Ensemble Learning

Author: Zhang Xiaozheng Data:2019/10/20

When a teacher draw a question on the class, you solve it and write down your answer, which may be right or wrong. But what if let all your classmates to vote and then get the final answer? Intuitively, this answer is more likely to be the correct answer.

So ensemble learning algorithm just work like this. We train lots of individual learner and then combine them with some strategy. In our first example, we just choose the most answer that "individual learners" get.

But there are some problems. First look at the forms below. (Suppose the right answers of three test data are all √)

Individual learner	test data 1	test data 2	test data 3
h1	√	√	×
h2	×	√	√
h3	√	×	√
Ensemble	√	√	√

Individual learner	test data 1	test data 2	test data 3
h1	√	√	×
h2	√	√	×
h3	√	√	×
Ensemble	√	√	×

Individual learner	test data 1	test data 2	test data 3
h1	√	×	×
h2	×	√	×
h3	×	×	√
Ensemble	×	×	×

So from these forms, we can see:

1. If all of individual learners are same, we don't need to use the ensemble learning algorithm.
2. If just a little individual learners can give the right answer, we will still get a wrong answer.

To make the ensemble learning works well, the individual learners need to be more accurate and different. But the individual learners are trained from a same data set, if we split data set into many parts to train the individual learners separately, every individual learner may be bad because of lack of train data. If we train the individual learners using a same, large data set, these

individual learners may be exactly same as each other.

To solve these problems, there are two algorithm, which are Bagging and Random Forest.

Bagging

We hope use different training sets to train individual learners, at the same time we also hope these training sets are large enough. So we can collect our training sets by using bootstrap sampling. Then we these training sets to train our individual learners.

Random Forest

On the basis of Bagging algorithm, we choose Decision Tree as the individual learner. Besides, while training the decision tree, we don't choose the best feature to split the train data in all available features. Instead, we random choose ***k*** features and then choose the best features in these ***k*** features. This make individual learners more different.