Aggregating Regressive Estimators: Gradient-Based Neural Network Ensemble

Jiang Meng¹ and Kun An²

¹ School of Mechanical Engineering and Automatization, North University of China, Taiyuan 030051, Shanxi, China
rivermeng@gmail.com
² School of Information and Communication Engineering, North University of China, Taiyuan 030051, Shanxi, China
ankun06@gmail.com

Abstract. A gradient-based algorithm for ensemble weights modification is presented and applied on the regression tasks. Simulation results show that this method can produce an estimator ensemble with better generalization than those of bagging and single neural network. The method can not only have a similar function to GASEN of selecting many subnets from all trained networks, but also be of better performance than GASEN, bagging and best individual of regressive estimators.

1 Introduction

There is a growing realization that combinations of estimators can be more effective than the single estimator. Neural network as a single estimator for regression tasks, can be used to approximate any continuous function at any precision according to the characteristics of nonlinear and learnability. Another characteristic of generalization is more important to the regression tasks. However, the generalization performance of neural network is not so good, for there is overfitting [1] during the course of network training. Aimed at the situation, an idea of combining many neural networks is presented in 1990s, which is called neural network ensemble [2]. The ensemble can improve the generalization effectively and easily by aggregating the outputs of single networks without additional complicated operations.

Taking neural networks as component estimators, the output of the ensemble is either weighted-averaging or simple-averaging for regression problems. The favorite algorithms to train component networks (named subnets) are bagging and boosting. Based on the bootstrap sampling, bagging [3] is used to train networks independently with random subsets generated from original training set, as might improve ensemble generalization because of different training subsets for different subnets. Boosting [4] can produce a series of subnets sequentially, whose training sets are determined by the performance of former ones. Training instances that are wrongly predicted by the former subnets will play more important roles during the training course of later subnets. Because boosting and revised version AdaBoost [5][6] both focus on only classification problems, AdaBoost.R2 [7], as the modification of AdaBoost.R [8], big error margin (BEM) [9] and AdaBoost.RT [10] are proposed respectively to solve regression problems last decade.
Two important issues of an ensemble are to decide how to train the subnets and how to combine the subnets. For the first issue, there are roughly three major approaches [11] to training subnets, i.e., independent training, sequential training and simultaneous training. In independent training, each subnet in the ensemble is trained independently to minimize the error between the target and its output, such as bagging; in sequential training, the ensemble subnets are trained one after one not only to minimize the error between targets and outputs, but also to decorrelate their errors from previously trained networks, such as boosting series algorithms; in simultaneous training, ensemble subnets are trained simultaneously and interactively, with a penalty item in the error function interacting on the ensemble subnets such as CELS [12], or alternating between training step and observational step such as OLA [13]. For the second issue, the combination of subnets is divided into simple averaging [1] and weighted averaging [14]. The former combines the subnet outputs in the way of equal-weighted average, while the latter sums unequal-weighted outputs of ensemble subnets.

Discussing from both theories and simulations, Zhou proposed an idea of "many selected better than all combined" [15], which means that ensembling many of available subnets may has better generalization than ensembling all of subnets by calculating the correlations among the ensemble subnets. Based on this idea, Zhou gave GASEN [16] to select many subnets from all the subnets using the genetic algorithm. On one hand, GASEN proves theoretically there are unequal-weighted distributions for ensemble weights with smaller generalization error than the equal-weighted one, and provides a feasible way to achieve one solution using the heuristic optimization algorithm; on the other hand, it still adopts simple averaging technique to combine a new ensemble with solution of selected subnets. Therefore, GASEN can be regarded as a tradeoff between simple averaging and weighted averaging. It seems that there should be more reasonable weight distributions with better generalization, if modifying ensemble weights in an appropriate strategy for optimization. In this paper, a subnet weight modification algorithm (subWMA) is presented to optimize ensemble weights in the strategy of gradient optimization.

2 Analysis of Bias-Variance Decomposition

2.1 Bias-Variance Decomposition in Neural Network

The regression problem for neural network is to construct a function $f(x)$ based on a training set of $(x_1, y_1), \ldots, (x_P, y_P)$, for the purpose of approximating $y$ at future observations of $x$, which is called generalization. To be explicit about the dependence of $f$ on the data $D = \{(x_k, y_k)\}_{k=1}^P$, we can rewrite $f(x)$ to $f(x; D)$. Given $D$ and a particular $x$, the cost function $\xi$ of the neural network is

$$\xi = \frac{1}{2} \sum_{k=1}^{P} (y_k - f(x_k; D))^2 = \frac{1}{2} E[(y - f(x; D))^2 | x, D]$$

where $E[\cdot]$ means the expectation of data set $D$.

From Reference [17], Eq. (1) can be written as the following equation,