Adaptive Random SubSpace Learning (RSSL) Algorithm for Prediction

Mohamed Elshrif   MME4362@RIT.EDU

Rochester Institute of Technology (RIT), 102 Lomb Memorial Dr., Rochester, NY 14623 USA

Ernest Fokoué   EPFEQA@RIT.EDU

Rochester Institute of Technology (RIT), 102 Lomb Memorial Dr., Rochester, NY 14623 USA

Abstract

We present a novel adaptive random subspace learning algorithm (RSSL) for prediction purpose. This new framework is flexible where it can be adapted with any learning technique. In this paper, we tested the algorithm for regression and classification problems. In addition, we provide a variety of weighting schemes to increase the robustness of the developed algorithm. These different wighting flavors were evaluated on simulated as well as on real-world data sets considering the cases where the ratio between features (attributes) and instances (samples) is large and vice versa. The framework of the new algorithm consists of many stages: first, calculate the weights of all features on the data set using the correlation coefficient and F-statistic statistical measurements. Second, randomly draw n samples with replacement from the data set. Third, perform regular bootstrap sampling (bagging). Fourth, draw without replacement the indices of the chosen variables. The decision was taken based on the heuristic subspacing scheme. Fifth, call base learners and build the model. Sixth, use the model for prediction purpose on test set of the data. The results show the advancement of the adaptive RSSL algorithm in most of the cases compared with the synonym (conventional) machine learning algorithms.

1. Introduction

Given a dataset $\mathscr{D} = \{\mathbf{z}_i = (\mathbf{x}_i^\top, \mathbf{y}_i)^\top, \ i = 1, \cdots, n\}$, where $\mathbf{x}_i = (\mathbf{x}_{i1}, \cdots, \mathbf{x}_{ip})^\top \in \mathscr{X} \subset \mathbb{R}^p$ and $\mathbf{y}_i \in \mathscr{Y}$ are realizations of two random variables X and Y respectively, we seek to use the data $\mathscr{D}$ to build estimators $\widehat{f}$ of the underlying func-

tion f for predicting the response Y given the vector X of explanatory variables. In keeping with the standard in statistical learning theory, we will measure the predictive performance of any given function f using the theoretical risk functional given by

$$\mathscr{R}(f) = \mathbb{E}[\ell(Y, f(X))] = \int_{\mathscr{X} \times \mathscr{Y}} \ell(\mathbf{x}, \mathbf{y}) dP(\mathbf{x}, \mathbf{y}), \qquad (1)$$

with the ideal scenario corresponding to the universally best function defined by

$$\widehat{f}^* = \arg\inf_f \{\mathscr{R}(f)\} = \arg\inf_f \{\mathbb{E}[\ell(Y, f(X))]\}. \qquad (2)$$

For classification tasks, the most commonly used loss function is the zero-one loss $\ell(Y, f(X)) = 1_{\{Y \neq f(X)\}}$, for which the theoretical universal best defined in (2) is the Bayes classifier given by $f^*(\mathbf{x}) = \arg\max_{\mathbf{y} \in \mathscr{Y}} \{\Pr[Y = \mathbf{y}|\mathbf{x}]\}$. For regression tasks, the squared loss $\ell(Y, f(X)) = (Y - f(X))^2$ is by far the most commonly used, mainly because of the wide variety of statistical, mathematical and computational benefits it offers. For regression under the squared loss, the universal best defined in (2) is also known theoretically known to be the conditional expectation of Y given X, specifically given by $f^*(\mathbf{x}) = \mathbb{E}[Y|X = \mathbf{x}]$. Unfortunately, these theoretically expressions of the best estimators cannot be realized in practice because the distribution function $P(\mathbf{x}, \mathbf{y})$ of (X, Y) defined on $\mathscr{X} \times \mathscr{Y}$ is unknown. To circumvent this learning challenge, one has to do essentially two foundational thing, namely: (a) choose a certain function class $\mathscr{F}$ (approximation) from which to search for the estimator $\widehat{f}$ of the true but unknown underlying f, (b) specify the empirical version of (1) based on the given sample $\mathscr{D}$, an use that empirical risk as the practical objective function. However, in this paper, we do not directly construct our estimating classification functions from the empirical risk. Instead, we build the estimators using other optimality criteria, and then compare their predictive performances using the average test error AVTE($\cdot$), namely

$$\text{AVTE}(\widehat{f}) = \frac{1}{R} \sum_{r=1}^{R} \left\{ \frac{1}{m} \sum_{t=1}^{m} \ell(\mathbf{y}_{i_t}^{(r)}, \widehat{f}_r(\mathbf{x}_{i_t}^{(r)})) \right\}, \qquad (3)$$

where $\widehat{f}_r(\cdot)$ is the r-th realization of the estimator $\widehat{f}(\cdot)$ built using the training portion of the split of $\mathscr{D}$ into training set and test set, and $\left(\mathbf{x}_{i_t}^{(r)}, \mathbf{y}_{i_t}^{(r)}\right)$ is the t-th observation from the test set at the r-th random replication of the split of $\mathscr{D}$. In this paper, we consider both multiclass classification tasks with response space $\mathscr{Y} = \{1, 2, \cdots, G\}$ and regression tasks with $\mathscr{Y} = \mathbb{R}$, and we focus on learning machines from a function class $\mathscr{F}$ whose members are ensemble learners in the sense of Definition (1). In machine learning, in order to improve the accuracy of a regression function, or a classification function, scholars tend to combine multiple estimators because it has been proven both theoretically and empirically (Tumer & Ghosh, 1995; Tumer & Oza, 1999) that an appropriate combination of good base learners leads to a reduction in prediction error. This technique is known as ensemble learning (aggregation). In spite of the underlying algorithm used, the ensemble learning technique most of the time (on average) outperforms the single learning technique, especially for prediction purposes (van Wezel & Potharst, 2007). There are many approaches of performing ensemble learning. Among these, there are two popular ensemble learning techniques, bagging (Breiman, 1996) and boosting (Freund, 1995). Many variants of these two techniques have been studied previously such as random forest (Breiman, 2001) and AdaBoost (Freund & Schapire, 1997) and applied in a prediction problem. Our proposed method belongs to the subclass of ensemble learning methods known as random subspace learning.

Definition 1 *Given an ensemble $\mathscr{H} = \{h_1, h_2, \cdots, h_L\}$ of base learners $h_l : \mathscr{X} \longrightarrow \mathscr{Y}$, with relative weight $\alpha_l \in \mathbb{R}_+^*$ (usually $\alpha_l \in (0, 1)$ for convex aggregation), the ensemble representation $f^{(L)}$ of the underlying function f is given by the aggregation (weighted sum)*

$$f^{(L)}(\cdot) = \sum_{l=1}^{L} \alpha_l h_l(\cdot). \qquad (4)$$

A question naturally arises as to how the ensemble $\mathscr{H}$ is chosen, and how the weights are determined. Bootstrap Aggregating also known as bagging (Breiman, 1996), boosting (Freund & Schapire, 1996), random forests (Breiman, 2001), and bagging with subspaces (Panov & Dzeroski, 2007) are all predictive learning methods based on the ensemble learning principle for which the ensemble is built from the provided data set $\mathscr{D}$ and the weights are typically taken to be equal. In this paper, we focus on learning tasks involving high dimension low sample size (HDLSS) data, and we further zero-in on those data sets for which the number of explanatory variables p is substantially larger than the sample size n. As our main contribution in this paper, we introduce, develop and apply a new adaptation of the theme of random subspace learning (Ho, 1998) using the traditional multiple linear regression (MLR) model as our base learner in regression and the generalized linear model (GLM) as a base learner in classification. Some applications by nature posses few instances (small n) with large number of features ($p \ggg n$) such as fMRI (Kuncheva et al., 2010) and DNA microarrays (Bertoni et al., 2005) data sets. It is hard for a traditional (conventional) algorithm to build a regression model, or to classify the data set when it possesses a very small instances to features ratio. The prediction problem becomes even more difficult when this huge number of features correlated are highly correlated, or irrelevant for the task of building such a model, as we will show later in this paper. Therefore, we harness the power of our proposed adaptive subspace learning technique to guide the choice/selection of good candidate features from the data set, and therefore select the best base learners, and ultimately the ensemble yielding the lowest possible prediction error. In most typical random subspace learning algorithms, the features are selected according to an *equally likely* scheme. The question then arises as to whether one can *devise a better scheme to choose the candidate features for efficiently with some predictive benefits*. On the other hand, it is interesting to assess *the accuracy of our proposed algorithm under different levels of the correlation of the features*. The answer to this question constitutes one of the central aspect of our proposed method, in the sense *we explore a variety of weighting scheme for choosing the features, most of them (the schemes) based on statistical measures of relationship between the response variable and each explanatory variable*. As the computational section will reveal, the weighting schemes proposed here lead to a substantially improvement in predictive performance of our method over random forest on all but one data set, arguably due to the fact that our method because it leverages the accuracy of the learning algorithm through selecting many good models (*since the weighting scheme allows good variables to be selected more often and therefore leads to near optimal base learners*).

2. Related Work

Traditionally, in a prediction problem, a single model is built based on the training set and the prediction is decided based solely on this single fitted model. However, in bagging, bootstrap samples are taken from the data set, then, for each instance, the model is fitted. Finally, the prediction is made based on the average of all bagged models. Mathematically, the prediction accuracy for the constructed model using bagging outperforms the traditional model and in the worst case it has the same performance. However, it must be said that it depends on the stability of the modeling procedure. It turns out that bagging reduces the variance without affecting the bias, thereby leading to an overall reduction in prediction error, and hence its great appeal. Any

set of predictive models can be used as an ensemble in the sense defined earlier. There are many ensemble learning approaches. These approaches could be categorized into four classes: (1) algorithms that use heterogeneous predictive models such as stacking (Wolpert, 1992). (2) algorithms that manipulate the instances of the data sets such as bagging (Breiman, 1996), boosting (Freund & Schapire, 1996), random forests (Breiman, 2001), and bagging with subspaces (Panov & Dzeroski, 2007). (3) algorithms that maniplulate the features of the data sets such as random forests (Breiman, 2001), random subspaces (Ho, 1998), and bagging with subspaces (Panov & Dzeroski, 2007). (4) algorithms that manipulate the learning algorithm such as random forests (Breiman, 2001), neural networks ensemble (Hansen & Salamon, 1990), and extra-trees ensemble (Geurts et al., 2006). Since our proposed algorithm manipulates both the instances and features of the data sets, we will focus on the algorithms in the second and third categories (Breiman, 1996; 2001; Panov & Dzeroski, 2007; Ho, 1998).

Bagging (Breiman, 1996), or bootstrap aggregating is an ensemble learning method that generates multiple predictive models. These models are based on performing bootstrap replicates of the learning (training) data set and utilizing from each replicate to build a separate predictive model. The bootstrap sample is attained through randomly (uniformly) sampling with replacement from instances of the training data set. The decision is made based on averaging the predictor classifiers in regression task and taking the majority vote in classification task. Bagging tend to decrease the variance and keeps the bias as in the case of a single classifier. The bagging accuracy increases when the applied learner is unstable, which means that for any small fluctuation on the training data set causes large impact on the test data set such as trees (Breiman, 1996). Random forests (Breiman, 2001), is an ensemble learning method that averages the prediction results from multiple independent predictor (tree) models. It also performs bootstrap replicates, like bagging (Breiman, 1996), to construct different predictors. For each node of the tree, randomly selecting subset of the attributes. It is considered to improve over bagging through *de-correlating the trees. Choose the best attribute from the selected subset*. As (Denil et al., 2014) mentions that when building a random tree, there are three issues that should be decided in advance; (1) the leafs splitting method, (2) the type of predictor, and 3- the randomness method. Random subspace learning (Ho, 1998), is an ensemble learning method that constructs base models based on different features. It chooses a subset of features and then learns the base model depending only on these features. The random subspaces reaches the highest accuracy when the number of features is large as well as the number of instances. In addition, it performs good

when there are redundant features on the data set. Bagging subspaces (Panov & Dzeroski, 2007), is an ensemble learning method that combines both the bagging (Breiman, 1996) and random subspaces (Ho, 1998) learning methods. It generates a bootstrap replicates of the training data set, in the same way as bagging. Then, it randomly chooses a subset from the features, in the same manner as random subspaces. It outperforms the bagging and random subspaces. Also, it is found to yield the same performance as random forests in case of using decision tree as a base learner. In the simulation part of this paper, we aim to answer the following research questions: (1) Is the performance of the adaptive random subspace learning (RSSL) better than the performance of single classifiers? (2) What is the performance of the adaptive RSSL compared to the most widely used classifier ensembles? (3) Is there a theoretical explanation as to why adaptive RSSL works well for most of the simulated and real-life data sets? (4) How does adaptive RSSL perform on different parameter settings and with various percentages of the instance-to-feature ratio (IFR)? (5) How does the correlation between features affect the predictive performance of adaptive RSSL?

3. Adaptive RSSL

In this section, we present an adaptive random subspace learning algorithm for the prediction problem. We start with the formulation of the problem, followed by our suggested solution (proposed algorithm) to tackle (handle) it. A crucial step of assessing the candidate features for building the models is explained in detail. Finally, we elucidate the strength of the new algorithm, from a theoretical perspective.

3.1. Problem formulation

As we said earlier our proposed method belongs to the category of random subspace learning where each base learner is constructed using a bootstrap sample and a subset of the original p features. The main difference here is that we use base learners that are typically considered not to lead to any improvement when aggregated, and we also select features using weighting schemes that inspired for the strength of the relationship between each feature and the response (target). Each base learner is driven by the subset $\{j_1^{(l)}, \cdots, j_d^{(l)}\} \subset \{1, 2, \cdots, p\}$ of d variables of predictors that are randomly select to build it, and the subsample $\mathscr{D}^{(l)}$ drawn with replacement from $\mathscr{D}$. For notational convenience, we use vectors of indicator variables to denote these two important quantities. The *sample indicator* $\boldsymbol{\delta}^{(l)} = (\delta_1^{(l)}, \cdots, \delta_n^{(l)}) \in \{0, 1\}^n$, where

$$\delta_i^{(l)} = \left\{ \begin{array}{ll} 1 & \texttt{if } \mathbf{z}_i \in \mathscr{D}^{(l)} \\ 0 & \texttt{otherwise} \end{array} \right.$$

The *variable indicator* $\boldsymbol{\gamma}^{(l)} = (\gamma_1^{(l)}, \cdots, \gamma_p^{(l)}) \in \{0,1\}^p$, where

$$\gamma_j^{(l)} = \begin{cases} 1 & \text{if } j \in \{j_1^{(l)}, \cdots, j_d^{(l)}\} \\ 0 & \text{otherwise} \end{cases}$$

The estimator of the lth base learner can therefore fully and unambiguously denoted by $\widehat{g}^{(l)}(\cdot; \boldsymbol{\gamma}^{(l)}, \boldsymbol{\delta}^{(l)})$ which we refer to as $\widehat{g}^{(l)}(\cdot)$ for notational simplicity. Each $\delta_j^{(\ell)}$ is chosen according to one of the weighting schemes. To increase the strength of the developed algorithm, we introduce a weighting scheme procedure to select the important features, which facilitates building a proper model and leverage the prediction accuracy. Our weighting schemes are

- Correlation coefficient: We measure the strength of the association between each feature vector and the response (target), and take the square of the resulting sample correlation

- F-statistic: For classification tasks especially, we use the observed F-statistic resulting from the analysis of variance with x_j as the response and the class label the treatment group.

Using the ensemble $\left\{\widehat{g}^{(l)}(\cdot), l = 1, \cdots, L\right\}$ we form the ensemble estimator of class Membership as

$$\widehat{f}^{(L)}(\mathbf{x}) = \arg\max_{y \in \mathscr{Y}} \left\{ \sum_{l=1}^{L} \left(\mathbf{1}_{\{y=\widehat{g}^{(l)}(\mathbf{x})\}} \right) \right\},$$

and the ensemble estimator of regression response as

$$\widehat{f}^{(L)}(\mathbf{x}^*) = \frac{1}{L} \sum_{l=1}^{L} \widehat{g}^{(l)}(\mathbf{x}^*).$$

4. Experiments

We used a collection of simulated and real-world data sets for our experiments. In addition, we used real-world data sets from previous papers, which aim to solve the same problem, for comparison purpose. We report the mean square error (MSE) for each individual algorithm and task purposes, i.e., regression, or classification.

4.1. Simulated data sets

We designed our artificial data sets to fit six scenarios based on the factors, which are the dimensionality of the data (number of features), the number of sample size ((number of instances), and the correlation of the data.

4.2. Real data sets

We benefit from the public repository of the UCI University real-life data sets in our paper. For the purposes of consistency and completeness, we choose the real data sets that carries different characteristics in terms of the number of instances and the number of features along with variety of applications. The real data sets can be represented based on the task as follows:
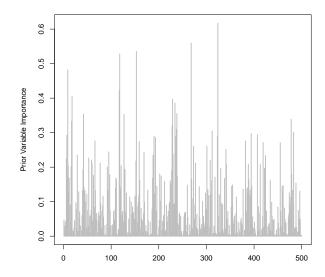

Figure 1. Prior Feature Importance: A representative simulation results for regression analysis on synthetic dataset of scenario with number of instances n=25, number of features p=500, correlation coefficient ρ=0.5, number of learners=450, and number of replications=100.

To elucidate the performance of our developed model, we compare the accuracy of the RSSL with random forest and ... on the same real data sets they used before.

5. Discussion

As revealed (experienced) from our experiments on synthetic data sets that when the number of selected features is higher than 15-20 (for our particular dataset) yields ensemble classifiers that are highly accurate and stable. The reason for this is that only if the number of voters is Òlarge enoughÓ does the random process of attribute selection yield suKcient number of qualitatively different classifiers that ensure high accuracy and stability of the ensemble.

how many bootstrap replications are useful? The evidence both experimental and theoretical is that bagging can push a good but unstable procedure a significant step towards optimality. why the training set in real dataset was chosen to be large and in simulated dataset the test set used to be large? The bootstrap sample was repeated 50 times. The

Table 1. Regression Analysis: Mean Square Error (MSE) for different machine learning algorithms on various scenarios of synthetic data sets.

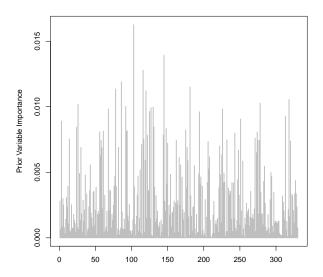| WEIGHTING	N	P	ρ	MLR	UNIFORM MLR	ADAPTIVE MLR	RF	BETTER?
	200	25	0.05	5.69±0.89	14.50±2.63	4.60±0.706	9.81±1.86	√
	200	25	0.5	4.78±0.81	11.67±2.55	4.77±0.94	8.46±1.97	√
CORRELATION	25	200	0.05	974.37±5.E3	18.35±6.92	8.10±3.86	18.56±7.24	√
	25	200	0.5	5.E3±5.E4	18.83±8.72	8.27±5.24	18.18±8.65	√
	50	1000	0.05	2.E4±1.E5	28.36±11.51	12.38±5.91	27.92±11.78	√
	1000	50	0.05	4.66±0.34	16.62±1.37	4.33±0.33	6.73±0.62	√
	200	25	0.05	5.04±0.79	14.42±2.67	4.48±0.74	8.75±1.76	√
	200	25	0.5	4.49±0.76	12.06±2.04	5.51±1.09	8.33±1.59	√
F-STATISTICS	25	200	0.05	3.E4±2.E5	17.77±9.15	5.81±4.10	15.81±8.55	√
	25	200	0.5	1.E4±1.E5	23.09±16.06	12.53±10.27	24.11±16.31	√
	50	1000	0.05	4.E5±3.E6	16.65±5.38	7.65±2.83	15.54±5.31	√
	1000	50	0.05	4.19±0.33	15.97±1.15	3.90±0.30	6.24±0.55	√

Figure 2. Prior Feature Importance: A representative simulation results for classification analysis on real dataset of Lymphoma disease.

Figure 3. A representative results of synthetic dataset of scenario with number of instances n=50, number of features p=1000, correlation coefficient ρ=0.05, number of learners=450, and number of replications=100. We used the correlation weighting scheme for regression analysis on logarithmic scale.

random division of the data is repeated 100 times. Choosing between these two strategies is not an easy task since it involves a trade-off between bias and estimation variance over the forecast horizon.

Even though that our developed adaptive RSSL algorithm outperforms many classifier ensembles. It has limitations where this new algorithm can not deal with data set that has categorical features. Instead it necessities to encode these features numerically. A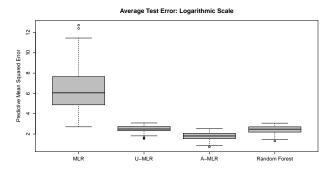lso, the algorithm is not designed to classify data sets with multiple classes. Moreover, the adaptive RSSL algorithms sometimes fails to select the optimal feature subsets?

6. Conclusion and Future Work

We presented a detailed quantitative analysis of the adaptive RSSL algorithm for an ensemble prediction problem. We support this analysis with deep theoretical (mathematical) explanation (formulation). The key important issues for the developed algorithm resides on four fundamental factors: generalization, flexibility, speed, and accuracy. We will explain each of these four factors. We present a rigorous theoretical justification of our propose algorithm. For now, we choose fixed number of attribute subset. However, the algorithm should evaluated based on the performance

Table 2. Regression Analysis: Mean Square Error (MSE) for different machine learning algorithms on real data sets.

DATA SET	WEIGHTING	MLR	UNI. MLR	ADAP. MLR	RF	BETTER?
BODYFAT	CORRELATION	17.41±2.69	23.59±3.71	19.25±3.06	19.72±3.18	×
	F-STATISTICS	17.06±2.50	23.07±3.46	17.46±2.65	19.51±2.99	×
ATTITUDE	CORRELATION	74.12±32.06	80.35±34.40	58.49±20.21	88.72±35.97	√
	F-STATISTICS	75.19±36.63	74.71±33.17	51.84±15.19	82.21±35.58	√
CEMENT	CORRELATION	10.76±7.25	NA	19.92±15.98	75.91±56.05	×
	F-STATISTICS	11.07±8.55	NA	24.27±18.27	62.20±46.53	×
DIABETES 1	CORRELATION	2998.13±322.37	3522.30±311.81	3165.74±300.86	3203.94±311.94	×
	F-STATISTICS	2988.32±341.20	3533.45±375.38	3133.60±324.75	3214.11±318.6931	×
DIABETES 2	CORRELATION	3916.98±782.35	4244.00±390.29	3016.54±285.89	3266.50±324.82	√
	F-STATISTICS	3889.00±679.55	4306.76±419.66	3076.77±338.08	3326.28±382.37	√
LONGLEY	CORRELATION	0.21±0.13	0.62±0.36	0.49±0.29	1.54±0.92	×
	F-STATISTICS	0.22±0.13	0.66±0.42	0.49±0.29	1.63±1.04	×
PRESTIGE	CORRELATION	66.68±15.31	73.32±14.77	64.87±13.93	55.96±11.83	×
	F-STATISTICS	65.77±15.96	72.33±16.64	63.27±14.71	56.02±12.66	×

Table 3. Classification Analysis: MisClassification Rate (MCR) for different machine learning algorithms on various scenarios of simulated data sets.

WEIGHTING	N	P	ρ	GLM	UNI. GLM	ADAP. GLM	RF	BETTER?
F-STATISTICS	200	25	0.05	0.070±0.033	0.486±0.172	0.071±0.032	0.101±0.053	√
	200	25	0.5	0.140±0.045	0.498±0.221	0.138±0.043	0.136±0.058	×
	50	200	0.05	0.102±0.093	0.673±0.123	0.100±0.092	0.320±0.103	√
	50	200	0.5	0.058±0.141	0.346±0.346	0.049±0.121	0.178±0.188	√
	50	1000	0.05	0.033±0.064	0.522±0.158	0.034±0.062	0.409±0.114	√
	1000	50	0.05	0.130±0.019	0.643±0.028	0.130±0.019	0.167±0.024	√

(accuracy) to determine the appropriate number (dimension) for single classifiers used in the ensemble learning. In addition, the adaptive RSSL algorithm is tested on a relatively small data sets. Our next step will be applying the developed algorithm on a big data sets.

Also, we show that the adaptive RSSL performs better than widely used ensemble algorithms even with the dependence of feature subsets.

Computational issues.

References

Bertoni, Alberto, Folgieri, Raffaella, and Valentini, Giorgio. Bio-molecular cancer prediction with random subspace ensembles of support vector machines. *Neurocomputing*, 63(0):535 – 539, 2005. ISSN 0925-2312. doi: http://dx.doi.org/10.1016/j.neucom.2004.07.007. URL http://www.sciencedirect.com/science/article/pii/S0925231204003637. New Aspects in Neurocomputing: 11th European Symposium on Artificial Neural Networks.

Breiman, Leo. Bagging predictors. *Machine Learning*, 24 (2):123–140, 1996. ISSN 0885-6125. doi: 10.1023/A: 1018054314350. URL http://dx.doi.org/10.1023/A%3A1018054314350.
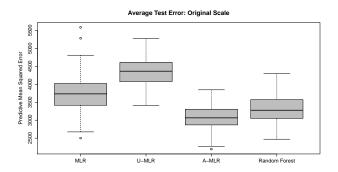
Breiman, Leo. Random forests. *Mach. Learn.*, 45(1):5–32, October 2001. ISSN 0885-6125. doi: 10.1023/A: 1010933404324. URL http://dx.doi.org/10.1023/A:1010933404324.
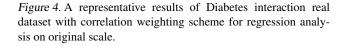
Denil, Misha, Matheson, David, and de Freitas, Nando. Narrowing the gap: Random forests in theory and in practice. In *International Conference on Machine Learning (ICML)*, 2014. URL http://jmlr.org/proceedings/papers/v32/denil14.html.

Freund, Y. Boosting a weak learning algorithm by

Table 4. Classification Analysis: MisClassification Rate (MCR) for different machine learning algorithms on real data sets.

DATA SET	W. S.	GLM	UNI. GLM	ADAP. GLM	RF	BETTER?
DIABETES IN PIMA	F-STAT	0.274±0.071	0.249±0.051	0.255±0.051	0.269±0.050	√
PROSTATE CANCER	F-STAT	0.425±0.113	0.355±0.093	0.332±0.094	0.343±0.098	√
GOLUB LEUKEMIA	F-STAT	0.427±	0.023±	0.021±	0.023±	√
DIABETES	F-STAT	0.034±0.031	0.068±0.039	0.038±0.034	0.031±0.029	×
LYMPHOMA	F-STAT	0.248±0.065	0.057±0.034	0.046±0.029	0.082±0.046	√
LUNG CANCER	F-STAT	0.113±0.051	0.038±0.023	0.037±0.024	0.051±0.030	√
COLON CANCER	F-STAT	0.296±0.124	0.168±0.095	0.124±0.074	0.199±0.106	√

Figure 4. A representative results of Diabetes interaction real dataset with correlation weighting scheme for regression analysis on original scale.

Figure 5. A representative results on synthetic dataset of scenario with number of instances n=200, number of features p=25, correlation coefficient ρ=0.05, number of learners=450, and number of replications=100. We used F-statistics weighting scheme for classification analysis.

majority. *Information and Computation*, 121 (2):256 – 285, 1995. ISSN 0890-5401. doi: http://dx.doi.org/10.1006/inco.1995.1136. URL http://www.sciencedirect.com/science/article/pii/S0890540185711364.

Freund, Yoav and Schapire, Robert E. Experiments with a new boosting algorithm, 1996.

Freund, Yoav and Schapire, Robert E. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119 – 139, 1997. ISSN 0022-0000. doi: http://dx.doi.org/10.1006/jcss.1997.1504. URL http://www.sciencedirect.com/science/article/pii/S002200009791504X.

Geurts, Pierre, Ernst, Damien, and Wehenkel, Louis. Extremely randomized trees. *Machine Learning*, 63 (1):3–42, 2006. ISSN 0885-6125. doi: 10.1007/s10994-006-6226-1. URL http://dx.doi.org/10.1007/s10994-006-6226-1.

Hansen, L. K. and Salamon, P. Neural network ensembles. *IEEE Trans. Pattern Anal. Mach. Intell.*, 12(10):993–1001, October 1990. ISSN 0162-8828. doi: 10.1109/
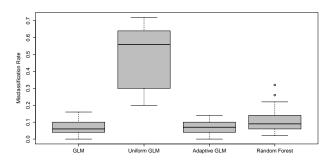
34.58871. URL http://dx.doi.org/10.1109/34.58871.

Ho, Tin Kam. The random subspace method for constructing decision forests. *IEEE Trans. Pattern Anal. Mach. Intell.*, 20(8):832–844, August 1998. ISSN 0162-8828. doi: 10.1109/34.709601. URL http://dx.doi.org/10.1109/34.709601.

Kuncheva, L.I., Rodriguez, J.J., Plumpton, C.O., Linden, D.E.J., and Johnston, S.J. Random subspace ensembles for fmri classification. *Medical Imaging, IEEE Transactions on*, 29(2):531–542, Feb 2010. ISSN 0278-0062. doi: 10.1109/TMI.2009.2037756.

Panov, Pance and Dzeroski, Saso. Combining bagging and random subspaces to create better ensembles. In R. Berthold, Michael, Shawe-Taylor, John, and Lavrafc, Nada (eds.), *Advances in Intelligent Data Analysis VII*, volume 4723 of *Lecture Notes in Computer Science*, pp. 118–129. Springer Berlin Heidelberg, 2007. ISBN 978-3-540-74824-3. doi: 10.1007/
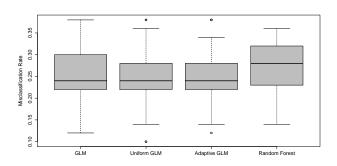
Figure 6. A representative results of the Diabetes in Pima Indian Women real dataset with F-statistics weighting scheme for classification analysis.

Table 5. Summary of the regression and classification real data sets.

DATA SET	# INST.	# FEAT.	IFR RATIO
REGRESSION			
BODYFAT	252	14	1,800%
ATTITUDE	30	7	428.5%
CEMENT	13	5	260%
DIABETES 1	442	11	4,018%
DIABETES 2	442	65	680%
LONGLEY	16	7	228.5%
PRESTIGE	102	5	2,100%
CLASSIFICATION			
DIABETES IN PIMA	200	8	2,500%
PROSTATE CANCER	79	501	15.8%
LEUKEMIA	72	3572	2.0%
DIABETES	145	4	3,625%
LYMPHOMA	180	662	27.2%

978-3-540-74825-0_11. URL http://dx.doi.org/10.1007/978-3-540-74825-0_11.

Tumer, Kagan and Ghosh, Joydeep. Classifier combining: Analytical results and implications. In *In Proceedings of the AAAI-96 Workshop on Integrating Multiple Learned Models for Improving and Scaling Machine Learning Algorithms*, pp. 126–132. AAAI Press, 1995.

Tumer, Kagan and Oza, Nikunj C. Decimated input ensembles for improved generalization, 1999.

van Wezel, Michiel and Potharst, Rob. Improved customer choice predictions using ensemble methods. *European Journal of Operational Research*, 181(1):436 – 452, 2007. ISSN 0377-2217. doi: http://dx.doi.org/10.1016/j.ejor.2006.05.029. URL http://www.sciencedirect.com/science/article/pii/S0377221706003900.
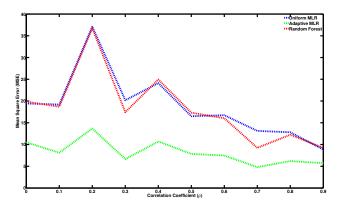
Figure 7. A representative results that exhibits the relationship between mean square error (MSE) and correlation coefficient (ρ) for different algorithms on synthetic dataset with correlation weighting scheme for regression analysis when p≫n.
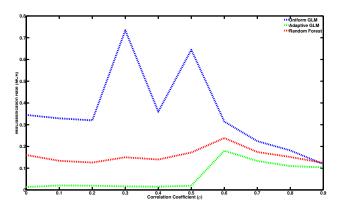

Figure 8. A representative results that exhibits the relationship between mean square error (MSE) and correlation coefficient (ρ) for different algorithms on synthetic dataset with F-statistics weighting scheme for classification analysis when n≫p.

Wolpert, David H. Stacked generalization. *Neural Networks*, 5:241–259, 1992.