Bagging

Oliver Zhao

1 Description

Bagging is when many classifiers - usually decision trees - are created and used in parallel, resulting in many outputs (predictions). Then, an ensemble classifier receives all of the individual predictions as inputs to output a final classification. The simplest way of aggregating the individual predictions is to simply count the number of "votes" among the individual classifiers and make the final classification based on which of the classes received more votes.

In order to generate each of the classifiers, n subsets $T_1, \ldots, T_n$ are generated by sampling the same amount of data points from the training set T with replacement. A classifier C_i is trained from a particular subset T_i. A subtle but distinct difference between the Decision Trees used for Bagging compared to Random Forests is that all features are considered at each node for a split. In contrast, only a subset of features are considered for each node for split in Random Forests. The central idea is that averaging all of the predictions from different trees results in a better overall final prediction.

2 Algorithm

Description: Consider a training set T split into n training subsets $T_1, \ldots, T_n$.

---

1. Using random sampling with replacement, generate n training subsets $T_1, \ldots, T_n$.
2. For each training subset T_i, generate a classifier C_i. See the algorithm for Decision Trees for how each classifier is generated.
3. Find the predictions of each classifier $C_1, \ldots, C_n$.
4. The master classifier determines which class received the most votes, to determine the final output.