Journal of the Korean Data & Information Science Society
2012, **23**(2), 375–383

한국데이터정보과학회지

Study on the ensemble methods with kernel ridge regression

Sunhwa Kim[1] · Daehyeon Cho[2] · Kyung Ha Seok[3]

[1]Department of Statistics, Pusan National University
[23]Department of Data Science, Inje University

Received 31 January 2012, revised 16 February 2012, accepted 24 February 2012

Abstract

The purpose of the ensemble methods is to increase the accuracy of prediction through combining many classifiers. According to recent studies, it is proved that random forests and forward stagewise regression have good accuracies in classification problems. However they have great prediction error in separation boundary points because they used decision tree as a base learner. In this study, we use the kernel ridge regression instead of the decision trees in random forests and boosting. The usefulness of our proposed ensemble methods was shown by the simulation results of the prostate cancer and the Boston housing data.

Keywords: Boosting, ensemble method, forward stagewise regression, kernel ridge regression, random forest.

1. Introduction

Ensemble methods are learning algorithms using a set of classifiers to predict a new data's response value by taking a (weighted) vote or averaging of their predictions in the set of classifiers. The first ensemble method is the one using Bayesian averaging. Also there are another three famous ensemble methods such as bagging (Breiman, 1996), boosting (Freund and Schapire, 1997) and random forests (RF) (Breiman, 2001), which are based on the decision trees. Especially, boosting and random forests are well known to be excellent in the accuracy of prediction in classification problems.

Bagging is a technique reducing the variance of an estimated prediction function and seems to work well expecially for high-variance and low-bias procedures, such as in the decision tree. For regression, we simply fit the same regression trees many times to bootstrap sampled versions of the training data, to average the results. For classification, we cast a vote for the predicted class in a set of trees.

[1] Ph.D student, Department of Statistics, Pusan National University, Busan 609-735, Korea.
[2] Corresponding author: Professor, Department of Data Science, Institute of statistical Information, Inje University, Kimhae 621-749. Korea. E-mail: statcho@inje.ac.kr
[3] Professor, Department of Data Science, Institute of statistical Information, Inje University, Kimhae 621-749. Korea.

Boosting was initially proposed as a committee method as well, although unlike bagging, the committee of weak learners evolves over time, and the members cast a weighted vote. Boosting appears to dominate bagging on most of problems.

The least absolute shrinkage and selection operator (LASSO) proposed by Tibshirani (1996) minimizes the residual sum of squares subjects to the sum of the absolute value of the coefficients being less than a constant. It tends to produce some coefficients that are 0 and hence gives interpretable models because of the nature of the constraint. It enjoys some of the favorable properties of both subset selection and ridge regression.

According to Hastie *et al.* (2007), while LASSO makes optimal progress in terms of reducing the residual sum of squares per unit increase in L1-norm of the coefficient, forward stagewise regression (FSR) is optimal per unit L1 arc-length traveled along the coefficient path. They compared the LASSO and FSR procedures in a simulation study involving a large number of correlated predictors and showed the efficiency of the FSR. It is known that FSR with infinitesimally small step size produces a set of solutions which coincides with the set of the LASSO solutions for some special cases (Efron *et al.*, 2004).

RF algorithm is known to be the best one among the classification ensemble methods which can classify the large amount of the data with great accuracy and is a substantial modification of bagging which builds a large collection of de-correlated trees and then averages them. According to recent studies, it is proved that RF and FSR havegood accuracies of prediction in regression problem. However the prediction error has probably increased in a boundary point of the separation because continuous variables are considered as discrete ones.

In this study, we propose to use kernel ridge regression (KRR) as a base learner instead of decision tree in RF and FSR. We see that the mean squared prediction errors obtained from the proposed method are relatively smaller than those from FSR and RF for the prostate cancer and Boston housing data in regression problems.

From the experiment with the prostate cancer and the Boston housing data, we show the usefulness of our proposed ensemble methods. The rest of this paper is organized as follows. In section 2 simple reviews of FSR and RF are given. In section 3 we present our two proposed methods. In section 4 we perform numerical studies through examples. In section 5 we give the conclusions.

2. Preliminaries

2.1. Forward stagewise regression

For a given training data, $D = \{(x_i, y_i)\}_{i=1}^{N}$, $\boldsymbol{x}_i \in X \subset R^d$, $y_i \in R$, we are interested in the following linear model.

$$f(x) = \sum_{j=1}^{J} \alpha_j T_j(x),$$

where $J = card(\mathrm{T})$ and $T = \{\mathrm{T_j}\}$ are set of all possible terminal node regression trees. Since the number of such trees is likely to be much larger than even the largest training data sets,

some form of regularization is required. Let $\widehat{\alpha}(\lambda)$ solve

$$\min_{\alpha}\left\{\sum_{i=1}^{N}\left(y_i-\sum_{j=1}^{J}\alpha_j T_j(x_i)\right)^2+\lambda\cdot P(\alpha)\right\},$$

where $P(\alpha)$ is a function of the coefficients that generally penalizes larger values and $\lambda\geq 0$ is a penalty parameter. For example, if $P(\alpha)=\sum_{j=1}^{J}\alpha_j^2$, the solution of α is the same that in ridge regression, and if $P(\alpha)=\sum_{j=1}^{J}|\alpha_j|$, the solution of α is the same that in LASSO.

The algorithm for FSR to estimate the coefficients of regression model is as follows:

1) Initialize $\widehat{a}_j=0, j=1,\cdots,J$. Set $\varepsilon>0$ to some small constant, and M large.

2) For $m=1$ to M :

   a) $(\beta^*,j^*)=\arg\min_{\beta,j}\sum_{i=1}^{N}\left(y_i-\sum_{l=1}^{J}a_l T_j(x_i)-\beta T_j(x_i)\right)^2$.

   b) $\widehat{a}_{j^*}\leftarrow\widehat{a}_{j^*}+\varepsilon\cdot sign(\beta^*)$.

3) Output $f_M(x)=\sum_{j=1}^{J}\widehat{a}_j T_j(x)$.

Although the algorithm is phrased in terms of tree basis functions T_j, itcan be used with any set of basis functions.

2.2. Random forests

Random forests is a substantial modification of bagging that builds a large collection of de-correlated trees and then averages them. The algorithm of random forests for regression or classification is as follows:

1) For $b=1$ to B :

   a) Draw a bootstrap sample Z^* of size N from the training data.

   b) Grow a random-forest tree T_b to the bootstrapped data, by recursivelyrepeating the following steps for each terminal node of the tree, until theminimum node size $n_{\min}$ is reached.

   i. Select m variables at random from the p variables.

   ii. Pick the best variable/split-point among the m

   iii. Split the node into two daughter nodes.

2) Output the ensemble of trees $\{T_b\}_1^B$.

To make a prediction at a new point x :

Regression: $\widehat{f}_{rf}^B(x)=\dfrac{1}{B}\sum_{b=1}^{B}T_b(x)$

Classification: Let $\widehat{C}_b(x)$ be the class prediction of the bth random-forest tree. Then $\widehat{C}_{rf}^B(x) = majority\ vote\left\{\widehat{C}_b(x)\right\}_1^B$.

For classification, the default value for m is $\sqrt{p}$ and for regression, the default value for m is $p/3$. In practice the best values for these parameters will depend on the problem, and they should be treated as tuning parameters.

3. Proposed methods

3.1. Forward stagewise kernel ridge regression

Forward stagewise kernel ridge regression (FSKRR) uses kernel ridge regression(KRR) instead of trees in FSR. In KRR, we used radial basis function

$$k(x_i, x_j) = e^{-\sigma||x_i - x_j||^2},$$

where σ is a kernel parameter controlling the sensitivity of the kernel function. In other words, $K_j\gamma_j$ were made instead of T_j in algorithm of FSR, where $(K_j^T K_j + \lambda_j I)^{-1} K_j^T y = \gamma_j$ and K_j is $N \times N$ matrix with $(i,q)th$ element $k(x_i, x_q) = e^{-\sigma_j||x_i - x_q||^2}$ in KRR considering all possible kernel parameter σ and penalty parameter λ.

The algorithm for FSKRR is as follows:

1) Divide training data into two data sets and construct appropriate J sets of σ and λ from KRR method.
2) Initialize $\widehat{a}_j = 0$, $j = 1, \cdots, J$. Set $\varepsilon > 0$ to some small constant, and M large.
3) For $m = 1$ to M :
   a) $(\beta^*, j^*) = \arg\min_{\beta,j} \sum_{i=1}^N \left(y_i - \sum_{l=1}^J a_l f_l(x_i) - \beta f_j(x_i)\right)^2$,
   where $f_j(x) = K_j\gamma_j, j = 1, \cdots, J$, and $\gamma_j = (K_j^T K_j + \lambda_j I)^{-1} K_j^T y$.
   b) $\widehat{a}_{j^*} \leftarrow \widehat{a}_{j^*} + \varepsilon \cdot sign(\beta^*)$.
4) Output $f_M(x) = \sum_{j=1}^J \widehat{a}_j f_j(x)$.

The important part in this algorithm is to divide the training data into two datasets to avoid overfitting.And we calculate $f_j(x)$'s using one data set and we select a β which minimizes the formula 3)-a) using the other remaining data set. If the previous step does not be carried out, overfitting problem might happeneven if (σ, λ) was chosen properly.

Differentiating the equation 3)-a) in the above algorithm, we can find the β value that minimizes the equation. The result is as follows.

$$\widehat{\beta} = \frac{\sum_{i=1}^N \left(y_i - \sum_{l=1}^J a_l f_l(x_i)\right) f_j(x_i)}{\sum_{i=1}^N f_j(x_i)^2}.$$

Calculate all $\widehat{\beta}'$ s for every $K_j\gamma_j$, $j = 1, 2, ..., J$, and find $K_{j^*}\gamma_{j^*}$ value that minimizes the value of 3)-a) and get $f_M(x)$. If the predicted value does not change after some iterations in

our algorithm,then we stop the iteration. We used the generalized cross validation (GCV) to determine to stop the iteration. The GCV is a very effective procedure in variable selection (Cho *et al.*, 2010; Hwang, 2010; Cho, 2011; Shim, 2011). In this paper, the GCV for chosen β^* at each trial was evaluated by applying the following procedure.

$$GCV = \frac{\sum_{i=1}^{N}\left(y_i - \sum_{l=1}^{J} a_l f_l(x_i)\right)^2}{(N-a)^2},$$

where $a = trace\left\{\sum_{l=1}^{J} \widehat{a}_l K_l (K_l^T K + \lambda_l I) K_l^T\right\}$.

3.2. Random kernel ridge regression

Random kernel ridge regression (RKRR) uses KRR instead of the decision tree. Owing to the characteristics of KRR, we need to construct initial values of kernel parameter σ and penalty parameter λ. We used the GCV in KRR to determine the initial value of (σ, λ) and pick a few pairs that are around the chosen one.

The algorithm of RKRR. is as follows:

1) Determine σ and λ
2) For $b = 1$ to B :
   a) Draw a bootstrap sample Z^* of size N from the training data X.
   b) Select m variables at random from the p variables and calculate K_b using Z^* with m selected variables, where
   $K_b = (e^{-\sigma_b ||x_i - x_j||^2})_{N\times N}, i, j = 1, \cdots, N.$
   c) Compute the following (Y^* is the response variable of Z^*).
   $\widehat{f}_b(Z^*) = K_b \widehat{a}_b$, where $\widehat{a}_b = (K_b^T K_b + \lambda I)^{-1} K_b^T Y^*$.
3) Output the ensemble of $\left\{\widehat{f}_b\right\}_1^B$.

To make a prediction at a new point x :

$$\widehat{f}_{rf}^B(x) = \frac{1}{B}\sum_{b=1}^{B} \widehat{f}_b(x),$$

where $\widehat{f}_b(x) = K_b(x)\widehat{a}_b$ and $K_b(x) = (e^{\sigma_b ||x - x_i||^2})_{1\times N}, i = 1, 2, \cdots, N.$

First, determine the value of (σ, λ), then sample N number of observation by using bootstrap method. Next, select randomly m number of variables then calculate kernel by using the bootstrap sample with m chosen variables. In 2)-c), $\widehat{f}_b(Z^*)$ is calculated by using m variables and bootstrap samples in each iteration. It should be noted that $\widehat{a}_b$ must use bootstrap samples just as in 2)-c), $\widehat{f}_b(Z^*)$ will become $K_b\widehat{a}_b$.

4. Comparison

To compare the proposed methods, we use mean squared error (MSE) and mean squared prediction error (MSPE). We divided data into two parts, of which 70% is used for training and 30% is for testing. MSE and MSPE have a same formula. However, MSE is obtained by using training data set and MSPE is obtained byusing test data set. The formula of MSE is as follows.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}\left(\widehat{f}(x_i) - f(x_i)\right)^2,$$

where N is the number of observations.

The data of prostate cancer and the Boston housing data are used. Prostate cancer data comes from a study that examined the correlation between the level of prostate specific antigen and a number of clinical measures in men who were about to receive a radical prostatectomy. It has 97 observations with 9 variables. The Boston housing data for 506 census tracts of Boston housing from the 1970 census. It has 506 observations with 14 variables.

4.1. Comparison of FSR and FSKRR

Figure 4.1 shows the result of FSKRR and FSR for regression problem. The horizontal dotted line represents MSPE for the result of FSR. The perpendicular dotted line indicates the stop point where the GCV was minimum. Since two errors do not change after the stop points, we can say that the stop points were selectedwell.

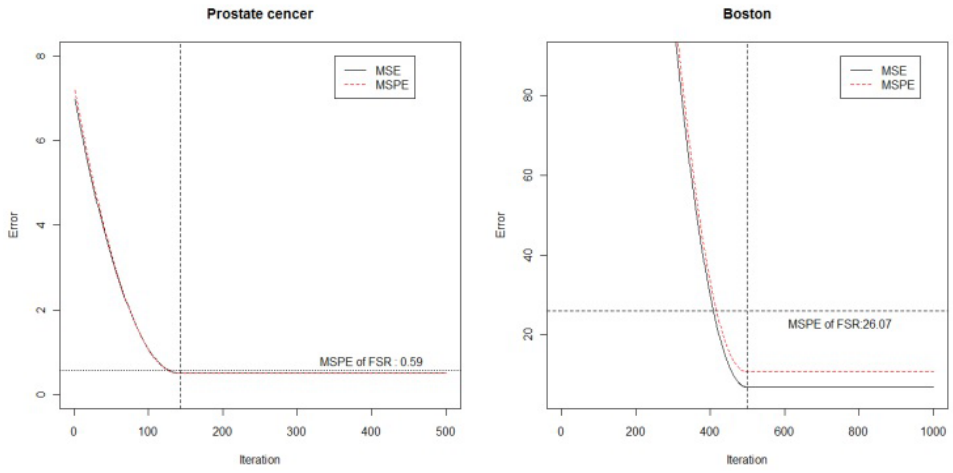

Figure 4.1 The results of FSR and FSKRR

Since two errors decreased rapidly, we can see that the predictability increased significantly. Table 4.1 shows that MSPEs obtained from our proposed method are smaller than those from FSR in both of two data sets.

Table 4.1 MSE and MSPE of FSR and FSKRR

	FSR		stop point	FSKRR	
	MSE	MSPE		MSE	MSPE
Prostate	0.386	0.658	143	0.458	0.607
Boston housing	17.964	25.781	500	5.254	14.893

Table 4.2 The result of t-test of MSPE

		Mean of MSPE	Diff(1-2) Mean	Diff(1-2) Std Dev	t-value	p-value
Prostate	FSR(1)	0.656	0.099	0.126	7.857	<.0001
	FSKRR(2)	0.557				
Boston housing	FSR(1)	24.024	7.987	4.190	19.062	<.0001
	FSKRR(2)	16.037				

We repeated the two processes of FSR and FSKRR to obtain MSPEs with different data sets 100 times. Table 4.2 shows the results of our simulations.It shows that the MSPE from FSKRR is significantly smaller than that from FSR in each case of the prostate data and the Boston housing data.

4.2. Comparison of RF and RKRR

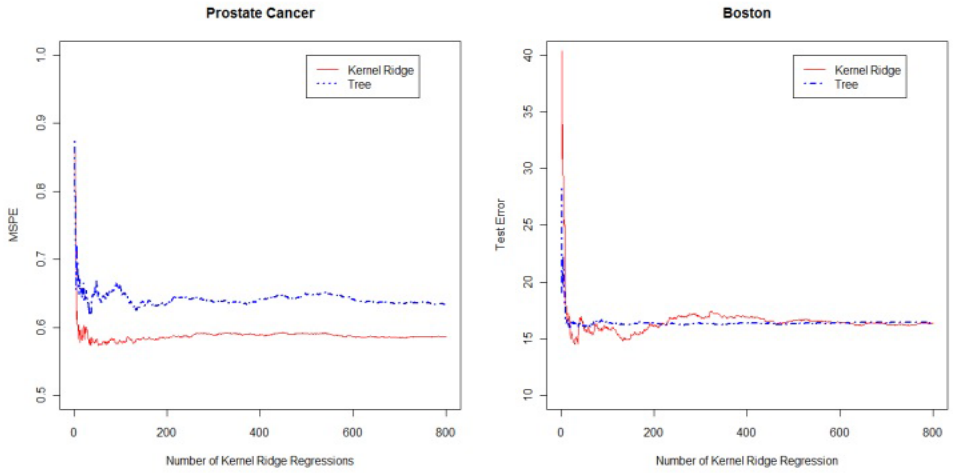

Figure 4.2 The results of RF and RKRR

Table 4.3 MSE and MSPE of RF and RKRR

	RF		RKRR	
	MSE	MSPE	MSE	MSPE
Prostate	0.348	0.634	0.507	0.587
Boston housing	9.446	16.400	11.431	16.390

Figure 4.2 shows the results of RF and RKRR. From Figure 4.2 we can see that two MSPEs decreased rapidly in a small number of iteration and MSPEs obtained from RKRR is smaller than those from RF. Table 4.3 is the results of the last two models from RF and RKRR. It shows that MSPEs obtained from our proposed method RKRR are smaller than those from RF in both of two data sets.

Table 4.4 The result of t-test of MSPE

		Mean of MSPE	Diff(1-2) Mean	Diff(1-2) Std Dev	t-value	p-value
Prostate	RF(1)	0.641	0.008	0.197	0.406	0.685
	RKRR(2)	0.632				
Boston housing	RF(1)	17.243	0.207	4.372	0.473	0.636
	RKRR(2)	17.036				

We repeated the two processes of RF and RKRR to obtain MSPEs with different training and test sets 100 times. Table 4.4 shows the results of our simulations. It shows that the MSPE obtained from RKRR is smaller than that from RF in each case of prostate and Boston housing data but is not statistically significant.

5. Conclusion

In this study, KRR was used instead of the decision tree in RF and FSR to investigate whether KRR increases the accuracy of prediction in regression problems. From the simulation results with the data of prostate cancer and Boston housing data, we see that the MSPEs obtained from FSKRR are significantly smaller than those from FSR in regression problems. And we see that the MSPEs obtained from RKRR are relatively smaller than those from RF in regression problems. In FSKRR all the variables were used at each step so that we could determine the kernel parameter σ and penalty parameter λ easier than in RKRR. The error curves in FSKRR were simpler than those in RKRR, so we could find the final model after the smaller number of iterations in FSKRR than those in RKRR.

Considering all these facts, FSKRR is found to be more effective than RKRR in regression problems.

In the near future we want to investigate these two methods of FSKRR and RKRR with more various examples for classification as well as regression.

References

Breiman, L. (1996). Bagging predictors. *Machine Learning Journal*, **26**, 123-140.

Breiman, L. (2001). Random forests. *Machine Learning Journal*, **45**, 5-32.

Cho, D. (2010). Mixed-effects LS-SVR for longitudinal data. *Journal of the Korean Data & Information Science Society*, **21**, 363-369.

Cho, D., Shim, J. and Seok, K. H. (2010). Doubly penalized kernel method for heteroscedastic autoregressive data. *Journal of the Korean Data & Information Science Society*, **21**, 155-162.

Efron, B., Hastie, T., Johnstone, I. and Tibshirani, R. (2004). Least angle regression. *Annals of Statistics*, **32**, 407-451.

Freund, Y. and Schapire, R. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, **55**,119-139.

Hastie, T., Taylor, J., Tibshirani, R. and Walther. G. (2007). Forward stagewise regression and the monotone lasso. *Electronic Journal of Statistics*, **1**, 1-29.

Hwang, H. (2010). Variable selection for multiclassification by LS-SVM. *Journal of the Korean Data & Information Science Society*, **21**, 959-965.

Shim, J. (2011). Variable selection in the kernel Cox regression. *Journal of the Korean Data & Information Science Society*, **22**, 795-801.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society B*, **58**, 267-288.