**-Decision tree**

**-random forest:** decision tree+bagging+randomness

**-GBDT, gradient boosting decision tree:** boosting decision tree+gradient boosting

BDT: Each tree in random forest is not related to each other (classification tree), while in GBDT, trees are related to each other(regression tree).

Each tree aims to fit the residual of the previous tree.(For example, we are going to predict a person’s age, suppose that in training data, A’s age is 30, the first tree’s learning result is to categorize A into the leaf node with age od 27, then residue is 30-27=3. The goal of the second tree is to fit the result of 3.) When we get k trees after training, we need to predict the score of a sample. In fact, according to the features of this sample, each feature will match a corresponding leaf node, and each leaf node corresponds to a score. Finally, just add the scores corresponding to each tree together, then we get the predicted value of the sample.

The algorithm ends when the number of trees reach the parameter we set in the model.

GB: GBDT use gradient to decide the direction of optimization

**-XGBOOST, eXtreme Gradient Boosting:** GBDT+regularization (A regular term is added to the objective function of each iteration to further reduce the risk of overfitting.)

So, it can be said that, based on random forest, XGBOOST makes improvement in 1) can solve regression problems 2) add regular term to further reduce overfitting