New Ensemble Combination Scheme

Namhyoung Kim, Youngdoo Son, and Jaewook Lee, Member, IEEE

Abstract—Recently many statistical learning techniques are successfully developed and used in several areas. However, these algorithms sometimes are not robust and does not show good performances. The ensemble method can solve these problems. It is known that the ensemble learning sometimes improves the generalized performance of machine learning tasks as well as makes it robust. However, the combining weights of the ensemble model are usually pre-determined or determined with the concept that the ensemble model is a superposition of individual ones. Thus we proposed a new ensemble combination scheme which consider the ensemble model is a factor affects the individual predictors. Through experiments, the proposed method shows better performance than other existing methods in the regression problems and shows competitive performance in the classification problems.

I. INTRODUCTION

Several learning models such as clustering, classification, regression, etc., are successfully developed in these days and they are used in many areas [1], [2], [3], [4]. However, sometimes these algorithms show ill performances or are not robust. Ensemble can be a solution to overcome these problems. Ensemble combines multiple models together in specific ways. The ensemble learning is known to improve the generalized performance of a single predictor and it has won some success. For example, the famous one million dollars Netflix Prize competition, the team named The Ensemble won the first place. They were known to ensemble several algorithms. In other data mining competitions, most winning teams employed an ensemble of learning models. Also, the ensemble method has been widely used in several practical area [5], [6], [7].

Along with these practical successes, many different ensemble algorithms have been proposed and developed [3], [8], [9]. However, some of them use the pre-determined weights. Most of the rest find weights with the concept that the ensemble model is a superposition of individual predictors. Unlike existing ensemble models, we propose a new ensemble combination scheme which consider individual predictors as realizations of the ensemble model. In other words, in the view of factor analysis, the ensemble model becomes a factor which affects individual models.

The organization of this paper is as follows. In the next section we summarize related literature and algorithms. In Section 3 we describe the proposed method. Next we present the results of an empirical analysis and comparison with other ensemble methods in Section 4. Then we conclude in Section 5.

II. LITERATURE REVIEW

Most of the contemporary ensemble methods develop ensemble model based on the following equation.

$$f = \sum_{i=1}^{m} w_i C_i$$

where $f$ is a real function predicted, $C_i$ is an individual base model and $w_i$ is weight. It is illustrated in Figure 1, that is, the predictions of the individual models $C_i, i = 1, 2, ..., m$, are combined using the weight $w_i$. The weights can vary depending on combination scheme. According to combination scheme, ensembles are divided into two main classes, combine by learning and combine by consensus. Boosting combines multiple base predictors by learning. It utilizes the information of the performance of previous predictors so this kind of ensemble is also known as supervised scheme. In bagging, the ensemble is formed by majority voting. Bagging doesn’t consider the performance of each predictor. It averages the predictions of individual models in unsupervised scheme.

A. Bagging

Bagging is the simplest algorithm to construct an ensemble[10]. It creates individual predictors by training randomly selected training set. Each training set is generated by randomly with replacement, $n$ examples. In bagging, the weights are uniform, i.e., the ensemble prediction is given by

$$f = \frac{1}{m} \sum_{i=1}^{m} C_i.$$  \hspace{1cm} (2)

In the case of classification, majority voting is used to predict the class of data.

B. Boosting

Boosting algorithms learn iteratively weak classifiers using a training set selected according to the previous classified results then combine them with different weights to create an ensemble[12]. The weights are determined by the weak learners’ performance.

Our method uses both supervised and unsupervised scheme. In the following section, we describe our proposed method.

III. PROPOSED METHOD

In this paper, we propose a new ensemble combination scheme. We assume that the prediction of each individual predictor $C_i$ reflects the real function value $f$ by $\lambda_i$ and a specific error term $\delta_i$. In equation form, we would write the following.
We can rewrite the above equation in matrix form as following.

\[ C = fA' + \Delta \]

where

\[ C = \begin{bmatrix} C_{11} & C_{12} & \cdots & C_{1n} \\ C_{21} & C_{22} & \cdots & C_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ C_{n1} & C_{n2} & \cdots & C_{nn} \end{bmatrix}, \quad f = \begin{bmatrix} f(1) \\ f(2) \\ \vdots \\ f(n) \end{bmatrix}, \quad A = [\lambda_1 \lambda_2 \cdots \lambda_n]^T, \quad \text{and} \quad \Delta = \begin{bmatrix} \delta^{(1)} \\ \delta^{(2)} \\ \vdots \\ \delta^{(n)} \end{bmatrix} \]

\( n \) is the number of cases of data set. In addition, we have the following three assumptions about the proposed method.

**Assumption 1:**
1) Variance of \( f \) is 1.

\[ (n-1)f'f = 1 \]  

2) Specific factors \( \delta \) are mutually uncorrelated with diagonal covariance matrix:

\[ \Theta = \frac{1}{n-1} \Delta \Delta = diag(\theta_1^2, \theta_2^2, \ldots, \theta_n^2). \]

3) Common factor \( f \) and specific factors \( \delta \) are uncorrelated.

The approximation of the correlation matrix \( S \) is

\[ S = 1/(n-1)CC' = 1/(n-1)(fA' + \Delta)'(fA' + \Delta) = 1/(n-1)(Af + \Delta)'(Af + \Delta) \]

By assumptions, the correlation matrix becomes simplified as following

\[ S = AA' + \Theta \]

ML solution: \( S(AA' + \Theta)^{-1}A = A \)

\[ f_i = A(AA' + \Theta)^{-1}C_i \] or \( f_i = (1 + \Lambda' \Theta^{-1}A)^{-1}A^{\top}C_i. \)

Let\( f_i^* = \Lambda' (\Lambda' \Theta^{-1}A)^{-1}C_i + \epsilon \) and \( C_i = \Lambda f_i^* \)

Then \( \Lambda f_i^* = C_i \) and \( \Lambda (\Lambda' \Theta^{-1}A)^{-1}C_i + \Lambda \epsilon \)

\[ A' \Lambda \epsilon = A' [I_p - \Lambda (\Lambda' \Theta^{-1}A)^{-1}]C_i \]

\[ \epsilon_i = A' [I_p - \Lambda (\Lambda' \Theta^{-1}A)^{-1}]C_i \]

Bayesian Latent Ensemble

\[ f_i^* = \frac{\Lambda' (\Lambda' \Theta^{-1}A)^{-1}C_i}{1 + \sum_{l=1}^m \lambda_l^2 \delta_l^2} \]

\[ f^* = \sum_{k=1}^m \frac{\lambda_k^2 \delta_k^2}{1 + \sum_{l=1}^m \lambda_l^2 \delta_l^2} C_k + \epsilon \]

In supervised scheme, we assume the given output of training sets as real function value. Then, the process becomes much simpler.

**IV. EXPERIMENTAL RESULTS**

**A. Data Sets**

To evaluate the performance of the proposed method, we use both regression and classification data sets from UCI Machine Learning Repository[11]. Table I shows the summary of the data sets used. The data shown in Table I is widely used real-world problems in the machine learning...
The data sets vary in terms of both the number of cases and features.

### B. Experiments settings

In this study, neural networks are used as a base model. Neural networks consist of a series of input, hidden and output layers. It is needed to set the size of hidden layer. Some parameters are also required for the neural networks. We trained the neural networks with a backpropagation function, a learning rate of 0.15, a momentum constant of 0.9 and randomly selected weights between -0.5 and 0.5. The size of hidden layer was determined according to the number of input and output units having five hidden units as a minimum. The number of epochs was based on the number of cases. We used a larger value of epochs as the size of data increases. The training parameters used in the experiments are shown in Table I.

For comparison, other popular ensemble algorithms, bagging and boosting, and single model are applied to the same data sets. We averaged test error of each method over five standard 10-fold cross validation experiments. The data set is divided into 10 equal-sized sets, then each set is used as the test set in turns then the other nine sets are used for training. In all ensemble methods we used an ensemble size of 25 for each fold.

### C. Results

Table II and Table III show the classification error for classification problems and root mean squared error for regression problems, respectively. The columns indicate the different methods, a single model(Single), Bagging(Bagging), Arcing(Arc), Ada-boosting(Ada) and the proposed method(FA). Each row shows the results for each data set. From the tables we can notice that the proposed method usually shows good performances for several data sets.

In classification problems(Table II), the proposed method
does not show the worst performance and its results are often the best or close to the best. In regression problems (Table III), the proposed method shows the best performances except for the housing_boston data set. In that data set, bagging algorithm performs the best but the proposed method showed competitive result with the bagging. Therefore, we can say that our proposed method performs better than other existing ensemble methods in general.

V. CONCLUSION

In this study, we presented novel methods for ensemble combination scheme with a new perspective on a relationship between real function and each classifier. We assumed predicted value of each classifier is composed of the real function value and a specific error term. According to this assumption, we solved weights for ensemble combination in both supervised and unsupervised scheme. The experimental results show the proposed methods are superior to the existing methods. They also show self-correction skills, so an ensemble can adjust weights noticing corrupt classifiers in learning process.

The proposed ensemble method seems to be robust with some corrupting predictors unlike other ensemble schemes whose performances may decrease with corrupting predictors. In other words, the proposed method has self-correction reliability. Thus, for future work, the analogous proof and experimental results for this intuition are necessary. We also need to prove and show the statistical stability of the proposed model using statistical tests like Friedman test, $t$-test, or sign-test.

REFERENCES

## TABLE II
### Classification results (classification error)

<table>
<thead>
<tr>
<th>Data set</th>
<th>Neural Network</th>
<th>Boosting</th>
<th>FA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Single</td>
<td>Bagging</td>
<td>Arc</td>
</tr>
<tr>
<td>Breast-cancer-w</td>
<td>3.59%</td>
<td>3.15%</td>
<td>3.35%</td>
</tr>
<tr>
<td>credit-a</td>
<td>14.17%</td>
<td>13.94%</td>
<td>15.04%</td>
</tr>
<tr>
<td>ionosphere</td>
<td>12.91%</td>
<td>11.71%</td>
<td>10.17%</td>
</tr>
<tr>
<td>statlog_aust</td>
<td>15.48%</td>
<td>13.83%</td>
<td>15.22%</td>
</tr>
<tr>
<td>statlog_german</td>
<td>26.66%</td>
<td>24.68%</td>
<td>27.48%</td>
</tr>
<tr>
<td>hepatitis</td>
<td>36.80%</td>
<td>34.13%</td>
<td>36.40%</td>
</tr>
<tr>
<td>sonar</td>
<td>17.40%</td>
<td>16.60%</td>
<td>14.40%</td>
</tr>
<tr>
<td>diabetes</td>
<td>24.16%</td>
<td>23.21%</td>
<td>24.82%</td>
</tr>
</tbody>
</table>

## TABLE III
### Regression results (RMSE)

<table>
<thead>
<tr>
<th>Data set</th>
<th>Neural Network</th>
<th>FA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Single</td>
<td>Bagging</td>
</tr>
<tr>
<td>Abalone</td>
<td>2.0989</td>
<td>2.0733</td>
</tr>
<tr>
<td>housing_boston</td>
<td>3.8296</td>
<td>3.2316</td>
</tr>
<tr>
<td>ailerons</td>
<td>0.0005</td>
<td>0.0003</td>
</tr>
<tr>
<td>pole telecom</td>
<td>13.6437</td>
<td>10.5535</td>
</tr>
</tbody>
</table>