoost is using the entire training data to fit the original target variable Y for each tree, where each observation has different weight
- Gradient boosted decision trees is using the entire training data to fit the residuals of the target variable Y from previously trained models

$$F^{t-1}(x_i) + h^t(x_i) = y_i, i = 1, 2, \ldots, \mathrm{n}$$

- If we call $y_i - F^{t-1}(x_i)$ as the residual of the previous t-1 trees, $h^t(x_i)$ is a regression tree for the residuals, with the training data like [(x_1, y_1- $F^{t-1}(x_1)$), (x_2, y_2- $F^{t-1}(x_2)$), …, (x_n, y_n- $F^{t-1}(x_n)$)]
- Final prediction will be, where ρ is named the shrinkage rate (learning speed):

$$\boldsymbol{F}^{t}(\boldsymbol{x}_i) = \sum_{\boldsymbol{k}=\mathbf{0}}^{\boldsymbol{t}} \rho \boldsymbol{h}^{\boldsymbol{k}}(\boldsymbol{x}_i) = \boldsymbol{F}^{t-1}(\boldsymbol{x}_i) + \rho h^t(x_i)$$

Advantages and Disadvantages of Gradient Boosted Decision Trees

- Advantages:
  - Can be more accurate than adaboost and random forest

- Disadvantages:
  - More trees can bring severe overfitting, since each additional tree is fitting on the residuals
  - Not easy to parallelize since tree t+1 is depending on the residuals from the previous trees

Summary

- Introduced Adaboost and Gradient Boosted Decision Trees
- Practices Gradient Boosted Decision Trees in Python