applied sciences

MDPI

Article

Optimal Feature Set Size in Random Forest Regression

Sunwoo Han [1] **and Hyunjoong Kim** [2,*]

[1] Fred Hutchinson Cancer Research Center, Vaccine and Infectious Disease Division, Seattle, WA 98006, USA; shan@fredhutch.org
[2] Department of Applied Statistics, Yonsei University, Seoul 03722, Korea
* Correspondence: hkim@yonsei.ac.kr; Tel.: +82-2-2123-2545

Abstract: One of the most important hyper-parameters in the Random Forest (RF) algorithm is the feature set size used to search for the best partitioning rule at each node of trees. Most existing research on feature set size has been done primarily with a focus on classification problems. We studied the effect of feature set size in the context of regression. Through experimental studies using many datasets, we first investigated whether the RF regression predictions are affected by the feature set size. Then, we found a rule associated with the optimal size based on the characteristics of each data. Lastly, we developed a search algorithm for estimating the best feature set size in RF regression. We showed that the proposed search algorithm can provide improvements over other choices, such as using the default size specified in the *randomForest* R package and using the common grid search method.

Keywords: random forest; feature set size; grid search; regression

Citation: Han, S.; Kim, H. Optimal Feature Set Size in Random Forest Regression. *Appl. Sci.* **2021**, *11*, 3428. https://doi.org/10.3390/app11083428

Academic Editor: Federico Divina

Received: 26 March 2021
Accepted: 9 April 2021
Published: 12 April 2021

1. Introduction

Random forest [1] is one of the most successful ensemble machine learning methods because it has many advantages. It is suitable for large data with fast model fitting and evaluating, is applicable to both regression and classification, is robust to outliers, can deal with simple linear and complicated nonlinear associations, and produces competitive prediction accuracy for high-dimensional data [2,3]. These unique advantages lead RF to gain increasing interests in a variety of research fields. In particular, it has been actively used and shown great accomplishment in many biomedical applications. For example, RF has been successfully utilized in cancer prognosis and diagnosis [4,5], to predict infectious diseases with high accuracy [6,7], and to recognize disease associated genes in microarray data [8,9].

The RF prediction performance can be further improved by tuning its hyper-parameters, a factor to determine the learning process, which are usually set by users. In the RF algorithm, there are three main hyper-parameters that highly affects the performance [10]: (1) *ntrees*, which controls how many trees are constructed in the ensemble; (2) *nodesize*, which decides the size of each tree by controlling minimum number of observations in terminal nodes; and (3) *mtry*, which controls the feature set size to search for the best split rules at each node of trees. Many studies have been conducted to explore the influence of the three hyper-parameters on the RF prediction performance. Oshiro et al. [11] and Probst and Boulesteix [12] investigated how to obtain the optimal number of trees. Han and Kim [13] examined the effect of tree size and developed a new ensemble method for improving the RF model by growing deeper trees. Bernard et al. [14], Goldstein et al. [15], and Han and Kim [16] studied the impact of the feature set size on the prediction performance in the context of classification.

Feature set size, i.e., *mtry*, can be considered as the salient hyper-parameter in the sense that it controls the accuracy of individual trees and the diversity between pairs of trees in the ensemble [1]. Setting a large value makes accurate individual trees, but the trees

would be similar to each other; setting a small value enhances the diversity between trees, but each tree would have poor prediction accuracy. The two indicators have a trade-off relationship, which must be balanced to achieve the best RF prediction accuracy. Hence, using the appropriate feature set size is the most important task in the RF model fitting to obtain good prediction accuracy. Unfortunately, no theoretical proof has been developed to select the optimal feature set size that provides the highest accuracy, and, in most cases, the default value specified in a software package is used. For example, the default *mtry* value is $p/3$ for the *randomForest* R package and p for the *RandomForestRegressor* in Python's *sklearn.ensemble* package, where p denotes the number of features.

Various hyper-parameter optimization algorithms have been applied to obtain the appropriate feature set size. (1) Manual tuning is a traditional optimization method which is performed by users manually. It depends on the trial and error process, so it would be effective only for experienced users. (2) Random search is a method of randomly selecting feature set sizes, evaluating the RF model for each size, and then estimating the optimal size based on RF predictions. It is very fast since only a small number of candidates is considered, but the true optimal size can be missed. (3) Grid search, which has a similar mechanism with the random search, except that every feature set size is explored. It can estimate the true optimal size with high accuracy, but it is computationally intensive. (4) Recently, a Bayesian model-based hyper-parameter optimization algorithm called BOA [17] has been increasingly implemented in machine learning and deep learning communities [18,19]. This method selects the hyper-parameters of the next step using the probability model constructed by the previous step. We propose a novel algorithm that can more efficiently replace the grid search method without using a probability model.

We investigated the effect of feature set size on the RF prediction performance in the context of regression because most existing studies have been developed focusing on classification problems. There are not enough references in regression problems, even though classification and regression are quite different analysis task. Using many datasets, a total of 56 real and artificial datasets, we first investigated whether the RF prediction performance is affected by the feature set size, and then we found a rule associated with the optimal size for each dataset. Finally, we developed a search algorithm that combines a typical grid search and two unique concepts for estimating the best size in RF regression. We compared the proposed method with the typical grid search algorithm. In addition, we compared the size estimated by the proposed method with the optimal size and the default size specified in the *randomForest* R package.

The rest of the paper is organized as follows. Section 2 begins with a brief introduction to the RF algorithm in regression. Through experimental studies, we first explored the influence of feature set size on the RF prediction performance, and then we examined relationships between an optimal feature set size and characteristics of given datasets. In Section 3, we propose a search algorithm for estimating the best size in regression. In Section 4, we study the impact of our proposed search algorithm on the RF prediction by comparing it with a default size and the optimal size. In addition, we compare the proposed algorithm with a typical grid search method. The paper ends with conclusion in Section 5.

2. Feature Set Size (*mtry*)

2.1. Random Forest Algorithm in Regression

RF, a popular ensemble machine learning method, is a modified version of bagging [20] since they share conceptually same algorithms to each other, but the main difference is that RF further reduces the variance of a prediction model by combining more de-correlated trees than bagging. The RF algorithm starts with generating bootstrapped samples from a training dataset. Multiple regression trees are fitted on the bootstrapped samples. When the trees are constructed, RF uses a random feature subset instead of all features to find the best split rule at each split of trees. The results of each tree are combined to produce a final prediction. The detailed RF algorithm in regression is described in Algorithm 1.

Algorithm 1 The RF algorithm in regression.

Training Phase :
Given :
- D : training set with n observations, p features, and the response variable.
- B : number of regressors in the ensemble.

Procedure :
For $b = 1$ to B
1. Generate bootstrapped sample D_b^* from training set D.
2. Grow a regression tree using the bootstrapped sample D_b^*.
   For a given node t,
   (i) Randomly sample *mtry* features from the full features.
   (ii) Find the best split rule using the random feature subset.
   (iii) Split the node t into two child nodes using the best split rule.
   Repeat (i)–(iii) until stopping rules are met.
3. Obtain a trained regressor R_b.

Test Phase :
For a test instance x, the prediction estimated by the B regressors is given as :

$$R(x) = \frac{1}{B}\sum_{b=1}^{B} R_b(x)$$

2.2. Influence on Prediction Performance

In the literature, three *mtry* values, 1, $log_2(p) + 1$, and $p/3$, have been primarily used in the RF model fitting. The first two were introduced by Breiman [1], and the last was recommended for regression problems by Hastie et al. [2] and Liaw and Wiener [21]. We consider $p/3$ as the default *mtry* value throughout this paper.

We conducted experimental studies to investigate the influence of the feature set size on the RF prediction accuracy in regression. The experiments were based on 56 real or artificial datasets that were used in other studies or came from the UCI data repository [22]. Tables 1 and 2 contain information about the datasets.

Table 1. The description of 56 datasets for regression problems.

Name	Observation	Feature	Source
aba	4177	8	UCI (Abalone)
air	1503	5	UCI (Airfoil Self-Noise)
ais	202	12	[23]
alc	2462	18	[24]
ame	3044	21	[25]
app	19,735	25	UCI (Appliances energy prediction)
att	838	9	[26]
aut	398	7	UCI (Auto MPG)
beh	135	17	UCI (Behavior of the urban traffic in Sao Paulo)
blo	208	134	R Library *caret* (Blood-Brain Barrier)
bos	506	13	R Library *mlbench (Boston Housing data)*
bre	198	32	UCI (Breast Cancer Wisconsin)
bsd	731	11	UCI (Bike Sharing with count daily)
bsh	17,379	12	UCI (Bike Sharing with count hourly)
bud	1729	10	[27]
can	3775	9	[28]
car	804	17	R Library *caret* (cars)
cdo	378	9	[29]
com	1994	100	UCI (Communities and Crime)
con	1030	8	UCI (Concrete Compressive Strength)

Table 1. *Cont.*

Name	Observation	Feature	Source
cou	3114	13	[30]
cox	462	255	R Library *caret* (Cox-2 Activity)
cps	534	10	[31]
dee	696	13	[31]
dia	366	15	[30]
eec	1296	8	UCI (Energy efficiency with Cooling)
eeh	1296	8	UCI (Energy efficiency with Heating)
ele	10,000	12	UCI (Electrical Grid Stability Simulated)
fac	500	18	UCI (Facebook metrics)
fam	1318	22	[32]
fat	252	17	[33]
fir	517	8	UCI (Forest Fires)
fis	6806	24	[34]
hat	100	13	[35]
hit	322	18	R Library *ISLR* (Hitters)
ins	5822	85	UCI (Insurance Company Benchmark)
ist	538	7	UCI (ISTANBUL STOCK EXCHANGE)
lab	2953	18	[36]
lah	200	16	[37]
mdp	1495	8	R Library *COUNT* (medpar)
med	4406	21	[38]
ozo	366	12	R Library *mlbench* (Ozone)
pks	5875	16	UCI (Parkins Telemonitoring with total-UPDRS)
rat	144	10	[38]
rwm	3874	14	R Library *COUNT* (rwm1984)

Table 2. The description of 56 datasets for regression problems.

Name	Observation	Feature	Source
sce	113	10	[39]
sms	141	10	[39]
sol	1066	10	UCI (Solar Flare with C-class)
spm	395	30	UCI (Student Performance with Math)
spp	649	30	UCI (Student Performance with Portuguese)
tre	100	8	[40]
tri	186	60	[41]
wag	3380	17	[42]
wqr	1599	11	UCI (Wine Quality with red)
wqw	4898	11	UCI (Wine Quality with white)
yac	308	6	UCI (Yacht Hydrodynamics)

The design of the experiments is as follows: we randomly split a dataset into 60% training set for fitting and 40% test set for evaluating. For the RF model fitting, we used the full range of values from 1 to p as the value of *mtry*. We set the number of trees as 100, which is a common choice in many RF applications [15,16,43]. For evaluation, root mean squared error (RMSE) was calculated to measure the RF prediction performance. All experiments were repeated 100 times to obtain stable results.

The results are organized with three figures. First, Figure 1 compares the RMSE depending on different *mtry* values using four representative datasets.

The large fluctuation of RMSE means that the performance of the RF is severely affected by *mtry* values. Moreover, the results clearly show that the optimal *mtry* that achieves the smallest RMSE differs from the common choices in all four datasets. Second, in Figure 2, we explore a relative distance, which is defined as (optimal *mtry*—default

mtry)/p, to measure the difference between the optimal size and the default size. There are only four datasets where the default size is matched to the optimal size, and, in most cases, the default size fails to offer the best RF prediction accuracy. In addition, there are more datasets where the optimal size is larger than the default size than there are datasets in the opposite case.

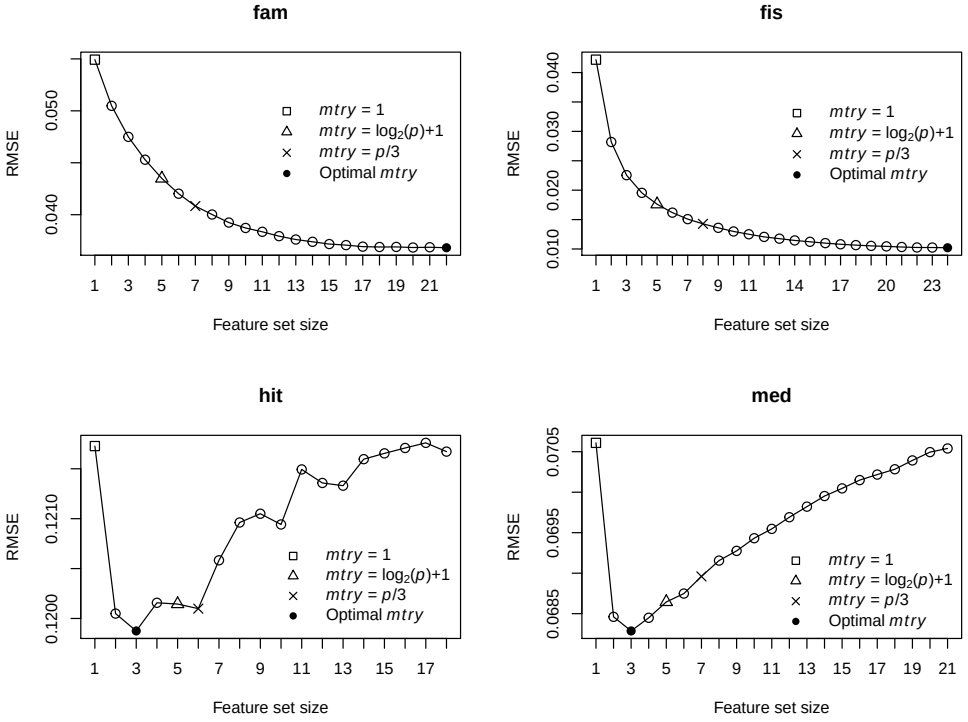

Figure 1. Comparison of RMSE depending on different feature set sizes (*mtry*) on four representative datasets (fam, fis, hit, and med). Each point is the averaged RMSE over 100 replicates.

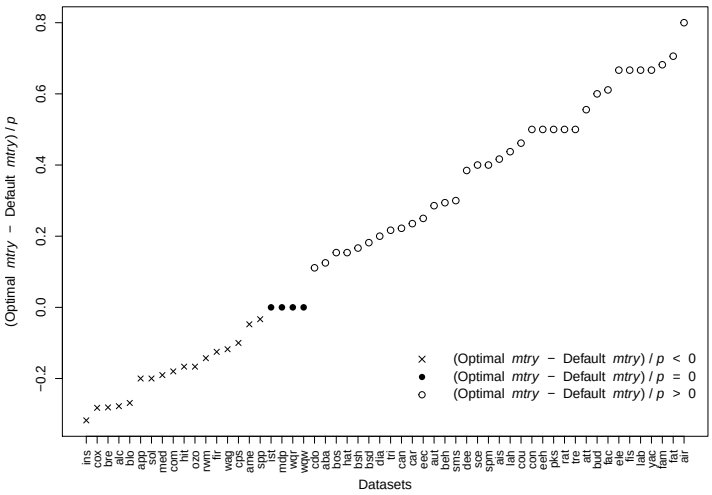

Figure 2. Comparison between the optimal *mtry* and the default *mtry* ($p/3$). The x-axis and y-axis indicate 56 datasets and the relative distance, respectively.

Last, Figure 3 compares the optimal size and the default size by using a relative RMSE, defined as $log(\frac{\text{RMSE with default } mtry}{\text{RMSE with optimal } mtry})$. The relative RMSE greater than 0 means that the optimal size achieves more accurate prediction than the default size. The RF model with the optimal size is obviously more accurate than that with the default size because almost all boxes are located above 0. The performance difference is huge when the optimal size is larger than the default size, but the difference is quite small when the optimal size is

smaller than the default size. In summary, through the experimental results, we observed that: (1) the *mtry* value has an important role in the RF prediction performance; (2) the default *mtry* value cannot guarantee the best RF performance in regression problems; (3) the optimal *mtry* value differed dataset to dataset; and (4) there is no clear pattern for the optimal size, thus it is difficult to guess the best *mtry* value in advance.

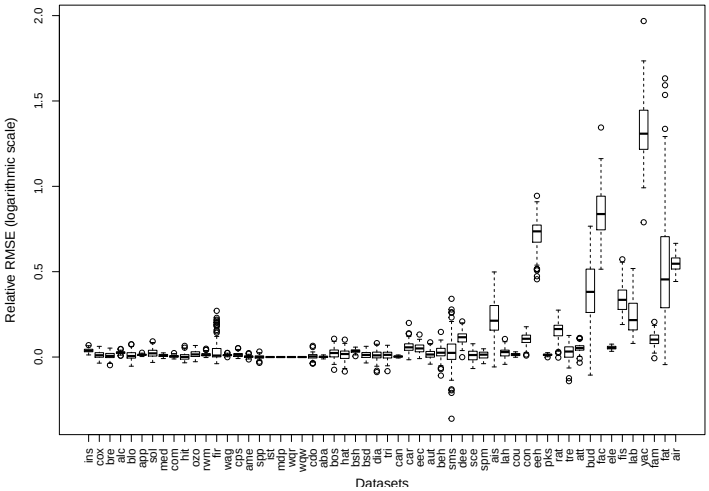

Figure 3. Comparison of the RF prediction accuracies using by the optimal *mtry* and the default *mtry*. The x-axis indicates 56 datasets with the same order as in Figure 2. The y-axis indicates the relative RMSE. Each box-plot is based on 100 relative RMSEs.

2.3. Rule for Optimal Feature Set Size

In this section, we further try to find a rule associated with the optimal *mtry* using characteristics of datasets. In the context of classification, Bernard et al. [14] and Goldstein et al. [15] observed that the optimal size tends to be related to the number of relevant features, where the relevance means that a feature is significantly associated with the target. If there are many relevant features in a given dataset, a smaller *mtry* value is preferred to utilize the relevant features equally for splitting nodes in the tree construction. As a result, the trees will be more diverse. In this situation, larger *mtry* value would cause similar trees in the ensemble because the most dominant feature is repeatedly selected for the split rule at each node. On the other hand, if there are only a few relevant features in a given dataset, a larger *mtry* value may be advantageous to increase the accuracy of individual trees. However, these observations are limited on classification problems and may not be valid in regression problems. Hence, we sought a relationship between the optimal *mtry* and the characteristics of given datasets, focusing on the relevance of features, in the context of regression.

How do we measure the relevance between the response variable Y and a feature X_i, where $i = 1, \cdots, p$, in regression? The typical Pearson correlation coefficient between Y and X_i can be ineffective because it deals with linear associations only. Hence, we consider a modified Pearson correlation coefficient between Y and $\hat{Y}_i$, which is a predicted response for X_i, obtained by a regression tree between Y and X_i. In detail, a decision tree between Y and X_i is fitted with pruning, and then the predicted response $\hat{Y}_i$ is obtained by applying X_i on the fitted tree. We can capture both linear and nonlinear associations by considering the decision tree [2]. We denote the modified Pearson correlation coefficients as R_i, $i = 1, \cdots, p$, and the larger R_i means stronger evidence that X_i is a relevant feature to the response Y.

To investigate whether the optimal *mtry* is related to the relevance of features, we consider a classification tree, where the target variable Y_{dt} is a binary outcome that is 1 if the optimal size is larger than the default size, and is 0 if the optimal size is smaller than or equal to the default size. We also create a variety of factors X_{dt} to be used in the decision tree: (1) *nrel*, number of relevant features; (2) *nirr*, number of irrelevant features; (3) *scor*, standard deviation of R_i; (4) *mcor*, mean of R_i; (5) *ccor*, coefficient of variation of R_i; and (6)

pn, ratio of features and observations. The definitions of Y_{dt} and X_{dt} are summarized in Table 3.

Table 3. Description of Y_{dt} and X_{dt} for classification tree analysis. The definitions of X_{dt} are as follows: (1) *nrel*, the number of relevant features ($Q_3(R)$: the third quartile of R); (2) *nirr*, the number of irrelevant features ($Q_1(R)$: the first quartile of R); (3) *scor*, the standard deviation of R; (4) *mcor*, the mean of R; (5) *ccor*, the coefficient of variation of R; and (6) *pn*, the ratio of features and observations. Note that R is the collection of R_i, $i = 1, ..., p$.

Attribute	Name	Definition
Target (Y_{dt})	Y_{dt}	$Y_{dt} = 1$ if Optimal *mtry* $>$ Default *mtry* $Y_{dt} = 0$ if Optimal *mtry* $\leq$ Default *mtry*
Factor (X_{dt})	*nrel*	$Length(R > Q_3(R))/p$
	nirr	$Length(R < Q_1(R))/p$
	scor	$Sd(R)$
	mcor	$Mean(R)$
	ccor	$Sd(R)/Mean(R)$
	pn	p/n

Figure 4 depicts the classification tree with pruning between Y_{dt} and X_{dt} based on a total of 56 datasets, where *scor*, standard deviation of R_i, is the most significant factor to discriminate the 56 datasets. The result shows that, for *scor* < 0.135, there are many datasets with the optimal size less than or equal to the default size; for *scor* ≥ 0.135, most datasets have an optimal size larger than the default size. These results can be interpreted as follows. If features in a given dataset have similar relevance to each other, the optimal size tends to be small, thus setting smaller *mtry* value can be helpful to achieve good prediction performance. Conversely, if features are widely distributed from irrelevant to relevant, the optimal size tends to be large, thus setting larger *mtry* value may be preferred.

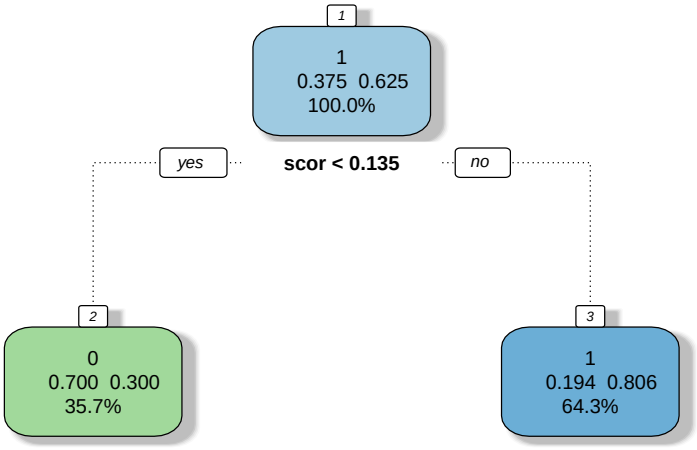

Figure 4. Classification tree with pruning between Y_{dt} and X_{dt} based on a total of 56 datasets. At each node, the first line indicates the predicted class, the second line means the proportion of classes. The last line indicates the ratio of observations belonging to the node among the total observations.

We found that the optimal feature set size may be related to the relevance of features in a given dataset, but it is not enough to select a specific *mtry* in RF model fitting. Hence, in the next section, we develop a search algorithm for estimating the best *mtry* in RF regression.

3. Search Algorithm for Optimal Feature Set Size in Random Forest Regression

We develop a search algorithm for estimating the best *mtry* in RF regression using the findings in Section 2. This algorithm is also motivated by Han and Kim [16].

Our proposed algorithm combines a typical grid search method and two unique concepts: (1) *SearchDirection*, which controls the direction of search, "forward" and "backward"; and (2) *SearchSize*, which decides how many features are searched at a time, set $ceiling(p/10)$ by default. These two concepts make our proposed search algorithm more efficient than the typical grid search method by reducing the number of searches. The proposed algorithm searches using the out-of-bag mean squared error (OOB-MSE), which is computed on the out-of-bag samples. The OOB-MSE estimate is known as a good estimate for the true MSE [1]. In the *SearchDirection* = "forward" setting, the proposed algorithm starts the searches from $mtry = floor(p/3)$ to p in increasing order. Note that, if *SearchDirection* is "backward", the searches are performed from $mtry = floor(p/3)$ to 1 in decreasing order.

We now describe our proposed search algorithm using a simple example. Suppose that *SearchDirection* is "forward" and a dataset has 15 features ($p = 15$); then, *SearchSize* is set as $ceiling(15/10) = 2$ by default. The proposed method first computes the OOB-MSE of the RF model with $mtry = (15/3) = 5$, which is denoted as $\hat{e}_5$. Then, it compares $\hat{e}_5$ with the next two OOB-MSEs, $\hat{e}_6$ and $\hat{e}_7$, which are obtained by the RF model with $mtry = 6$ and $mtry = 7$, respectively, given that *SearchSize* is 2. If $\hat{e}_5$ is smaller than the minimum of $\hat{e}_6$ and $\hat{e}_7$, the algorithm estimates the best *mtry* as 5 and terminates the search. If $\hat{e}_5$ is larger than one of $\hat{e}_6$ and $\hat{e}_7$, the minimum of $\hat{e}_6$ and $\hat{e}_7$ is continuously compared with the next two OOB-MSE, $\hat{e}_8$ and $\hat{e}_9$. The algorithm is reiterated until stopping criteria are met. The detailed search algorithm is in Algorithm 2.

Algorithm 2 A search algorithm for estimating the best *mtry* in RF regression.

Require : $SearchDirection \leftarrow$ "$forward$" or "$backward$", and $F \leftarrow floor(p/3)$
Require : $SearchSize \leftarrow ceiling(p/10)$
Require : $\hat{e}_f$: out-of-bag Mean Squared Error with $mtry = f$
Procedure :
1: **if** $SearchDirection =$ "$backward$" **then**
2: Compute $\hat{e}_F$
3: **for** x from $(F-1)$ to $(F - SearchSize)$ with decrement 1
4: **if** there is $\hat{e}_x$ already **then skip**
5: **else** compute $\hat{e}_x$
6: **end if**
7: **end for**
8: $j \leftarrow argmin_j(\hat{e}_j \mid j \in [F - SearchSize, \cdots, F-1])$
9: **if** $\hat{e}_j < \hat{e}_F$ **then** $F \leftarrow j$ and go to 3
10: **endif**
11: **if** $SearchDirection =$ "$forward$" **then**
12: Compute $\hat{e}_F$
13: **for** x from $(F+1)$ to $(F + SearchSize)$ with increment 1
14: **if** there is $\hat{e}_x$ already **then skip**
15: **else** compute $\hat{e}_x$
16: **end if**
17: **end for**
18: $j \leftarrow argmin_j(\hat{e}_j \mid j \in [F+1, \cdots, F + SearchSize])$
19: **if** $\hat{e}_j < \hat{e}_F$ **then** $F \leftarrow j$ and go to 13
20: **endif**
21: Return F

4. Experimental Study

This section studies the impact of our proposed search algorithm by comparing the RF prediction accuracy using an estimated *mtry* by our proposed algorithm, with that using the optimal *mtry*, the default *mtry*, and an estimated *mtry* by a typical grid search method. We determine the *SearchDirection* by the results of the classification tree in Figure 4. Specifically,

if a dataset falls into the node number 2, the *SeachDirection* is assigned as "backward"; if a dataset falls into the node number 3, the *SearchDirection* is assigned as "forward".

4.1. Estimated Size vs. Optimal Size

The results of comparison between the estimated *mtry* by our proposed algorithm and the optimal *mtry* are presented with two figures. First, Figure 5 compares the relative distance between the estimated *mtry* and the optimal *mtry*, where the relative distance between the optimal *mtry* and the default *mtry* is added above as a reference. The two subfigures clearly show that the estimated *mtry* is closer to the optimal *mtry* than the default, and, in 23 out of 56 datasets, the estimated *mtry* is exactly matched to the optimal *mtry*. It demonstrates that the proposed search algorithm performs well to estimate the optimal *mtry*. Second, Figure 6 compares the relative RMSE between the estimated *mtry* and the optimal *mtry*. Since almost all boxes are located near 0, it seems that the estimated *mtry* and the optimal *mtry* produce similar RF prediction performances.

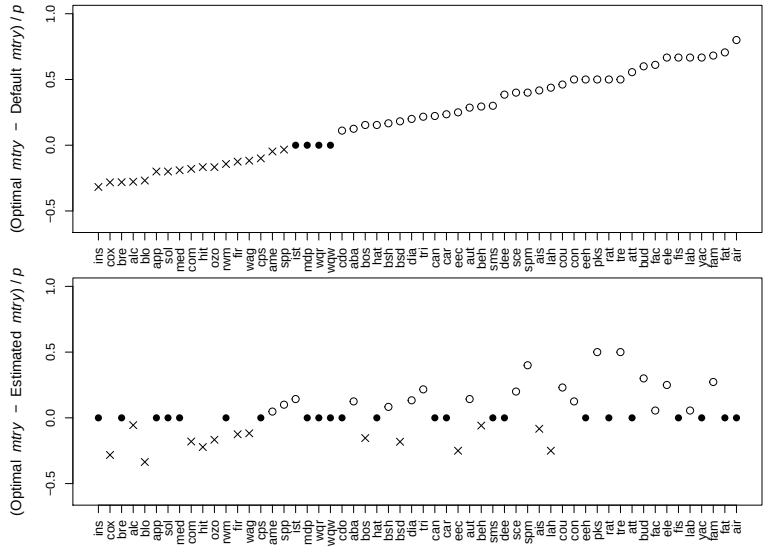

Figure 5. (**Top**) Comparison between the optimal *mtry* and the default *mtry*; and (**Bottom**) comparison between the estimated *mtry* by the proposed search algorithm and the optimal *mtry*. The *x*-axis and *y*-axis indicate 56 datasets and the relative distance, respectively.

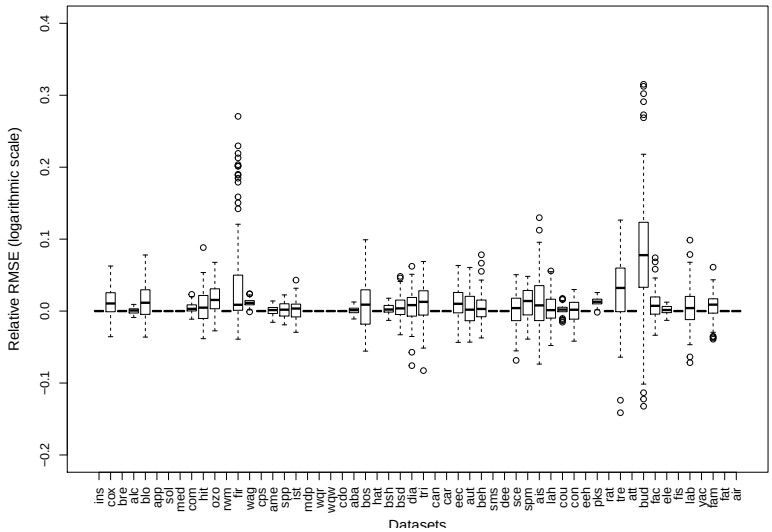

Figure 6. Comparison of the RF prediction performances using the estimated *mtry* by the proposed search algorithm and the optimal *mtry*. The *x*-axis and *y*-axis indicate 56 datasets and the relative RMSE, respectively. Each box-plot is based on 100 relative RMSEs.

4.2. Estimated Size vs. Default Size

Figure 7 compares the relative RMSE between the estimated *mtry* and the default. The results show that using the estimated *mtry* offers improvement over the default size in terms of the RF prediction because many boxes are located above 0. Interestingly, the distribution of the boxes is quite similar with that in Figure 3. It confirms that the optimal *mtry* and the estimated *mtry* have similar performance. We further conducted the paired t-tests based on the 100 RMSE pairs of the estimated *mtry* and the default *mtry* for each dataset. We observed that the RF prediction using the estimated *mtry* is statistically better than that using the default *mtry* on 38 out of 56 datasets, and, for the other 18 datasets, their differences are not statistically significant. There is no dataset where the default *mtry* produces better prediction than the estimated *mtry*.

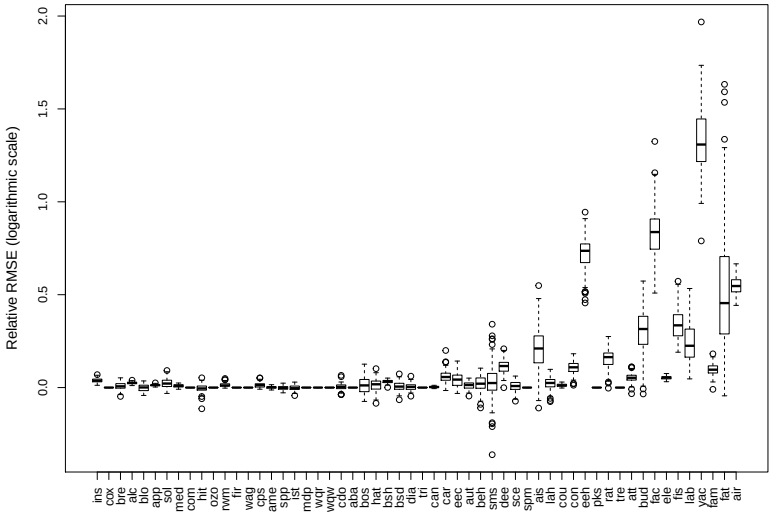

Figure 7. Comparison of RF prediction using the estimated size by the proposed search algorithm and the default size of $p/3$. The x-axis and y-axis indicate 56 datasets and the relative RMSEs, respectively. Each box-plot is based on 100 relative RMSEs.

4.3. Comparison between the Proposed Search and Typical Grid Search

Figure 8 compares the estimated *mtry* by the proposed search algorithm and the one by the typical grid search method in terms of prediction accuracy and computational cost. The typical grid search method can be considered as a special version of the proposed search algorithm, where the searches are always performed from $mtry = 1$ to p with the *Searchsize* of p. The results demonstrate that, while the RF prediction accuracies using the two estimated sizes are not different because almost all boxes are located near 0, the proposed search algorithm is much faster than the typical grid search method.

5. Conclusions

We focused on the influence of *mtry* on the RF prediction in regression problems because most existing studies about *mtry* have been primarily conducted focusing on classification problems. Using many datasets, a total of 56 real or artificial datasets, we carried out several experiments and observed that *mtry* has a significant impact on the RF regression prediction. In addition, the default *mtry* of $p/3$ cannot guarantee the highest accuracy in regression.

We also found that it is difficult to find a specific rule for estimating the best *mtry* since the optimal size seems domain dependent. Fortunately, we found a relationship between the optimal size and characteristics of datasets. If the features of the dataset have similar relevance to each other, the optimal size tends to be relatively small, thus smaller *mtry* values are preferred. In the opposite case, larger *mtry* values are preferred.

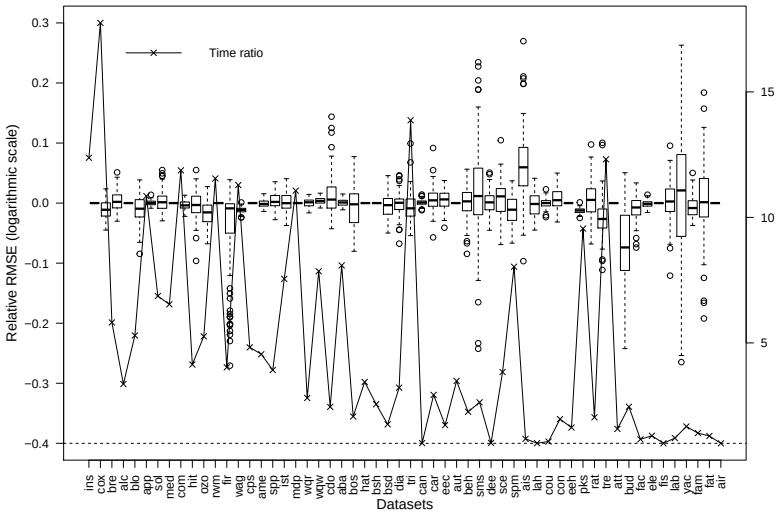

Figure 8. Comparison between the proposed search algorithm and the typical grid search method in terms of the relative RMSE and the relative computing time. The x-axis represents 56 datasets. The y-axis on the left is log(relative RMSE), which the y-axis on the right is the relative computing time, which is defined as log(Computing time of the proposed method/Computing time of the grid search algorithm).

We developed a search algorithm for estimating the optimal *mtry* in RF regression, which is motivated from a method for RF classification [16]. The proposed search algorithm combines a typical grid search method and two unique concepts of "*SearchDirection*" and "*SearchSize*". The two additional concepts allow our proposed search algorithm to estimate the best *mtry* more efficiently than the typical grid search method. In the experimental studies, we demonstrated that the estimated size by the proposed algorithm is close to the optimal size and produces better prediction accuracy than the default size. We also confirmed that the proposed algorithm provides similar prediction accuracy to the typical grid search method, but it is more efficient in terms of computational cost.

Author Contributions: Conceptualization, S.H. and H.K.; Methodology, S.H. and H.K.; Formal analysis, S.H.; Supervision, H.K.; Manuscript writing, S.H. and H.K.; and Manuscript editing, S.H. and H.K. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by Basic Science Research program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science, and Technology (No. 2016R1D1A1B02011696).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data used in this study are available at https://github.com/shan-stat/rf_reg, (accessed on 12 March 2021).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
2. Hastie, T.; Tibshirani, R.; Friedman, J. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*; Springer: New York, NY, USA, 2001.
3. James, G.; Witten, D.; Hastie, T.; Tibshirani, R. *An Introduction to Statistical Learning*; Springer: New York, NY, USA, 2013.
4. Sun, G.; Li, S.; Cao, Y.; Lang, F. Cervical Cancer Diagnosis based on Random Forest. *Int. J. Perform. Eng.* **2017**, *13*. [CrossRef]
5. Dai, B.; Chen, R.; Zhu, S.; Zhang, W. Using random forest algorithm for breast cancer diagnosis. In Proceedings of the International Symposium on Computer, Consumer and Control (IS3C), Taichung, Taiwan, 6–8 December 2018; pp. 449–452.
6. Fang, X.; Liu, W.; Ai, J.; He, M.; Wu, Y.; Shi, Y.; Shen, W.; Bao, C. Forecasting incidence of infectious diarrhea using random forest in Jiangsu Province, China. *BMC Infect. Dis.* **2020**, *20*, 1–8. [CrossRef] [PubMed]

7. Kamal, S.; Urata, J.; Cavassini, M.; Liu, H.; Kouyos, R.; Bugnon, O.; Wang, W.; Schneider, M. Random forest machine learning algorithm predicts virologic outcomes among HIV infected adults in Lausanne, Switzerland using electronically monitored combined antiretroviral treatment adherence. *AIDS Care* **2020**, *33*, 530–560. [CrossRef] [PubMed]
8. Moorthy, K.; Mohamad, M. Random forest for gene selection and microarray data classification. In Proceedings of the Third Knowledge Technology Week, Kajang, Malaysia, 18–22 July 2011; pp. 174–183.
9. Anaissi, A.; Kennedy, P.J.; Goyal, M.; Catchpoole, D.R. A balanced iterative random forest for gene selection from microarray data. *BMC Bioinform.* **2013**, *14*, 1–10. [CrossRef] [PubMed]
10. Probst, P.; Wright, M.N.; Boulesteix, A.L. Hyperparameters and tuning strategies for random forest. *Wires. Data. Min. Knowl.* **2019**, *9*, e1301. [CrossRef]
11. Oshiro, T.M.; Perez, P.S.; Baranauskas, J.A. How many trees in a random forest? In Proceedings of the International Workshop on Machine Learning and Data Mining in Pattern Recognition, Berlin, Germany, 13–20 July 2012; pp. 154–168.
12. Probst, P.; Boulesteix, A.L. To tune or not to tune the number of trees in random forest. *J. Mach. Learn. Res.* **2017**, *18*, 6673–6690.
13. Han, S.; Kim, H.; Lee, Y. Double random forest. *Mach. Learn.* **2020**, *109*, 1569–1586. [CrossRef]
14. Bernard, S.; Heutte, L.; Adam, S. Influence of hyperparameters on random forest accuracy. In Proceedings of International Workshop on Multiple Classifier Systems (MCS), Reykjavik, Iceland, 10–12 June 2009; pp. 171–180.
15. Goldstein, B.A.; Polley, E.C.; Briggs, F.B.S. Random forests for genetic association studies. *Stat. Appl. Genet. Mol.* **2001**, *10*. [CrossRef]
16. Han, S.; Kim, H. On the Optimal Size of Candidate Feature Set in Random forest. *App. Sci.* **2019**, *9*, 898. [CrossRef]
17. Martin, P.; David, E.G.; Erick, C.-P. BOA: The Bayesian optimization algorithm. In *Proceedings of the 1st Annual Conference on Genetic and Evolutionary Computation—Volume 1*; Morgan Kaufmann Publishers Inc.: San Francisco, CA, USA, 1999; pp. 525–532.
18. Wu, J.; Chen, X.Y.; Zhang, H.; Xiong, L.D.; Lei, H.; Deng, S.H. Hyperparameter optimization for machine learning models based on Bayesian optimization. *J. Electron. Sci. Technol.* **2019**, *17*, 26–40.
19. Fiorentini, N.; Maboudi, M.; Leandri, P.; Losa, M.; Gerke, M. Surface Motion Prediction and Mapping for Road Infrastructures Management by PS-InSAR Measurements and Machine Learning Algorithms. *Remote Sens.* **2020**, *12*, 3976. [CrossRef]
20. Breiman, L. Bagging Predictors. *Mach. Learn.* **1996**, *24*, 123–140. [CrossRef]
21. Liaw, A.; Wiener, M. Classification and Regression by randomForest. *R News* **2002**, *2*, 18–22.
22. Dua, D.; Graff, C. *UCI Machine Learning Repository*; University of California, Irvine: Irvine, CA, USA, 2007. Available online: http://archive.ics.uci.edu/ml/index.php (accessed on 12 March 2021).
23. Cook, R.D.; Weisberg, S. *An Introduction to Regression Graphics*; John Wiley & Sons: Hoboken, NJ, USA, 2009.
24. Kenkel, D.S.; Terza, J.V. The effect of physician advice on alcohol consumption: Count regression with an endogenous treatment effect. *J. Appl. Econom.* **2001**, *16*, 165–184. [CrossRef]
25. Chattopadhyay, S. A repeated sampling technique in assessing the validity of benefit transfer in valuing non-market goods. *Land Econ.* **2003**, *79*, 576–596. [CrossRef]
26. Cochran, J.J. Data management, exploratory data analysis, and regression analysis with 1969–2000 major league baseball attendance. *J. Stat. Educ.* **2002**, *10*. [CrossRef]
27. Bollino, C.A.; Perali, F.; Rossi, N. Linear household technologies. *J. Appl. Econom.* **2000**, *15*, 275–287. [CrossRef]
28. Denman, N.; Gregory, D. *Analysis of Sugar Cane Yields in the Mulgrave Area, for the 1997 Sugar Cane Season*; MS305 Data Analysis Project; Department of Mathematics, University of Queensland: St Lucia, QLD, Australia, 1998.
29. Bryant, P.G.; Smith, M.A. *Practical Data Analysis: Case Studies in Business Statistics, Richard D*; Irwin Publishing: Homewood, IL, USA, 1995.
30. Harrell, F.E. *Regression Modeling Strategies, with Applications to Linear Models, Survival Analysis and Logistic Regression*; Springer: Berlin/Heidelberg, Germany, 2001.
31. Berndt, E.R. *The Practice of Econometrics: Classic and Contemporary*; Addison-Wesley Publishing Company: Boston, MA, USA,1991.
32. Cochran, J.J. Career records for all modern position players eligible for the major league baseball hall of fame. *J. Stat. Educ.* **2000**, *8*. Available online: http://www.amstat.org/publications/jse (accessed on 12 March 2021)
33. Penrose, K.W.; Nelson, A.G.; Fisher, A.G. Generalized body composition prediction equation for men using simple measurement techniques. *Med. Sci. Sports Exerc.* **1985**, *17*, 189. [CrossRef]
34. Fernández, C.; Ley, E.; Steel, M.F.J. Bayesian modeling of catch in a north-west Atlantic fishery. *J. R. Stat. Soc. C Appl.* **2002**, *51*, 257–280. [CrossRef]
35. Hair, J.F.; Black, W.C.; Babin, B.J.; Anderson, R.E.; Tatham, R.L. *Multivariate Data Analysis*; Prentice Hall: Upper Saddle River, NJ, USA, 1998.
36. Aaberge, R.; Colombino, U.; Strøm, S. Labour supply in Italy: An empirical analysis of joint household decisions, with taxes and quantity constraints. *J. Appl. Econom.* **1999**, *14*, 403–422. [CrossRef]
37. Afifi, A.A.; Azen, S.P. *Statistical Analysis: A Computer Oriented Approach*; Academic Press: Cambridge, MA, USA, 2014.
38. Deb, P.; Trivedi, P.K. Demand for medical care by the elderly: A finite mixture approach. *J. Appl. Econom.* **1997**, *12*, 313–336. [CrossRef]
39. Neter, J.; Kutner, M.H.; Nachtsheim, C.J.; Wasserman, W. *Applied Linear Statistical Models*; Irwin: Chicago, IL, USA, 1996.
40. Rawlings, J.O. *Applied Regression Analysis: A Research Tool*; Wadsworth & Brooks: Pacific Grove, CA, USA, 1988.
41. Torgo, L.F.R.A. Inductive Learning of Tree-Based Regression Models. Ph.D. Thesis, Universidade do Porto, Porto, Portugal, 1999.

42. Schafgans, M.M.A. Ethnic wage differences in Malaysia: Parametric and semiparametric estimation of the Chinese–Malay wage gap. *J. Appl. Econom.* **1998**, *13*, 481–504. [CrossRef]
43. Zhang, H.; Wang, M. Search for the smallest random forest. *Stat. Interface* **2009**, *2*, 381. [PubMed]