BayesRandomForest: An R implementation of Bayesian Random Forest for Regression Analysis of High-dimensional Data

Oyebayo Ridwan Olaniran (rid4stat@yahoo.com)
Universiti Tun Hussein Onn Malaysia

Mohd Asrul Affendi Bin Abdullah (afendi@uthm.edu.my)
Universiti Tun Hussein Onn Malaysia

ABSTRACT

Random Forest (RF) is a popular method for regression analysis of low or high-dimensional data. RF is often used with the later because it relaxes dimensionality assumption. RF major weakness lies in the fact that it is not governed by a statistical model, hence probabilistic interpretation of its prediction is not possible. RF major strengths are distribution free property and wide applicability to most real life problems. Bayesian Additive Regression Trees (BART) implemented in R via package BayesTree or bartMachine offers a bayesian interpretation to random forest but it suffers from high computational time as well as low efficiency when compared to RF in some specific situation. In this paper, we propose a new probabilistic interpretation to random forest called Bayesian Random Forest (BRF) for regression analysis of high-dimensional data. In addition, we present BRF implementation in R called BayesRandomForest. We also demonstrate the applicability of BRF using simulated dataset of varying dimensions. Results from the simulation experiment shows that BRF has improved efficiency over its competitors.

Keywords: *Random Forest, Bayesian Additive Regression Trees, High-dimensional, R*

JEL Classification*: C11, C39*

1 INTRODUCTION

Ensemble of trees based methods have become popular choices for predicting qualitative response (classification) and quantitative response (regression) [1]. Ensemble methods train multiple learners to construct one learner from training data. Random Forest (RF, [2]) and Gradient Boosting Machine (GBM, [3]) are the two well established ensemble based method. The two methods focused on improving the unstable prediction problem in Classification and Regression Trees (CART, [2]). Apart from RF and GBM,

other recently developed ensemble based tree methods are Bayesian Additive Regression Trees (BART, [4]), dynamic trees [5], Bayesian Forest and Empirical Bayesian Forest (BF; EBF, [6]) and Bayesian Additive Regression Trees using Bayesian Model Averaging (BART-BMA, [7]). [7] conducted a simulation study using [8] data originally motivated by Multivariate Adaptive Regression Splines (MARS) to compare the predictive performance of RF, BART and BART-BMA. The simulation results revealed that BART is only better than RF in terms of computation time.

RF procedure requires selection of bootstrapped sample of training data $(n \times p)$ and subsample of feature set p to create splits used in the split selection stage. The number of subsampled feature is often held fixed as $\approx \sqrt{p}$ for classification or $\approx p/3$ for regression. This subsample size does not take into account the number of relevant features in the entire feature set, thus the chance of selecting relevant features increases with increased p. Therefore, using same data configuration, the predictive performance of RF reduces with increasing p. However, if the feature space is well populated with relevant features, RF predictive performance will spontaneously increase due to increase in the hypergeometric probability.

In this paper, we extend RF by introducing Bayesian weighted sampling and splitting and called it Bayesian Random Forest (BRF). The weighted splitting is achieved via Bayesian inference of the two sampling procedures involved in RF. We developed a fully Bayesian ensemble of tree procedure that is similar in spirit to RF. We also implemented the procedure in an ongoing R package "BayesRandomForest".

2 OVERVIEW OF BAYESIAN RANDOM FOREST

Bayesian Random Forest (BRF) is a Bayesian implementation of the nonparametric function estimates obtainable from regression trees. Regression trees rely on recursive binary partitioning of predictor space into a set of hyper rectangles in order to approximate some unknown function *f* [1]. The main advantages of Tree-based regression models are attributed to their success in modelling of linear, non-linear and interaction effects. The main weakness of single tree method is the instability in prediction due to the top down approach involved in recursive partitioning. This suggests the need to average many trees to reduce variance. BRF is an adaptive version of RF but guided with Bayesian reasoning. That's every stage of estimation in RF is mimicked but estimation fully relies on Bayesian paradigm.

Formally, given training dataset $[Y_i, x_{i1}, x_{i2}, \ldots, x_{ip}, i = 1, 2, \ldots, n]$, where y_i assumes continuous values and x_i is the vector of features, BRF can be describe as;

$$Y = \sum_{j=1}^{J} \mathfrak{I}_j(\beta_m : x \in R_m) + \varepsilon \tag{1}$$

where β_m is an estimate of Y in region m [R_m], J is the number of trees in the forest, $\mathfrak{I}_j(\beta_m : x \in R_{...})$ is a single regression tree, ε is the random noise that occurs in estimating β_m and its assumed to be independent and identically Gaussian distributed with mean zero and variance σ^2 over all trees. BRF model is very much similar to BART [4] but differs in terms of prior specification and posterior estimation approach.

2.1 Priors and Posterior Specification for Bayesian Random Forest

BRF has three major parameters which can be attributed to its ensemble nature. The first parameter is attributed to model uncertainty, which in this case is tree $\mathfrak{I}$ uncertainty. Here we propose a uniform prior $\mathfrak{I}_0 \sim U(0,1)$ such that $P(\mathfrak{I}_0) = 1$ for any candidate tree. We used this prior specification to retain the average weighing procedure of RF so that each tree $\mathfrak{I}$ has equal right. The advantage of this prior is to retain the strength of RF in terms of correcting overfitting problem. The second form of prior is prior on terminal node parameter β_m, here we propose Gaussian prior $N(\mu, \sigma_\mu^2)$. We adapted the bootstrap prior technique [9 - 11] to obtain the prior hyperparameters μ and σ_μ^2 for each tree. The major advantage of bootstrap prior is that it guarantees an unbiased estimate of β_m for each tree. For the parameter σ^2, we propose the standard gamma default prior $G(\alpha, \theta)$ [12] with $\alpha = \theta$ such that $E[\sigma^2 | G(\alpha, \theta)] = 1$. The complete prior specification for BRF is thus;

$$P(\mathfrak{I}_1, \mathfrak{I}_2, \dots, \mathfrak{I}_m) = \prod_{j=1}^{J} P(\mathfrak{I}_j, \beta_{jm}) P(\sigma^2) \tag{2}$$

$$P(\mathfrak{I}_1, \mathfrak{I}_2, \dots, \mathfrak{I}_m) = \prod_{j=1}^{J} P(\beta_{jm}) P(\sigma^2) \tag{3}$$

(3) follows from (2) since $P(\mathfrak{I}_0) = 1$. The posterior distribution using $L(\mathfrak{I}_1, \mathfrak{I}_2, \dots, \mathfrak{I}_m | y, x)$ and (3) is then obtain via Metropolis Hasting (MCMC) algorithm [13].

Now to mimic RF completely, we also specified some procedural priors similar in spirit to bootstrapping and features subsampling in RF. For the two procedures, we proposed Bayesian simple random sampling with replacement and Bayesian simple random sampling without replacement with posterior densities given in (4) and (5):

$$P(\pi|\, a, b) = \frac{\Gamma(n+a+b)}{\Gamma(a+1)\Gamma(b+n-1)}\pi^{a}(1-\pi)^{b+n}, 0 \leq \pi \leq 1 \quad (4)$$

$$P(V|h, l, p, S, T) = \frac{\binom{S+V}{S+1}\binom{T+p-V}{T+l-h}}{\binom{S+T+p+1}{S+T+l+1}}, h \leq v \leq p-l+h \quad (5)$$

where π is the probability of selecting any $i \in n$ in each j step, $\Gamma(\mathrm{d})$ is the gamma function evaluated at d, a is the prior expected number of times any $i \in n$ could be selected, b is its complement, V is the number of relevant features whose posterior is sought, h is the sample realization of relevant features, p is the total number of features, l is the number of subsampled features as in RF, S is the prior number of relevant features and T is the prior number of irrelevant features.

If we denote the posterior density in (4) and (5) as ω and δ, we then obtain a weighted Bayesian regression tree at each j step by weighing the data by ω and then weighing the impurity at each split by δ. For a Sum Squares Error (SSE) impurity, we propose a weighted impurity using;

$$SSE(\delta) = (1-\delta)\left[\sum_{i=1}^{n_m}\left(y_i - \hat{\beta}_m\right)^2\right] \quad (6)$$

where $\hat{\beta}_m$ is the posterior mean of β_m at each node m. The variable with weight $\delta \to 1$, will correspond to variable with minimal unweighted $SSE(\delta)$ and therefore useful for further splitting step. If on the other hand $\delta \to 0$, implies the variable is not useful and therefore expected to yield a maximal unweighted $SSE(\delta)$. In this case, the proposed weighted $SSE(\delta)$ returns the unweighted $SSE(\delta)$ so that the variable is dropped at the splitting stage. The idea behind this is to control the mixture behavior of hypergeometric distribution. The dominant category determines the estimates of categories probability. RF fails to balance this gap by specifying $l = \sqrt{p}$, for example if $p = 10000$; $l = 100$, which implies taking a random sample of 100 features to be used in each split. Suppose there are 5 relevant features as in Friedman (1991), the hypergeometric probability of selecting any relevant features is approximately 0.049. This implies that at each splitting step, there is about 95% chance of selecting irrelevant feature. This high probability can be attributed to fewer number of relevant features in relation to large number of features p. Thus RF assumes that the entire feature space p is reasonably populated with relevant features. The dilemma with RF is to think of increasing l, yes this will increase the hypergeometric probability but at the expense of increasing the correlation between trees. This is indeed the situation where RF irretrievably breaks down [14].

3 FRIEDMAN FIVE DIMENSIONAL DATA

Following (4, 6], [8] simulated data was used to compare the results of BRF, BF, RF, BART and GBM. The simulated datasets where $x_1,\ldots, x_p$ are $iid \sim U(0,1)$ random variables and $\varepsilon \sim N(0,1)$ were formulated as;

$$y = 10sin(x_1x_2) + 20(x_3 - 0.5)^2 + 10x_4 + 5x_5 + \varepsilon \quad (7)$$

The five methods were compared over various dataset sizes with five relevant features $[x_1,\ldots,x_5]$ and complement of $p = [100, 500, 1000, 5000, 10000]$ as the irrelevant features. The associated hypergeometric probabilities with p are $P(V|l = \sqrt{p}) = [0.416, 0.202, 0.150, 0.06, 0.05]$. The predictive performance of the methods was assessed with 10 folds cross validation of Root Mean Squared Error (RMSE) at sample size $n = 50, 100$. All analyses were carried out using R with newly developed function "BayesRandomForest" accessible in [15] for BRF, "bartMachine" [1] for BART, "gbm" [16] for GBM, "randomForest" [17] for RF and "ranger" [18] for [BF].

Table1 shows the 95% credible interval for Bayesian based methods (BRF, BF, BART) and confidence interval for frequentist based methods (GBM and RF) of RMSE. The intervals are computed using the bake-off 10 folds cross-validation of $10 \times n$. The results show that 95% width for BRF is the lowest when compared to other methods. The least performing method is BART with the maximum width. The next in category in terms of stability of RMSE is RF. The poor performance of BART with increasing p in Fig1. and Fig2. is attributed to the nature of tree growth which involves using all features for each tree.

95% Credible and confidence interval of RMSE at sample size $n = 50,100$.

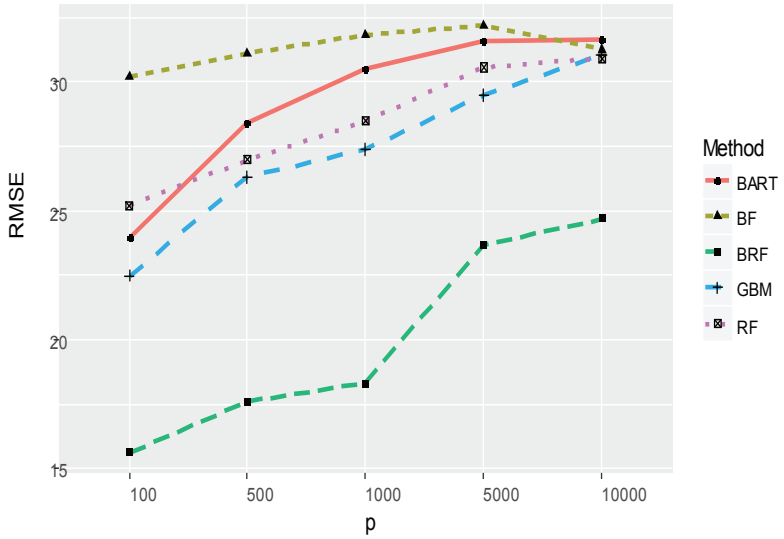
Table 1

Method	$n = 50$			$n = 100$		
	2.5%	97.5%	width	2.5%	97.5%	width
BRF	15.32	25.26	9.94	9.82	12.17	2.36
BF	28.50	33.03	4.53	24.24	32.41	8.18
RF	24.77	31.44	6.67	14.46	21.63	7.17
BART	23.18	32.53	9.36	8.41	34.01	25.61
GBM	19.92	36.70	16.77	6.71	16.88	10.17

This lead to overfitting and eventual poor performance in out of sample validation. Furthermore, the existing robust method to large p with low relevant feature is GBM because of its internally embedded feature

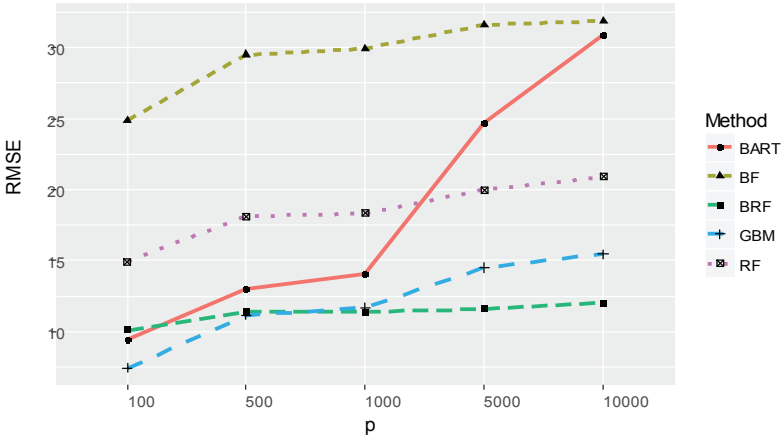
selection procedure [15]. BRF results challenged this claim with stable and better results at sample size $n = 50, 100$.

Figure 1



RMSE with increasing p at sample size $n = 50$, as expected increasing p reduces the hypergeometric probability which increases RMSE of RF (*purple dotted lines*). Similarly, GBM, BRF and BART are affected with the increase but the effect is minimal on BRF. BF tends to be stable over p but the RMSE is on the high side.

Figure 2



RMSE with increasing p at sample size $n = 100$, as expected increasing p reduces the hypergeometric probability which increases RMSE of RF (*purple dotted lines*). The increase in sample size tends to balance the methods RMSE but BART, GBM, BF and RF are still affected with the increase in p.

4 CONCLUSION

In this paper, we presented the theoretical framework for Bayesian Random Forest (BRF) for regression analysis of high-dimensional data. By way of example, We consider its application on simulated Friedman data set with large p and fewer number of relevant features. We also compared the predictive performance of the method with some existing methods using RMSE via 10 folds cross-validation. The results observed from the simulation study shows that BRF is highly robust to large p small relevant feature issue at a reasonable sample size n when compared with its competitors.

Funding

This work was supported by Universiti Tun Hussein Onn, Malaysia [grant numbers Vot, U607].

References

1. Kapelner, A. & Bleich, J. (2014a). bartmachine: Machine learning with bayesian additive regression trees. ArXiv e-prints.
2. Breiman, L. (2001). Random forests. *Machine Learning*, 45, pp. 5–32.
3. Friedman, J. H. (2001b). Greedy function approximation: A gradient boosting machine. *Ann. Statist.,* 29 (5), pp. 1189–1232.
4. Chipman, H.A. George, E. I. and McCulloch, R. E. (2010). BART: Bayesian Additive Regression Trees. *Annals Applied Statatistics*, 4, pp. 266–298.
5. Taddy MA, Gramacy RB, Polson NG (2011). "Dynamic Trees for Learning and Design." *Journal of the American Statistical Association*, **106**(493), 109–123. doi:10.1198/jasa.2011.ap09769.
6. Taddy, M., Chen, C. S., Yu, J., & Wyle, M. (2015). Bayesian and empirical Bayesian forests. *arXiv preprint arXiv:1502.02312*.
7. Hernández, B., Raftery, A. E., Pennington, S. R., & Parnell, A. C. (2015). Bayesian Additive Regression Trees using Bayesian Model Averaging. *arXiv preprint arXiv:1507.00181*.
8. Friedman, J. H. (1991). Multivariate adaptive regression splines (with discussion and a rejoinder by the author). Annals of Statistics, 19, 1{67.
9. Olaniran, O. R., & Yahya, W. B. (2017). Bayesian Hypothesis Testing of Two Normal Samples using Bootstrap Prior Technique. *Journal of Modern Applied Statistical Methods*, *In press*.
10. Olaniran, O. R., Olaniran, S. F., Yahya, W. B., Banjoko, A. W., Garba, M. K., Amusa, L. B. and Gatta, N. F. (2016): Improved Bayesian Feature Selection and Classification Methods Using Bootstrap Prior Techniques; Anale. Seria Informatică. 14(2), 46-52.

11. Olaniran, O. R., & Affendi, M. A. (2017). Bayesian Analysis of Extended Cox Model with Time-varying Covariates using Bootstrap prior. *Journal of Modern Applied Statistical Methods*, *In press*.
12. Yahya W. B., Olaniran O. R. and Ige, S. O. (2014): On Bayesian Conjugate Normal Linear Regression and Ordinary Least Square Regression Methods: A monte Carlo Study. Ilorin Journal of Science, 1(1): 216-227.
13. Gelman, A, Carlin, JB, Stern, HS, Dunson, DB, Vehtari, A, and Rubin, DB (2013). *Bayesian Data Analysis*. Boca Raton, FL: CRC Press.
14. Hastie T, Tibshirani R, Friedman J (2011). *The Elements of Statistical Learning: Prediction, Inference and Data Mining*. 2nd edition. Springer-Verlag, New York.
15. Olaniran O. R. (2017): BayesRandomForest: Bayesian Random Forest for Regression Trees: https://github.com/rid4stat/BayesRandomForest.
16. Greg, R with contributions from others (2017). gbm: Generalized Boosted Regression Models. R package version 2.1.3. https://CRAN.R-project.org/package=gbm
17. Liaw, A. Wiener, M. (2002). Classification and Regression by randomForest. R News 2(3), 18--22.
18. Marvin N. W., Andreas Z. (2017). ranger: A Fast Implementation of Random Forests for High Dimensional Data in C++ and R. Journal of Statistical Software, 77(1), 1-17. doi:10.18637/jss.v077.i01