

Geometry- and Accuracy-Preserving Random Forest Proximities

Jake S. Rhodes, Adele Cutler, and Kevin R. Moon

Abstract—Random forests are considered one of the best out-of-the-box classification and regression algorithms due to their high level of predictive performance with relatively little tuning. Pairwise proximities can be computed from a trained random forest which measure the similarity between data points relative to the supervised task. Random forest proximities have been used in many applications including the identification of variable importance, data imputation, outlier detection, and data visualization. However, existing definitions of random forest proximities do not accurately reflect the data geometry learned by the random forest. In this paper, we introduce a novel definition of random forest proximities called Random Forest-Geometry- and Accuracy-Preserving proximities (RF-GAP). We prove that the proximity-weighted sum (regression) or majority vote (classification) using RF-GAP exactly match the out-of-bag random forest prediction, thus capturing the data geometry learned by the random forest. We empirically show that this improved geometric representation outperforms traditional random forest proximities in tasks such as data imputation and provides outlier detection and visualization results consistent with the learned data geometry.

Index Terms—Random Forests; Proximities; Supervised Learning

1 INTRODUCTION

RANDOM forests [1] are well-known, powerful predictors comprised of an ensemble of binary recursive decision trees. Random forests are easily adapted for both classification and regression, are trivially parallelizable, can handle mixed variable types (continuous and categorical), are unaffected by monotonic transformations, are insensitive to outliers, scale to small and large datasets, handle missing values, are capable of modeling non-linear interactions, and are robust to noise variables [1], [2]. Random forests are simple to use, produce good results with little to no tuning, and can be applied to a wide-variety of fields. Citing some recent examples, Benali et al. [3] demonstrated superior solar radiation prediction using random forests over neural networks. The work of [4] showed that random forests produced the best results with lowest standard errors in classifying error types in rotating machinery when compared with more commonly used models in this application, such as the SVM and neural networks. Other recent successes include patient health prediction after exposure to COVID-19 [5], spatio-temporal COVID-19 case estimation [6], landslide susceptibility mapping [7], infectious diarrhea forecasting [8], cardiovascular disease prediction [9], deforestation rate prediction [10], nanofluid viscosity estimation [11], rural credit assessment [12], RNA pseudouridine site prediction [13], wearable-sensor activity classification [14], and heavy metal distribution estimation in agricultural soils [15].

In addition to their high predictive-power, random forests have a natural extension to produce pair-wise proximity (similarity) measures determined by the partitioning space of the decision trees which comprise them. The ran-

dom forest proximity measure between two observations was first defined by Leo Breiman as the proportion of trees in which the observations reside in the same terminal node [16]. As splitting variable values are determined to optimize partitions according to the supervised task, these proximities encode a supervised similarity measure.

Many machine learning methods depend on some measure of pairwise similarity (which is usually unsupervised) including dimensionality reduction methods [17], [18], [19], [20], [21], [22], [23], spectral clustering [24], and any method involving the kernel trick such as SVM [25] and kernel PCA [26]. Random forest proximities can be used to extend many of these problems to a supervised setting and have been used for data visualization [27], [28], [29], [30], [31], outlier detection [30], [32], [33], [34], and data imputation [35], [36], [37], [38]. See Section 2 for further discussion of random forest proximity applications.

Unlike unsupervised similarity measures, random forest proximities incorporate variable importance relative to the supervised task as these variables are more likely to be used in determining splits in the decision trees [2]. Ideally, random forest proximities should define a data geometry that is consistent with the learned random forest; that is, the random forest predictive ability should be recoverable from the proximities. In this case, applications involving random forest proximities, such as data visualization, can lead to improved interpretability of the random forests specifically, and more generally the data geometry relative to the supervised task.

One way to test for consistency is to compute a proximity-weighted predictor where a data point’s predicted label consists of a proximity-weighted sum of the labels of all other points. This predictor should match the random forest prediction if the proximities are consistent with the random forest. However under Breiman’s original definition, the proximity-weighted predictions do not match

• E-mail: jake.rhodes@usu.edu, adele.cutler@usu.edu, kevin.moon@usu.edu

• Department of Mathematics and Statistics, Utah State University.

those of the random forest, even when applied to the training data (see Section 5). Thus this definition does not capture the data geometry learned by the random forest, limiting its potential for improved interpretability of the random forest.

We define a new random forest proximity measure called Random Forest-Geometry- and Accuracy-Preserving proximities (RF-GAP) that defines a data geometry such that the proximity-weighted predictions exactly match those of the random forest for both regression and classification. Under our definition, an out-of-bag observation’s proximities are computed via in-bag (training) observations. That is, the sample used to generate a decision tree also generates the proximities of out-of-bag observations (observations not used to construct the tree). The proximities of an out-of-bag observation is the mean reciprocal of the number of in-bag observation in its shared terminal nodes. We prove the equivalence between the proximity-weighted predictions with those of the random forest and demonstrate this empirically. We then compare RF-GAP proximities with existing random forest proximities in common applications: data imputation, visualization, and outlier detection. In all cases, RF-GAP outperforms existing definitions, showing that it better captures the geometry learned by the random forest.

2 RANDOM FOREST PROXIMITY APPLICATIONS

Random forest proximities have been used in many applications. One common usage is visualization or clustering. Unsupervised random-forest clustering was introduced by Shi and Horvath in [39]. In this paper, they applied both classical and metric multi-dimensional scaling (MDS) to a number of datasets. Pouyan et al. used unsupervised random forest proximity matrices to generate 2D plots for visualization using t-SNE [27]. This approach was used to visualize two Mass cytometry data sets and provided clearer visual results over other compared distance metrics (Euclidean, Chebyshev, and Canberra). Similarly, [29] used an unsupervised random forest for clustering tumor profilings and showed improvement over other common clustering algorithms with microarray data. Random forest proximities have also been used in the supervised setting for visualizing the data, typically via MDS [30], [40], but other approaches have been used [31]. However, these approaches use a proximity definition which does not match the data geometry learned by the random forest. Thus the visualizations do not give a true representation of this geometry.

Random forest proximities have also been used in outlier detection in a supervised setting. An observation’s outlier score is typically defined to be inversely proportional to its average within-class proximity measure (see Section 6.3 for details). This approach was used to achieve better random forest error rates in both classification and regression in pathway analysis [30]. Nesa et al. demonstrated the superiority of random forests for detecting errors and events across internet of things (IOT) device sensors over other multivariate outlier detection methods [41]. The random forest outlier detection algorithm has been effective in other contexts such as modeling species distribution [42], detecting food adulteration via infrared spectroscopy [43], predicting galaxy spectral measurements [44], and detecting network anomalies [45].

Random forest proximities are used to impute missing data by replacing missing values of a given variable with a proximity-weighted sum of non-missing values. Pantanowitz and Marwala used this approach to impute missing data in the context of HIV seroprevalence [37]. They compared their results with five additional imputation methods, including neural networks, and random forest-neural network hybrids and concluded that random forest imputation produced the most accurate results with the lowest standard errors. Shah et al. compare random forest imputation with multivariate imputation by chained equations (MICE [46]) on cardiovascular health records data [38]. They showed that random forest imputation methods typically produced more accurate results and that in some circumstances MICE gave biased results under default parameterization. In [35] it was similarly shown that random forests produced the most accurate imputation in a comprehensive metabolomics imputation study.

Variable importance assessment is another application of random forest proximities. One approach is to use differences in proximity measures as a criterion for assessing variable permutation importance [47]. Variable selection criteria were compared across various data sets and some improvement over existing variable importance measures was achieved. In [48], feature contributions to the random forest decision space (defined by proximities) are explored. While many measures of variable importance are generally computed at a global level, the author’s propose a bit-wise, permutation-based feature importance which captures both the contribution (influence of the feature in the decision space) and closeness (position in the decision space relative to the in- and out-class), giving further insight to each features contribution at the terminal node level.

Multi-modal/multiview problems can also be approached using random forest proximities. Gray et al. additively combined random forest proximities from different modalities or views (FDG-PET and MR imaging) of persons with Alzheimer’s disease or with mild cognitive impairment. They applied MDS to the combined proximities to create an embedding used for classification. Classification on the multi-modal embedding showed significantly better results than classification on both modes separately [28]. Cao, Bernard, Sabourin, and Heutte explored variations of proximity-based classification techniques in the context of multi-view radiomics problems [49]. They compared with the work from [28] and joined proximity matrices using linear combinations. The authors explored random forest parameters to determine the quality of the proximity matrices. They concluded that a large number of maximum-depth trees produced the best quality proximities, quantified using a one-nearest neighbor classifier. Using our proposed random forest proximity measure which accurately reflects the random forest predictions from each view may add to the success of this method, thus creating a truer forest ensemble for multi-view learning.

Seoane et al. proposed the use of random forest proximities to measure gene annotations with some improvement in precision over other existing methods [50]. Zhao et al. presented two matching methods for observational studies, one propensity-based and the other random forest-proximity-based [51]. For the random-forest approach, sub-

jects of different classes were iteratively matched based on nearest proximity values. They showed that the proximity-based matching was superior to the propensity matching in addition to other existing matching techniques.

All of these applications used existing definitions of random forest proximities that do not match the data geometry learned by the random forest. In contrast, RF-GAP accurately reflects this geometry. Thus, the applications presented here should show improvement using our definition. Indeed, we show experimentally in Sections 5 that using RG-GAP gives data visualizations that more accurately represent the geometry learned from the random forests, outlier scores that are more reflective of the random forest’s learning, and improved random forest imputations.

3 RANDOM FOREST PROXIMITIES

Here we provide several existing definitions of random forest proximities followed by RF-GAP that preserves the geometry learned by the random forest. Let $\mathcal{M} = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_N, y_N)\}$ be the training data where each $\mathbf{x}_i \in \mathcal{X}$ is a d -dimensional vector of predictor variables with corresponding response, $y_i \in \mathcal{Y}$. We will use the following notation (see Fig. 1 for a visual example):

- \mathcal{T} is the set of decision trees in a random forest with $|\mathcal{T}| = T$.
- $B(t)$ is the multiset of indices in the bootstrap sample of the training data that is randomly selected to train the tree $t \in \mathcal{T}$. Thus $B(t)$ contains the indices of the in-bag observations.
- $O(t) = \{i = 1, \dots, n | i \notin B(t)\}$. Thus $O(t)$ is the set of indices of the training data that are not contained in $B(t)$. $O(t)$ is often referred to as the out-of-bag (OOB) sample.
- $S_i = \{t \in \mathcal{T} | i \in O(t)\}$. This is the set of trees in which the observation i is OOB.
- $v_i(t)$ contains the indices of all observations that end up in the same terminal node as \mathbf{x}_i in tree t .
- $J_i(t) = v_i(t) \cap B(t)$. This is the set of indices in $v_i(t)$ that correspond with the in-bag observations of t . I.e. these are the observations that are in-bag and end up in the same terminal node as \mathbf{x}_i .

Each decision tree t in a random forest is grown by recursively partitioning (splitting) the bootstrap sample into nodes, where splits are determined across a subset of feature variables to maximize purity (classification) or minimize the mean squares of the residuals (regression) in the resulting nodes. This process repeats until a stopping criterion is met. For classification, splits are typically continued until nodes are pure (one class). For regression, a common stopping criterion is a predetermined minimum node size (e.g. 5). The trees in random forests are typically not pruned. OOB samples are commonly used to provide an unbiased estimate of the forest’s generalization error rate [1].

The strength of the random forest is highly dependent on the predictive power of the individual decision trees (base learners) and on low correlation between the decision trees [16], [52]. In addition to bootstrap sampling, further correlational decrease between trees is ensured by selecting a random subset of predictor variables at each node for split optimization. The number of variables to be considered is

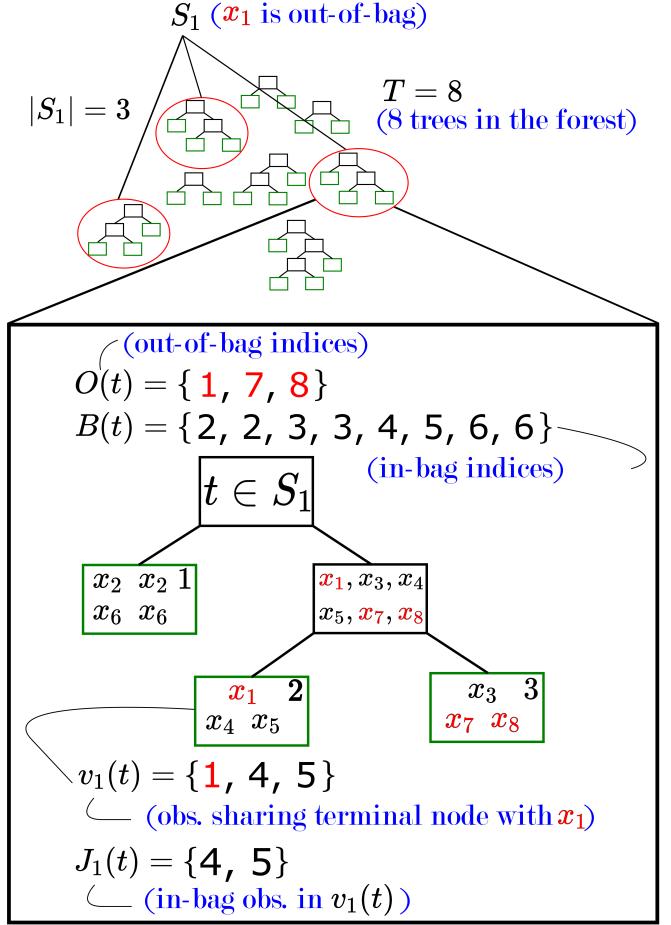


Fig. 1. An example of a random forest and notation with regards to a particular observation x_1 . The red-encircled trees are those in which x_1 is out of bag, making up the set of trees S_1 . A particular tree in S_1 is exhibited. The out-of-bag indices for the tree are given in red ($i \in O(t)$), while the in-bag indices ($i \in B(t)$) are shown in black. The indices of observations residing in the same terminal node as x_1 is given by the set $v_1(t)$. $J_1(t)$ gives the in-bag observation indices in the terminal node $v_1(t)$.

designated by the parameter `mtry` in many random forest packages [53], [54]. The resulting terminal nodes partition the input space \mathcal{X} . This partition is often used in defining random forest proximities as in Breiman’s original definition:

Definition 1 (Original Random Forest Proximity [16]). *The random forest proximity between observations i and j is given by:*

$$p_{Or}(i, j) = \frac{1}{T} \sum_{t=1}^T I(j \in v_i(t)),$$

where T is the number of trees in the forest, $v_i(t)$ contains the indices of observations that end up in the same terminal node as \mathbf{x}_i in tree t , and $I(\cdot)$ is the indicator function. That is, the proximity between observations i and j is the proportion of trees in which they reside in the same terminal node, regardless of bootstrap status.

Definition 1 (the original definition) does not capture the data geometry learned by the random forest as it does not take an observation’s bootstrap status (whether or not the observation was used in the training of any particular

tree) into account in the proximity calculation: both in-bag and out-of-bag samples are used. In-bag observations of different classes will necessarily terminate in different nodes (as trees are grown until pure). Thus this produces an over-exaggerated class separation in the proximity values.

Despite its weaknesses, this definition has been used for outlier detection, data imputation, and visualization. However, these applications may produce misleading results as this definition tends to overfit the training data, quantified by low error rates as proximity-weighted predictors. One attempt to overcome this issue redefines the proximity measure between observations i and j using only trees in which both observations are out-of-bag (OOB proximities):

Definition 2 (OOB Proximity [53], [55]). *The OOB proximity between observations with indices i and j is given by:*

$$p_{OOB}(i, j) = \frac{\sum_{t \in S_i} I(j \in O(t) \cap v_i(t))}{\sum_{t \in S_i} I(j \in O(t))},$$

where $O(t)$ denotes the set of indices of the observations that are out-of-bag in tree t , and S_i is the set of trees for which observation i is out-of-bag. In other words, this proximity measures the proportion of trees in which observations i and j reside in the same terminal node, both being out-of-bag.

Definition 2 [55] is currently used in the randomForest [53] package by Liaw and Wiener in the R programming language [56]. It may have been inadvertently used in papers which used this package but none made explicit mention of the use of OOB observations in building proximities. However, we find that this definition also does not characterize the random forest and generally produces higher error rates as a proximity-weighted predictor (when compared to the random forest's OOB error rate; see Section 5).

Several alternative random forest proximity measures beyond Definitions 1 and 2 have been proposed previously. In each case, source code has not been provided. In [57], the authors define a proximity-based kernel (PBK) which accounts for the number of branches between observations in each decision tree, defining the proximity between i and j as $p_{PBK}(i, j) = \frac{1}{T} \sum_{t=1}^T \frac{1}{e^{w \cdot g_{ijt}}}$, where T is the number of trees in the forest, w is a user-defined parameter, and g_{ijt} is the number of branches between observations i and j in tree t . g is defined to be 0 if the observations reside in the same terminal node. The proximity quality was quantified using the classification accuracy when applied as a kernel in a support-vector machine. This definition showed some improvement over the original definition (Definition 1) only when considering very small numbers of trees (5 or 10). However, PBK is computationally expensive as all pair-wise branch distances must be computed within each tree. This is not an issue for small numbers of trees, but typically the number of trees in a random forest is measured in hundreds or thousands (randomForest [53] has a default of 500 trees). Additionally, this method adds a user-defined, tunable parameter which adds to its complexity.

The authors in [58] describe an approach for computing random forest proximities in the context of a larger class of Random Partitioning Kernels. While most random forest proximities are determined primarily through associations

within terminal nodes, this approach selects a random tree height and partitions the data based on this higher-level splitting. The authors do not compare with other proximity definitions (nor do they frame their work in the context of random forest proximities) but they compare this random forest kernel to other typical kernels (linear, RBF, etc.) using a log-likelihood test. The random forest kernel outperformed the others in most cases and the authors visually demonstrated their kernel using 2D PCA plots. The code for this approach is not publicly available.

Cao et al. introduced two random forest dissimilarity measures which are used in the context of multi-view classification [59]. The first measure (denoted RFDisNC) weights the proximity values by the proportion of correctly-classified observations within each node, accounting for both in- and out-of-bag observations. The second (RFDisIH) is based on instance hardness. Euclidean distances between observations at each terminal node are calculated (using only feature variables which were used as splitting variables leading to the terminal node, to avoid the curse of dimensionality) and used as weights as a part of the dissimilarity measure. Given this distance, they use k -Disagreeing Neighbors (DN) in the formulation of the dissimilarity measure:

$$d_t(\mathbf{x}_i, \mathbf{x}_j) = \begin{cases} kDN(\mathbf{x}_j), & \text{if } v_i(t) = v_j(t) \\ 1, & \text{otherwise} \end{cases},$$

where

$$kDN(\mathbf{x}_i) = \frac{|\{\mathbf{x}_j : \mathbf{x}_j \in kNN(\mathbf{x}_i), y_j \neq y_i\}|}{k},$$

where $kNN(\mathbf{x}_i)$ is the set of the k -nearest neighbors of \mathbf{x}_i in the training data.

The use of k -DN gives a notion of difficulty in classifying a particular observation. In the multiview problem, the dissimilarities from different views are averaged before classification. The authors showed that RFDisIH performed better overall on classification tasks compared with other multi-view methods. However, RFDisIH was not compared with other random forest proximities in their commonly-used applications (e.g. visualization or imputation). A similarity measure can be constructed from RFDisIH as RFProxIH = $1 - \text{RFDisIH}$.

While these alternative definitions have shown promise in their respective applications, their connection to the data geometry learned by the random forest is not clear. In contrast, we present a new definition of random forest proximities that exactly characterizes the random forest performance on both in-bag and out-of-bag samples:

Definition 3 (Random Forest-Geometry- and Accuracy-Preserving Proximities (RF-GAP)). *Let $B(t)$ be the multiset of (potentially repeated) indices of bootstrap (in-bag) observations. We define $J_i(t)$ to be the set of in-bag observations which share the terminal node with observation i in tree t , or $J_i(t) = B(t) \cap v_i(t)$ with cardinality $|J_i(t)|$. Then, for given observations i and j , their proximity measure is defined as:*

$$p_{GAP}(i, j) = \frac{1}{|S_i|} \sum_{t \in S_i} \frac{I(j \in J_i(t))}{|J_i(t)|}.$$

That is, considering only trees for which observation i is out-of-bag, the proximity between i and j is the average proportion of

in-bag observations in the shared terminal node of i and j over all trees where i is out of bag and j is in bag.

This proposed definition is, in part, inspired by the work of Lin and Jeon [60]. We show in Section 4 that the random forest OOB prediction (and thus the generalization error rate) is exactly reproduced as a weighted sum (for regression) or a weighted-majority vote (for classification) using the proximities in Definition 3 as weights. Thus, this definition characterizes the random forest's predictions, keeping intact the learned data geometry. Subsequently, applications using this proximity definition will provide results which are truer to the random forest from which the proximities are derived.

4 RANDOM FORESTS AS PROXIMITY-WEIGHTED PREDICTORS

Here we show that the random forest prediction is exactly reproduced as a weighted sum (for regression) or a weighted-majority vote (for classification) using RF-GAP as weights. We first show that for a given observation, the proximities are non-negative and sum to one. In contrast, the proximities in Definitions 1 and 2 must be row-normalized to sum to one. Note that we require that $p_{GAP}(i, i) = 0$ for the proximities to sum to one, although the exact value for $p_{GAP}(i, i)$ does not matter in practice as it is not considered in the proximity-weighted prediction.

Proposition 1. *Defining $p_{GAP}(i, i) = 0$, the random forest proximities (under Definition 3) are non-negative and $\sum_{j=1}^N p_{GAP}(i, j) = 1$.*

Proof: It is clear from the definition that $p_{GAP}(i, j) \geq 0$ for all i, j . The sum-to-one property falls directly from the definition:

$$\begin{aligned} \sum_{j=1}^N p_{GAP}(i, j) &= \sum_{j=1}^N \frac{1}{|S_i|} \sum_{t \in S_i} \frac{I(j \in J_i(t))}{|J_i(t)|} \\ &= \frac{1}{|S_i|} \sum_{t \in S_i} \frac{1}{|J_i(t)|} \sum_{j=1}^N I(j \in J_i(t)) \\ &= \frac{1}{|S_i|} \sum_{t \in S_i} \frac{1}{|J_i(t)|} |J_i(t)| \\ &= \frac{1}{|S_i|} \sum_{t \in S_i} 1 \\ &= 1. \end{aligned}$$

□

This proposition allows us to directly use RF-GAP as weights for classification or regression. We show that the proximity-weighted prediction under Definition 3 matches the random forest OOB prediction, giving the same OOB error rate in both the classification and regression settings. The OOB error rate is typically used to estimate the forest's generalization error and quantify its goodness of fit, this indicates that RF-GAP accurately represents the geometry learned by the random forest.

Theorem 1 (Proximity-Weighted Regression). *For a given training data set $\mathcal{S} = \{(\mathbf{x}_1, y_1) \dots (\mathbf{x}_N, y_N)\}$, with $y_i \in \mathbb{R}$, the random forest OOB regression prediction is exactly determined by the proximity-weighted sum using RF-GAP (Definition 3).*

Proof: For a given tree, t , and $i \in O(t)$, the decision tree predictor of y_i is the mean response of the in-bag observations in the appropriate terminal node. That is,

$$\hat{y}_i(t) = \frac{1}{|J_i(t)|} \sum_{j \in J_i(t)} y_j.$$

The random forest prediction, \hat{y}_i , is the mean response over all trees for which i is out of bag. That is,

$$\begin{aligned} \hat{y}_i &= \frac{1}{|S_i|} \sum_{t \in S_i} \hat{y}_i(t) \\ &= \frac{1}{|S_i|} \sum_{t \in S_i} \frac{1}{|J_i(t)|} \sum_{j \in J_i(t)} y_j. \end{aligned}$$

The proximity-weighted predictor, \hat{y}_i^p , is simply the weighted sum of responses, $\{y_j\}_{j \neq i}$.

$$\begin{aligned} \hat{y}_i^p &= \sum_{j=1}^N p_{GAP}(i, j) y_j \\ &= \sum_{j=1}^N \left\{ \frac{1}{|S_i|} \sum_{t \in S_i} \frac{I(j \in J_i(t))}{|J_i(t)|} \right\} y_j \\ &= \frac{1}{|S_i|} \sum_{t \in S_i} \frac{1}{|J_i(t)|} \sum_{j=1}^N I(j \in J_i(t)) y_j \\ &= \frac{1}{|S_i|} \sum_{t \in S_i} \frac{1}{|J_i(t)|} \sum_{j \in J_i(t)} y_j \\ &= \hat{y}_i. \end{aligned}$$

□

Theorem 2 (Proximity-Weighted Classification). *For a given training data set $\mathcal{S} = \{(\mathbf{x}_1, y_1) \dots (\mathbf{x}_N, y_N)\}$, with $y_i \in \{1, \dots, K\}$ for all $i \in \{1, \dots, N\}$, the random forest OOB classification prediction is exactly determined by the weighted-majority vote using RF-GAP (Definition 3) as weights.*

Proof: Given the training set $\{(\mathbf{x}_1, y_1) \dots (\mathbf{x}_N, y_N)\}$, with $y_i \in \{1, \dots, K\}$, for a given tree t and observation $i \in O(t)$, the decision tree prediction of the label y_i is determined by the majority vote among in-bag observations within the shared terminal node:

$$\hat{y}_i(t) = \arg \max_{l=1, \dots, K} \sum_{j \in J_i(t)} I(y_j = l).$$

Thus, the random forest classification prediction, \hat{y}_i , is the most popular predicted class over all $t \in S_i$:

$$\begin{aligned} \hat{y}_i &= \arg \max_{k=1, \dots, K} \sum_{t \in S_i} I(\hat{y}_i(t) = k) \\ &= \arg \max_{k=1, \dots, K} \sum_{t \in S_i} I \left(\left\{ \arg \max_{l=1, \dots, K} \sum_{j \in J_i(t)} I(y_j = l) \right\} = k \right) \end{aligned}$$

We show equivalence with the proximity-weighted predictor. The proximity-weighted predictor predicts the class with the largest proximity-weighted vote:

$$\begin{aligned}
\hat{y}_i^p &= \arg \max_{k=1, \dots, K} \sum_{j=1}^N p_{GAP}(i, j) I(y_j = k) \\
&= \arg \max_{k=1, \dots, K} \left\{ \sum_{j=1}^N \left(\frac{1}{|S_i|} \sum_{t \in S_i} \frac{I(j \in J_i(t))}{|J_i(t)|} \right) I(y_j = k) \right\} \\
&= \arg \max_{k=1, \dots, K} \left\{ \frac{1}{|S_i|} \sum_{t \in S_i} \frac{1}{|J_i(t)|} \left(\sum_{j=1}^N I(j \in J_i(t), y_j = k) \right) \right\} \\
&= \arg \max_{k=1, \dots, K} \left\{ \frac{1}{|S_i|} \sum_{t \in S_i} \frac{1}{|J_i(t)|} \sum_{j \in J_i(t)} I(y_j = k) \right\} \\
&= \arg \max_{k=1, \dots, K} \left\{ \sum_{t \in S_i} \frac{1}{|J_i(t)|} \sum_{j \in J_i(t)} I(y_j = k) \right\}.
\end{aligned}$$

The last line holds as $|S_i|$ does not depend on k . As classification trees in a random forest are grown until terminal (leaf) nodes are pure, all in-bag observations belong to the same class. Denote the common class for any observation $j \in J_i(t)$ as $y_{i,t}$. Then the single tree predictor is given by

$$\begin{aligned}
\hat{y}_i(t) &= \arg \max_{l=1, \dots, K} \sum_{j \in J_i(t)} I(y_j = l) \\
&= y_{i,t}.
\end{aligned}$$

and the random forest predictor is

$$\hat{y}_i = \arg \max_{k=1, \dots, K} \sum_{t \in S_i} I(y_{i,t} = k).$$

The proximity-weighted predictor is thus

$$\begin{aligned}
\hat{y}_i^p &= \arg \max_{k=1, \dots, K} \left\{ \sum_{t \in S_i} \frac{1}{|J_i(t)|} \sum_{j \in J_i(t)} I(y_j = k) \right\} \\
&= \arg \max_{k=1, \dots, K} \left\{ \sum_{t \in S_i} \frac{1}{|J_i(t)|} \sum_{j \in J_i(t)} I(y_{i,t} = k) \right\} \\
&= \arg \max_{k=1, \dots, K} \left\{ \sum_{t \in S_i} \frac{1}{|J_i(t)|} |J_i(t)| I(y_{i,t} = k) \right\} \\
&= \arg \max_{k=1, \dots, K} \sum_{t \in S_i} I(y_{i,t} = k) \\
&= \hat{y}_i.
\end{aligned}$$

□
predictor using the original random forest proximity definition (Definition 1), the OOB adaptation (Definition 2), PBK [57], and RFProxiH [59].

We compared proximity-prediction results on 19 datasets from the UCI repository [61]. Each dataset was randomly partitioned into training (80%) and test (20%) sets. For each dataset, the same trained random forest was used to produce all compared proximities. Table 1 gives the absolute difference between the proximity-weighted training errors and the random forest OOB error rate. The proximity-weighted predictor using RF-GAP almost exactly matches the random forest OOB error rates; discrepancies are due to random tie-breaking. In contrast, the original definition, PBK, and RFProxiH typically produce much lower training error rates, suggesting overfitting. The OOB proximities (Definition 2) produce training error rates which are sometimes lower and sometimes higher than the random forest OOB error rate. Table 1 also provides the difference between the test error rates for the same datasets. Here, we still see that the proximity predictions using RF-GAP nearly always match those of the random forest, while this is not the case for the other proximity constructions.

The RF-GAP proximities also generally produce the lowest test errors. This can be seen in Fig. 2, which plots the training versus test error rates using the different proximity measures. Table 2 gives the regression slope for each proximity definition. From here it is clear that the original proximities, PBK, and RFProxiH overfit the training data on average. This is corroborated in Fig. 3, which plots the difference between the random forest out-of-bag error rates and the proximity-weighted errors across the same datasets, demonstrating that the RF-GAP predictions nearly perfectly match those of the random forest for both training and tests sets. It is clear that the original definition, PBK, and RFProxiH overfit the training data in contrast.

6 COMPARISON OF COMMON PROXIMITY-BASED APPLICATIONS

In this section we demonstrate that using RF-GAP in multiple applications leads to improved performance relative to the other random forest proximity measures. We perform comparisons with applying MDS to the proximities for visualization in Section 6.1, data imputation in Section 6.2, and outlier detection in Section 6.3. Random forest proximities have already been established as successful in these applications (see Section 2). Thus, we focus our comparisons in these applications on existing random forest proximity definitions to show that the improved representation of the random forest geometry in RF-GAP leads to improved performance. Additional results beyond those presented here are given in Appendix A.

5 EXPERIMENTAL VALIDATION OF PROXIMITY-WEIGHTED PREDICTION

To demonstrate that the RF-GAP proximities preserve the random-forest learned data geometry, we empirically validate Theorems 1 and 2 from Section 4, demonstrating that the random forest predictions are preserved in the proximity construction. We also compare to the proximity-weighted

6.1 Visualization using Multi-Dimensional Scaling

Leo Breiman [1], [16] first used multi-dimensional scaling (MDS) on random forest proximities to visualize the data. Since then, MDS with random forest proximities has been used in many applications including tumor profiling [29], visualizing biomarkers [40], pathway analysis [30], multi-view learning [28], [49], and unsupervised learning [39].

TABLE 1

Comparison of proximity-weighted predictions to the random forest errors. The random forest OOB error (on the training set) and test errors are given in the columns under RF. The absolute difference of the training and test errors with, respectively, the OOB error and RF test error are given for each proximity-weighted prediction. The results nearest the random forest predictions are bold. RF-GAP nearly perfectly matches the random forest results, with differences being accounted for by randomly broken ties in the forest. The other definitions tend to overfit the training data, as can be seen with the large differences between the OOB and test error rates. This is corroborated by Fig. 3, which plots the differences. Note that RFProxIH (based on RFDistH from [59]) is not written for data with a continuous response. Additionally, since it is generated using Euclidean distance, it is not compatible with datasets with categorical variables or missing values.

Type	RF		RF-GAP		Original		OOB		PBK		RFProxIH	
Data	OOB	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
Arrhythmia	0.258	0.297	0	0	0.042	0.077	0.067	0.088	0.031	0.077	NA	NA
Balance Scale	0.156	0.144	0.004	0	0.07	0.056	0.05	0.056	0.068	0.056	0.068	0.056
Banknote	0.009	0.011	0	0	0.001	0.011	0.009	0.011	0.011	0.018	0.001	0.011
Breast Cancer	0.03	0.043	0	0	0.007	0.014	0.02	0.014	0.011	0.014	0.007	0.014
Car	0.038	0.052	0.007	0.023	0	0.003	0.017	0.006	0.048	0.043	NA	NA
Diabetes	0.243	0.182	0.002	0	0.148	0.006	0.028	0.013	0.124	0	0.147	0
Ecoli	0.153	0.088	0	0	0.052	0	0.007	0.015	0.041	0.029	0.049	0
Glass	0.211	0.186	0	0	0.135	0.023	0.029	0.023	0.111	0.047	0.123	0.047
Heart Disease	0.417	0.475	0	0	0.302	0.115	0.194	0.115	0.252	0.18	NA	NA
Hill Valley	0.436	0.41	0.002	0	0.347	0.082	0.11	0.131	0.304	0.098	0.331	0.082
Ionosphere	0.075	0.042	0	0	0.025	0	0.004	0	0.021	0	0.025	0
Iris	0.05	0.067	0	0	0.033	0	0.008	0	0.033	0	0.033	0
Liver	0.305	0.336	0	0	0.186	0.078	0.071	0.078	0.162	0.052	NA	NA
Lymphography	0.153	0.2	0.008	0	0.093	0.033	0.008	0	0.068	0.033	0.085	0.033
Parkinsons	0.09	0.103	0	0	0.013	0	0.051	0	0.006	0	0.013	0
Seeds	0.063	0.075	0.006	0	0.038	0.025	0.019	0.05	0.019	0.05	0.038	0.025
Sonar	0.169	0.167	0	0	0.145	0.024	0.066	0.024	0.12	0.024	0.145	0.024
Statlog	0.234	0.28	0.001	0	0.139	0.005	0.044	0.015	0.116	0.005	NA	NA
Tic-Tac-Toe	0.048	0.042	0.003	0	0.031	0.026	0.038	0.021	0.022	0.057	NA	NA
Train - Test	-	-	0.003 ± 0.008		0.092 ± 0.019		0.004 ± 0.012		0.076 ± 0.016		0.071 ± 0.019	

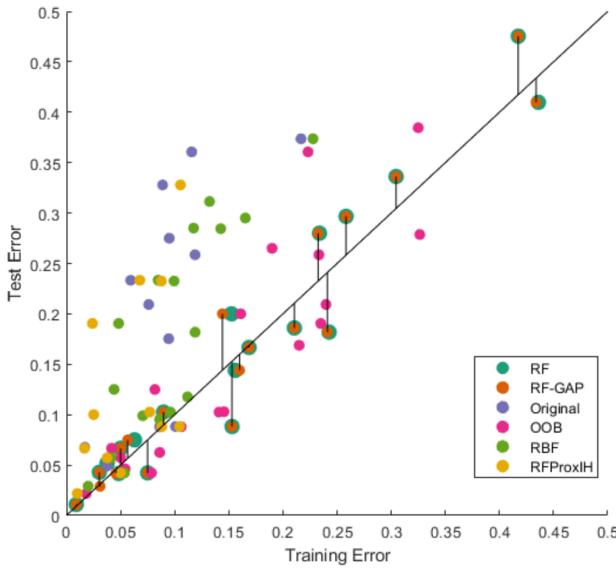


Fig. 2. Training vs. test error of the proximity-weighted predictions across multiple datasets. We see that the original proximities, PBK, and RFProxIH, tend to overfit the training data, as demonstrated by points above the line $y = x$. The random forest errors and RF-GAP nearly perfectly align in most cases and are each well-described by the line. OOB also follows the identity line well, but does not match the RF predictions.

Here, we compare visualizations using the five different proximities. In each case, metric MDS was applied to $\sqrt{1 - \text{prox}}$ to produce two-dimensional visualizations.

Figure 4 gives an example of MDS applied to the clas-

TABLE 2

The regression slopes of each proximity type corresponding to the points in Fig. 2. RF-GAP and OOB do not exhibit bias towards the training data as they have a slope close to one, while the larger slope of the other proximity definitions indicate they are overfitting the data.

Type	Slope
RF-GAP	1.036
Original	1.700
OOB	1.044
PBK	1.664
RFProxIH	1.3955

sic Sonar dataset from UCI [61] with an OOB error rate was 15.85%. RF-GAP proximities (Fig. 4 (a)) show two class-groupings with misclassified observations between the groups or within the opposing class. The original proximities, PBK, and RFProxIH (Fig. 4 (b, d, and e, respectively)) show a fairly clear separation between the two classes. For these proximities, they appear nearly linearly separable which does not accurately reflect the data nor the geometry learned by the random forest given an error rate of nearly 16%. Definition 2 (Fig. 4 (c)) has a similar effect as the RF-GAP definition, but with a less clear boundary and seemingly misplaced observations that are deep within the wrong class. These results suggest that RF-GAP can lead to improved supervised visualization and dimensionality reduction techniques. See Appendix A for further experiments.

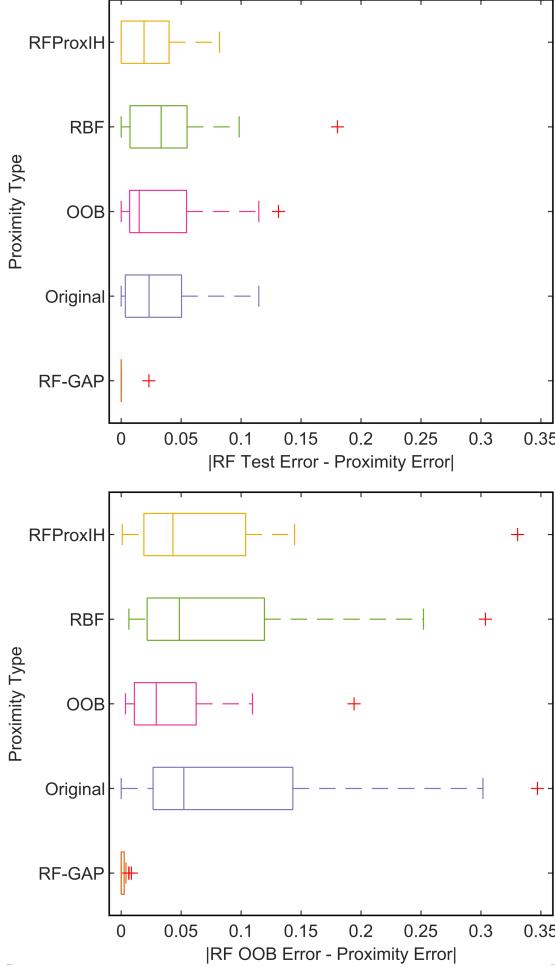


Fig. 3. These boxplots show the absolute difference between the proximity-weighted prediction training and test errors with the random forest OOB error rate and test error, respectively, across five proximity definitions. RF-GAP proximity predictions most nearly match the random forest predictions for both the training (left) and test (right) data, thus, best preserving the geometry learned by the random forest. Various UCI datasets were randomly split into training and test datasets for this (80% training, 20% test).

6.2 Imputation

In [35], experimental results showed that, out of nine compared imputation methods across seven different missing mechanisms (missing at random, missing completely at random, etc.), random forest-based imputation was generally the most accurate. This has been corroborated by [36], [37], [38]. Here we describe the algorithm for random forest imputation and compare results using various proximity definitions.

To impute missing values of variable j :

- (1) If variable j is continuous, initialize the imputation with the median of the in-class values of variable j ; otherwise, initialize it with the most frequent in-class value. In the regression context, the median or most frequent values are computed across all observations without missing values in variable j .
- (2) Train a random forest using the imputed dataset and
- (3) construct the proximity matrix from the forest.
- (4) If variable j is continuous, replace the missing values with the proximity-weighted sum of the non-missing

TABLE 3

The average ranks of the imputation scores across the various UCI datasets and five percentages of missing values. Each imputation experiment was repeated 100 times with different random initialization. For each percentage under 75%, RF-GAP always produced the best results. See Table A.1 for full imputation results.

	5%	10%	25%	50%	75%
RF-GAP	1.00	1.00	1.00	1.00	1.31
OOB	2.62	2.62	2.62	2.56	2.38
Original	2.69	2.88	2.62	2.69	2.44
RFProxIH	3.69	3.50	3.75	3.75	3.88

values. If categorical, replace the missing values with the proximity-weighted majority vote.

- (5) Repeat steps 2 - 4 as required. In many cases, a single iteration is sufficient.

We show empirically that random forest imputation is generally improved using the RF-GAP proximity definition. For our experiment, we selected various datasets from the UCI repository [61] and for each dataset, we removed 5%, 10%, 25%, 50%, and 75% of values at random (that is, the values are missing completely at random, or MCAR), using the missMethods R package. Two comparisons were made: 1) we computed the mean MSE across 100 repetitions using a single iteration, and 2) we computed the mean (across 10 repetitions) MSE at each of 15 iterations.

A summary of performance rankings is given in Table 3. Across all compared proximity definitions (RF-GAP, OOB, Original, and RFProxIH), RF-GAP achieved the best imputation scores at all percentages less than 75%, and outperformed across 69% of the datasets when the percentage of missing values was 75%. For full results, see Table A.1 in the supplementary materials which gives the mean MSE across 100 repetitions using a single iteration of the above-described algorithm. The number of observations and variables for each dataset are provided in the table.

Supplementary figures (Fig. A.7, A.8, and A.9) compare imputation results across 16 datasets, using 15 iterations in each experiment with the mean MSE and standard errors recorded for each of the repetitions. The value recorded at iteration 0 is the MSE given the median- or majority-imputed datasets. In many cases, the imputation appears to converge quickly with relatively few iterations. Generally, the RF-GAP proximities outperformed the other definitions at each number of iterations. For high percentages of missing values (at least 75%), or for small datasets, the random forest imputation does not always converge and performance may actually decrease as the number of iterations increases. These results suggest that RF-GAP can be used to improve random forest imputation.

6.3 Outlier Detection

Random forests can be used to detect outliers in the data. In the classification setting, outliers may be described as observations with measurements which significantly differ from those of other observations within the same class. In some cases, these outliers may be similar to observations in a different class, or perhaps they may distinguish themselves from observations in all known classes. In either case, outlying observations may negatively impact the training

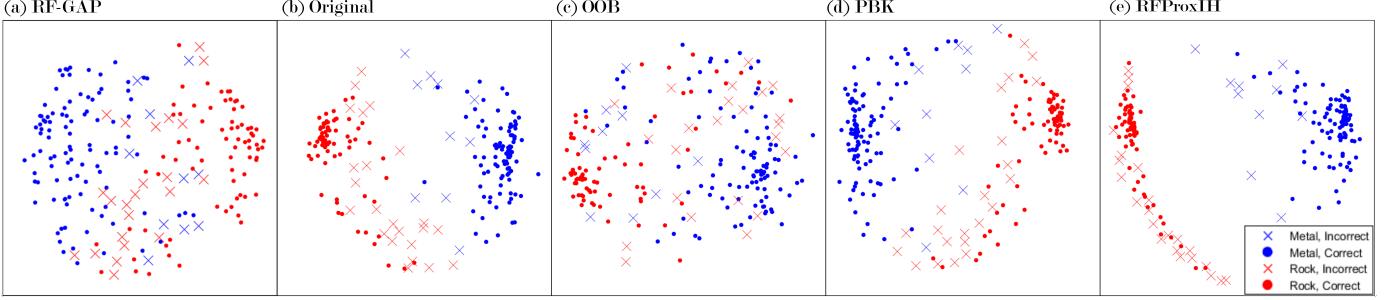


Fig. 4. Comparison of MDS embeddings using different RF proximity definitions. Proximities were constructed from a random forest trained on the two-class Sonar dataset (208 observations of 60 variables) from the UCI repository which gave an OOB error rate of 15.87%. Multi-dimensional scaling (MDS) was applied to $\sqrt{1 - \text{prox}}$ using (a) RF-GAP proximities, (b) the original proximities, (c) OOB proximities (Definition 2), (d) PBK [57], and RFProxIH [59]. Using RF-GAP proximities, the visualization depicts a good representation of the forest's classification problem. For correctly-classified points (dots), there are two clear groupings, while misclassified points (squares) are generally located between the groupings or found within the opposite class' cluster, albeit closer to the decision boundary than not. The original definition, PBK, and RFProxIH over-exaggerate the separation between classes. This is apparent in examples (b), (d), and (e) as the two classes appear nearly linearly-separable which does not accurately depict the random forest's performance on the dataset. Using only OOB samples to generate the proximities improves upon those three but seems to add some noise to the visualization. There are still two major class clusters, but some correctly classified points are found farther inside the opposite class' cluster compared to the RF-GAP visualization.

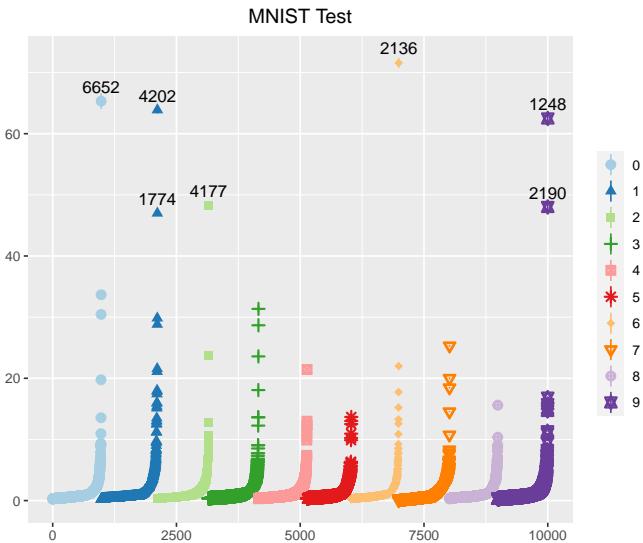


Fig. 5. A sorted plot of the outlier measures for the MNIST dataset as provided by RF-GAP. The vertical axis is the outlier measure as described in Section 6.2. The top seven outlying images are labeled with an index and shown in Fig. 6.

of many classification and regression algorithms, although random forests themselves are rather robust to outliers in the feature variables [2].

Random forest proximity measures can be used to detect within-class outliers as outliers are likely to have small proximity measures with other observations within the same class. Thus, small average proximity values within-class may be used as an outlier measure. We describe the algorithm as follows:

- (1) For each observation, i , compute the raw outlier measure score as $\sum_{j \in \text{class}(i)} \frac{n}{\text{prox}^2(i,j)}$.
- (2) Within each class, determine the median and mean absolute deviation of the raw scores.
- (3) Subtract the median from the raw score and divide by the mean absolute deviation.

The outlier detection measure may be used in conjunc-

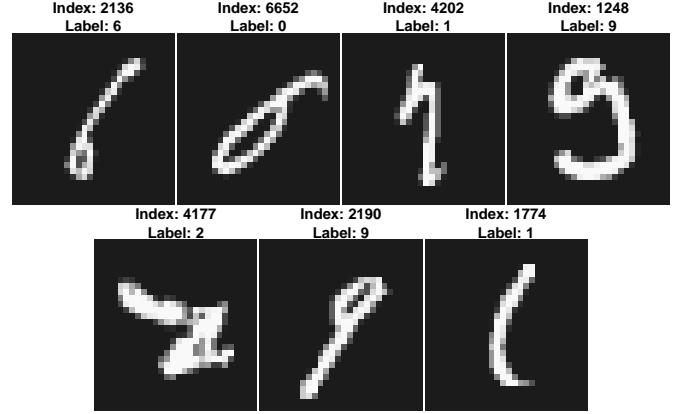


Fig. 6. The top seven outlying digits per RF-GAP (see Fig. 5). Some of these digits may even be difficult for a human to classify, corroborating the RF-GAP outlier score.

tion with MDS for visualization. See Fig. 7 for an example using the Gene Expression Cancer dataset from UCI which has 5 classes across 801 observations and 20531 variables. Here, the point sizes of the scatter plot are proportionally scaled to the outlier measure. From the figure it is clear that points outside of their respective class clusters have higher outlier measures. That the outlier measure is inversely proportional to the average proximity to within-class observations is clear in the case of RF-GAP proximities (Fig. 7 (a)) and is not very clear in the cases of the original definition (b) and RFProxIH (e). This suggests that RF-GAP can be used to improve random forest outlier detection. See Appendix A for further experiments.

7 CONCLUSION

In this paper, we presented a new definition of random forest proximities called RF-GAP that characterizes the random forest out-of-bag prediction results using a weighted nearest neighbor predictor. We proved that the performance of the proximity-weighted predictor exactly matches the out-of-bag prediction results of the trained random forest

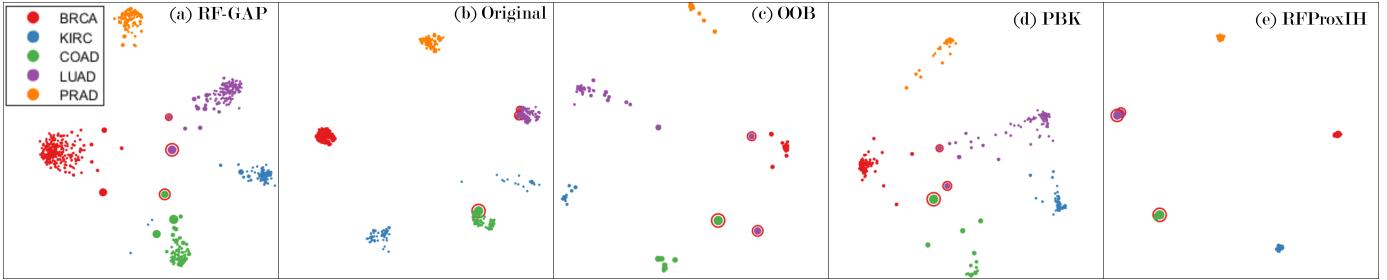


Fig. 7. MDS applied to the random forest proximities computed from the Gene Expression Cancer dataset from UCI [61]. The point sizes are inversely proportional to the average proximity of a given observation to all other within-class observations. Misclassified observations are designated by an outline of the color of the misclassified label. The misclassifications in (a) (RF-GAP), (c) (OOB), and (d) (PBK) are clear based on the distance from the blue cluster. The original proximities, (b), and RFProxIH (e) do not clearly account for the misclassified points. The outlier measure scaling in (a) gives a clear reflection of the distance of points to their respective clusters.

and also demonstrated this relationship empirically. Thus, RF-GAP proximities capture the random forest-learned data geometry which provides empirical improvements over other existing definitions in applications such as proximity-weighted prediction, missing data imputation, outlier detection, and visualization.

Additional random forest proximity applications can be explored in future works, including quantifying outlier detection performance and comparing against other, non-tree based methods, assessing variable importance, and applying RF-GAP to multi-view learning. This last application shows much promise as classification accuracy was greatly increased after combining proximities in [28], [49] using other definitions. An interesting visualization application was introduced in [31] which may see improvements using geometry-preserving proximities. Multi-view learning may also be paired with this approach in some domains to visualize and assess contributions from the various modes or to perform manifold alignment. An additional area of improvement is scalability. Our approach is useful for datasets with a few thousand observations, but we can expand its capabilities by implementing a sparse version of RF-GAP. Further adaptations may make random forest proximity applications accessible for even larger datasets.

REFERENCES

- [1] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, Oct. 2001.
- [2] A. Cutler, D. R. Cutler, and J. R. Stevens, *Random Forests*. Boston, MA: Springer US, 2012, pp. 157–175.
- [3] L. Benali, G. Notton, A. Fouillot, C. Voyant, and R. Dizene, "Solar radiation forecasting using artificial neural network and random forest methods: Application to normal beam, horizontal diffuse and global components," *Renewable Energy*, vol. 132, pp. 871–884, 2019.
- [4] T. Han, D. Jiang, Q. Zhao, L. Wang, and K. Yin, "Comparison of random forest, artificial neural networks and support vector machine for intelligent diagnosis of rotating machinery," *Trans. Inst. Meas. Control*, vol. 40, pp. 2681 – 2693, 2018.
- [5] C. Iwendi *et al.*, "Covid-19 patient health prediction using boosted random forest algorithm," *Front. Public Health*, vol. 8, p. 357, 2020.
- [6] C. M. Yeşilkanat, "Spatio-temporal estimation of the daily cases of covid-19 in worldwide using random forest machine learning algorithm," *Chaos, Solitons Fractals*, vol. 140, p. 110210, 2020.
- [7] V. Nhu *et al.*, "Shallow landslide susceptibility mapping by random forest base classifier and its ensembles in a semi-arid region of iran," *Forests*, vol. 11, no. 4, 2020.
- [8] X. Fang *et al.*, "Forecasting incidence of infectious diarrhea using random forest in jiangsu province, china," *BMC Infect. Dis.*, vol. 20, no. 1, p. 222, Mar 2020.
- [9] L. Yang *et al.*, "Study of cardiovascular disease prediction model based on random forest in eastern china," *Sci. Rep.*, vol. 10, no. 1, p. 5245, Mar 2020.
- [10] S. Saha, M. Saha, K. Mukherjee, A. Arabameri, P. T. T. Ngo, and G. C. Paul, "Predicting the deforestation probability using the binary logistic regression, random forest, ensemble rotational forest, reptree: A case study at the gumani river basin, india," *Sci. Total Environ.*, vol. 730, p. 139197, 2020.
- [11] M. Gholizadeh, M. Jamei, I. Ahmadianfar, and R. Pourrajab, "Prediction of nanofluids viscosity using random forest (rf) approach," *Chemom. Intell. Lab. Syst.*, vol. 201, p. 104010, 2020.
- [12] C. Rao, M. Liu, M. Goh, and J. Wen, "2-stage modified random forest model for credit risk assessment of p2p network lending to "three rurals" borrowers," *Appl. Soft Comput.*, vol. 95, p. 106570, 2020.
- [13] Z. Lv, J. Zhang, H. Ding, and Q. Zou, "Rf-pseu: A random forest predictor for rna pseudouridine sites," *Front. Bioeng. Biotechnol.*, vol. 8, pp. 134–134, Feb 2020.
- [14] S. Badar ud din Tahir, A. Jalal, and M. Batool, "Wearable sensors for activity analysis using smo-based random forest over smart home and sports datasets," in *2020 3rd Int. Conf. Adv. Comput. Sci. (ICACS)*, 2020, pp. 1–6.
- [15] K. Tan, H. Wang, L. Chen, Q. Du, P. Du, and C. Pan, "Estimation of the spatial distribution of heavy metal in agricultural soils using airborne hyperspectral imaging and random forest," *J. Hazard. Mater.*, vol. 382, p. 120987, 2020.
- [16] L. Breiman and A. Cutler, "Random forests," https://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm#prox, (Accessed on 04/15/2020).
- [17] J. Kruskal and M. Wish, *Multidimensional Scaling*. Sage Publications, 1978.
- [18] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, 2008.
- [19] K. R. Moon *et al.*, "Visualizing structure and transitions in high-dimensional biological data," *Nat. Biotechnol.*, vol. 37, no. 12, pp. 1482–1492, Dec 2019.
- [20] A. F. Duque, G. Wolf, and K. R. Moon, "Visualizing high dimensional dynamical processes," in *MLSP*, 2019.
- [21] A. F. Duque, S. Morin, G. Wolf, and K. Moon, "Extendable and invertible manifold learning with geometry regularized autoencoders," in *2020 IEEE Int. Conf. Big Data (Big Data)*, 2020, pp. 5027–5036.
- [22] J. B. Tenenbaum, V. Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.
- [23] L. McInnes, J. Healy, and J. Melville, "UMAP: Uniform manifold approximation and projection for dimension reduction," *arXiv*, vol. abs/1802.03426, 2018.
- [24] A. Ng, M. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," *Adv. Neural. Inf. Process. Syst.*, vol. 14, pp. 849–856, 2001.
- [25] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, Sep 1995.
- [26] B. Scholkopf, A. Smola, and K. Müller, "Kernel principal component analysis," in *Advances in Kernel Methods - Support Vector Learning*. MIT Press, 1999, pp. 327–352.

- [27] M. B. Pouyan, J. Birjandtalab, and M. Nourani, "Distance metric learning using random forest for cytometry data," in *2016 38th Annu. Int. Conf. of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2016, pp. 2590–2590.
- [28] K. R. Gray, P. Aljabar, R. A. Heckemann, A. Hammers, and D. Rueckert, "Random forest-based manifold learning for classification of imaging data in dementia," in *Machine Learning in Medical Imaging*, K. Suzuki, F. Wang, D. Shen, and P. Yan, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 159–166.
- [29] T. Shi, D. Seligson, A. S. Belldegrun, A. Palotie, and S. Horvath, "Tumor classification by tissue microarray profiling: random forest clustering applied to renal cell carcinoma," *Mod. Pathol.*, vol. 18, no. 4, pp. 547–557, Apr 2005.
- [30] H. Pang *et al.*, "Pathway analysis using random forests classification and regression," *Bioinformatics*, vol. 22, no. 16, pp. 2028–2036, 06 2006.
- [31] J. S. Rhodes, A. Cutler, G. Wolf, and K. R. Moon, "Random forest-based diffusion information geometry for supervised visualization and data exploration," in *2021 IEEE Stat. Signal Process. Workshop (SSP)*, 2021, pp. 331–335.
- [32] J. Zhang, M. Zulkernine, and A. Haque, "Random-forests-based network intrusion detection systems," *IEEE Trans. Syst. Man Cybern., Part C (Applications and Reviews)*, vol. 38, no. 5, pp. 649–659, 2008.
- [33] P. O. Gislason, J. A. Benediktsson, and J. R. Sveinsson, "Random forests for land cover classification," *Pattern Recognit. Lett.*, vol. 27, no. 4, pp. 294 – 300, 2006, pattern Recognition in Remote Sensing (PRRS 2004).
- [34] V. L. Narayana, A. P. Gopi, S. R. Khadherbhi, and V. Pavani, "Accurate identification and detection of outliers in networks using group random forest methodology," *J. Crit. Rev.*, vol. 7, no. 6, pp. 381–384, 2020.
- [35] M. Kokla, J. Virtanen, M. Kollehmainen, J. Paananen, and K. Hahnheva, "Random forest-based imputation outperforms other methods for imputing lc-ms metabolomics data: a comparative study," *BMC Bioinf.*, vol. 20, no. 1, p. 492, Oct 2019.
- [36] B. Ramosaj and M. Pauly, "Predicting missing values: a comparative study on non-parametric approaches for imputation," *Comput. Stat.*, vol. 34, no. 4, pp. 1741–1764, Dec 2019.
- [37] A. Pantanowitz and T. Marwala, "Missing data imputation through the use of the random forest algorithm," in *Advances in Computational Intelligence*, W. Yu and E. N. Sanchez, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 53–62.
- [38] A. D. Shah, J. W. Bartlett, J. Carpenter, O. Nicholas, and H. Hemingway, "Comparison of random forest and parametric imputation models for imputing missing data using mice: a caliber study," *Am. J. Epidemiol.*, vol. 179, no. 6, pp. 764–774, Mar 2014.
- [39] T. Shi and S. Horvath, "Unsupervised learning with random forest predictors," *J. Comput. Graphical Stat.*, vol. 15, no. 1, pp. 118–138, 2006.
- [40] E. J. Finehout, Z. Franck, L. H. Choe, N. Relkin, and K. H. Lee, "Cerebrospinal fluid proteomic biomarkers for Alzheimer's disease," *Ann. Neurol.*, vol. 61, no. 2, pp. 120–129, Feb. 2007.
- [41] N. Nesa, T. Ghosh, and I. Banerjee, "Outlier detection in sensed data using statistical learning models for iot," in *IEEE Wirel. Commun. Netw. (WCNC)*, 2018, pp. 1–6.
- [42] C. Liu, M. White, and G. Newell, "Detecting outliers in species distribution data," *J. Biogeogr.*, vol. 45, no. 1, pp. 164–176, 2018.
- [43] F. B. de Santana, W. Borges Neto, and R. J. Poppi, "Random forest as one-class classifier and infrared spectroscopy for food adulteration detection," *Food Chem.*, vol. 293, pp. 323–332, 2019.
- [44] D. Baron and D. Poznanski, "The weirdest SDSS galaxies: results from an outlier detection algorithm," *Mon. Not. R. Astron. Soc.*, vol. 465, no. 4, pp. 4530–4555, 11 2016.
- [45] V. Narayana, A. Gopi, S. Khadherbhi, and V. Pavani, "Accurate identification and detection of outliers in networks using group random forest methodology," *J. Crit. Rev.*, vol. 7, no. 6, pp. 381–384, 2020.
- [46] S. van Buuren and K. Groothuis-Oudshoorn, "mice: Multivariate imputation by chained equations in r," *J. Stat. Softw.*, vol. 45, no. 3, pp. 1–67, 2011.
- [47] Q. Zhou, W. Hong, L. Luo, and F. Yang, "Gene selection using random forest and proximity differences criterion on dna microarray data," *J. Convergence Inf. Technol.*, vol. 5, pp. 161–170, 2010.
- [48] L. S. Whitmore, A. George, and C. M. Hudson, "Explicating feature contribution using random forest proximity distances," 2018.
- [49] H. Cao, S. Bernard, R. Sabourin, and L. Heutte, "Random forest dissimilarity based multi-view learning for radiomics application," *Pattern Recognit.*, vol. 88, pp. 185 – 197, 2019.
- [50] J. A. Seoane, I. N. M. Day, J. P. Casas, C. Campbell, and T. R. Gaunt, "A random forest proximity matrix as a new measure for gene annotation," in *22th European Symposium on Artificial Neural Networks, ESANN 2014, Bruges, Belgium, April 23-25, 2014*, 2014.
- [51] P. Zhao, X. Su, T. Ge, and J. Fan, "Propensity score and proximity matching using random forest," *Contemp. Clin. Trials*, vol. 47, pp. 85–92, Mar 2016.
- [52] T. Hastie, R. Tibshirani, and J. H. Friedman, *The elements of statistical learning: Data mining, inference, and prediction*. Springer Open, 2017.
- [53] A. Liaw and M. Wiener, "Classification and regression by random forest," *R News*, vol. 2, no. 3, pp. 18–22, 2002.
- [54] M. N. Wright and A. Ziegler, "ranger: A fast implementation of random forests for high dimensional data in C++ and R," *J. Stat. Softw.*, vol. 77, no. 1, pp. 1–17, 2017.
- [55] T. Hastie, R. Tibshirani, and J. Friedman, *Random Forests*. New York, NY: Springer New York, 2009, pp. 587–604.
- [56] R Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2019.
- [57] C. Englund and A. Verikas, "A novel approach to estimate proximity in a random forest: An exploratory study," *Expert Syst. Appl.*, vol. 39, no. 17, p. 13046–13050, Dec. 2012.
- [58] A. Davies and Z. Ghahramani, "The random forest kernel and other kernels for big data from random partitions," 2014.
- [59] H. Cao, S. Bernard, R. Sabourin, and L. Heutte, "A novel random forest dissimilarity measure for multi-view learning," *ArXiv*, vol. abs/2007.02572, 2020.
- [60] Y. Lin and Y. Jeon, "Random forests and adaptive nearest neighbors," *J. Am. Stat. Assoc.*, vol. 101, no. 474, pp. 578–590, 2006.
- [61] D. Dua and C. Graff, "UCI machine learning repository," 2017.

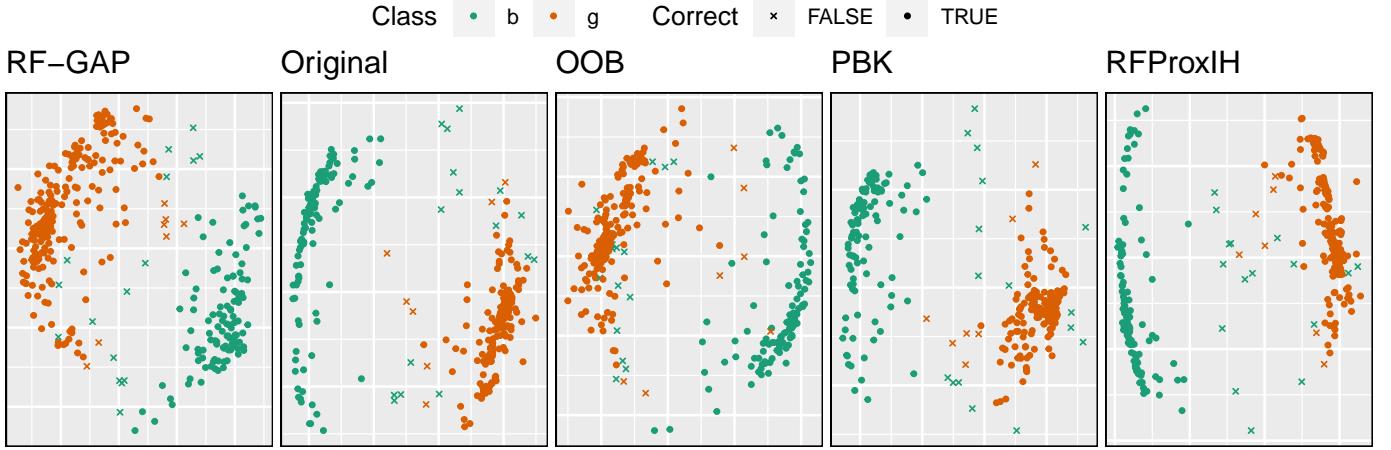


Fig. A.1. MDS applied to various random forest proximities on the Ionosphere dataset [61]. This binary classification problem predicts whether or not returned radar signals are representative of a structure (good) or not (bad). We see a similar pattern here regarding the MDS embeddings as in Figure 4. The class separations are somewhat exaggerated for the Original, PBK, and RFProxIH proximities, while points clearly susceptible to misclassification are identifiable in the RF-GAP and OOB plots.

APPENDIX A ADDITIONAL EXPERIMENTAL RESULTS

Here we present additional experimental results, demonstrating that using RF-GAP leads to improved imputation, visualization, and outlier detection over the other random forest proximity measures.

A.1 Multidimensional Scaling and Outlier Detection

Here we provide additional examples of MDS applied to the various random forest proximities. In Figure A.1, we compare the plotted MDS embeddings on the Ionosphere data from the UCI [61]. It is clear from the images that the random forest's misclassified points are typically found on the border between the two class clusters in the RF-GAP embeddings while this is not always the case for the other proximity measures. Additional figures (A.2, A.3, and A.4) are given to display MDS applied to the proximities. In Figure A.2 we see similar patterns which were displayed in Figure 4; exaggerated separation in the Original and RFProxIH and excess noise in OOB. RF-GAP seems to accurately portray why the misclassifications are made in the context of proximity-weighted predictions.

Figures A.4, A.5, and A.6 give additional examples of proximity-based outlier scores. Points that are farther from their respective class clusters can be viewed as outliers and are often misclassified. The point size is proportional to the outlier measure in the figure. This, however, may not be as clear when two or three points are far from their respective cluster but near to each other.

A.2 Data Imputation

Here we show extended results on data imputation using random forest proximities. Table A.1 shows the average imputation results for 100 trials across 16 datasets from the UCI repository [61] using four of the proximity measures with a single iteration. For each dataset, values were removed completely at random in amounts of 5%, 10%, 25%, 50% and 75%. The PBK proximities were omitted from this study due to their slow computational complexity. Additionally, some datasets were not compatible with RFProxIH due to continuous responses or categorical features.

The mean squared error (MSE) is almost universally lower when using RF-GAP for imputation. RF-GAP is only outperformed sometimes when the amount of missing values reaches 75%. Even in these cases, RF-GAP is always in second place. The Banknote, Ionosphere, Optical Digits, Parkinsons, and Waveform datasets particularly show good examples of RF-GAP for imputation. Here, RF-GAP outperforms each of the other definitions and the error decreases monotonically.

Figures A.7, A.8, and A.9 show imputation results across multiple iterations. Each experiment was repeated 100 times across 15 iterations. In general, RF-GAP outperforms the other proximity-weighted imputations although the imputation tends to be much noisier for smaller datasets (see Balance Scale, Ecoli, Iris, and Seeds, for example) and less reliable for large percentages of missing values. This is particularly prominent when 75% of the data is missing. In some of these cases, the error increases with the number of iterations, for example, in the Ecoli and Diabetes data sets.

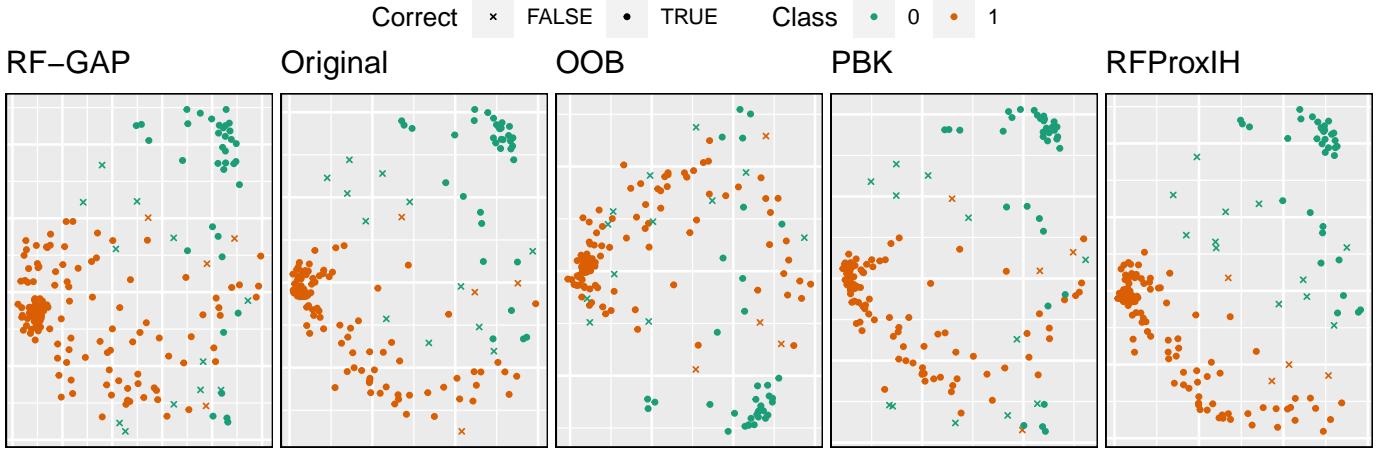


Fig. A.2. MDS applied to various random forest proximities on the Parkinson's dataset (UCI), which tests whether machine learning algorithms can discriminate between healthy and unhealthy speech signals recorded from people with Parkinson's disease. From the RF-GAP embeddings, it is clear that misclassified points are on the borders or edges of the main clusters. This provides an example where random forest predictions correspond to proximity-weighted predictions. This is not always clear in the other embeddings. For example, the Original MDS embeddings shows a misclassified 1 (in the bottom right of the figure) which is nearest observations of the same class. Again, RFProxIH shows nearly perfectly linear separation between classes, which is unreasonable with a random forest error rate of 8.2%.

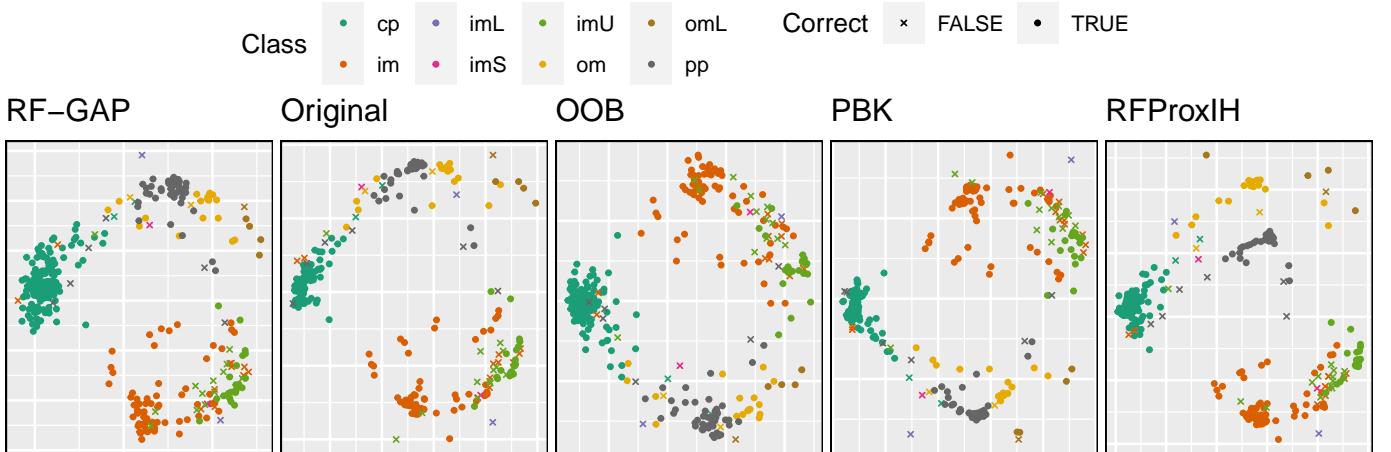


Fig. A.3. The ecoli dataset with eight classes after applying MDS to the random forest proximities. RF-GAP and OOB show looser clusters compared with the others. This is suggestive of less overfitting of the training data.

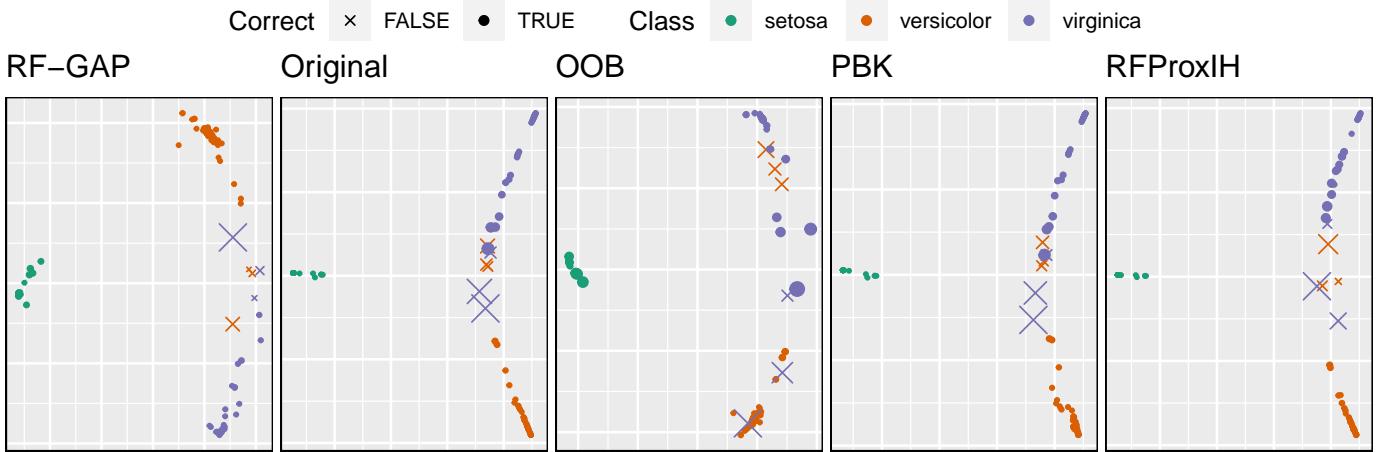


Fig. A.4. Fisher's Iris dataset [61] with MDS applied to the random forest proximities. Here, point size is proportional to the outlier score provided by each method. In each case, observations with high outlier scores corresponded to misclassified points.

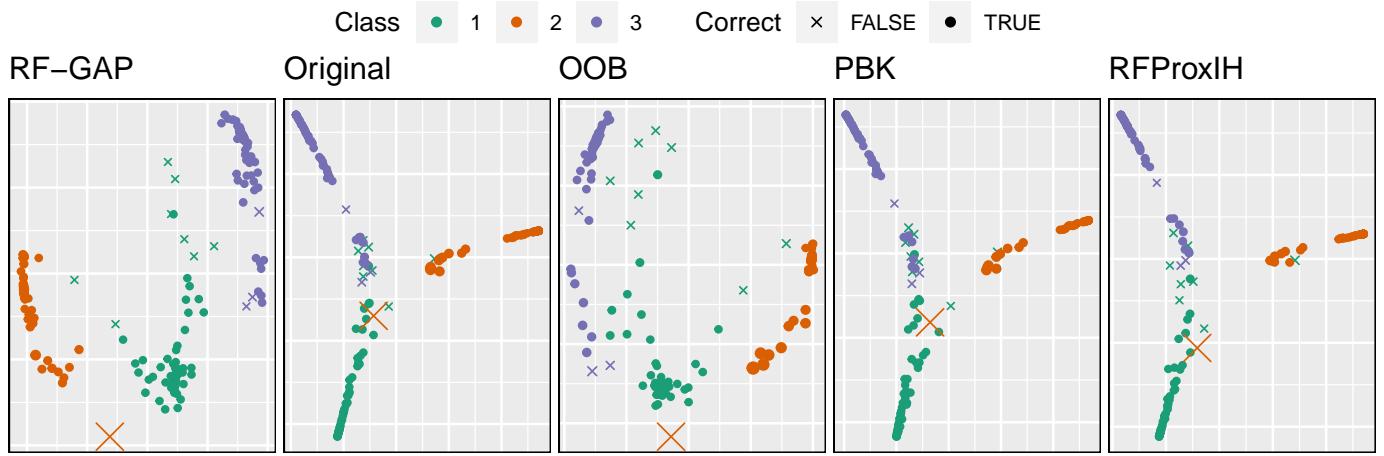


Fig. A.5. The seeds dataset compares three varieties of wheat seeds using geometric properties (e.g. width, length) as features. The OOB and RF-GAP proximities produce more cluster-like structures, vs. the branching seen by the other definitions. RF-GAP clearly shows why the misclassifications are taking place.

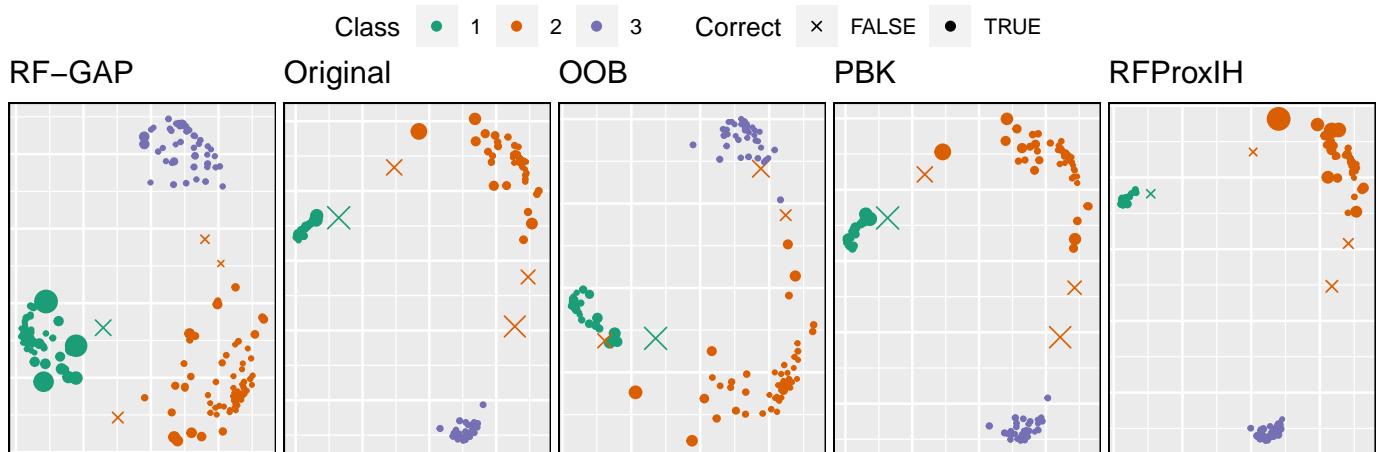


Fig. A.6. The wine dataset consists of three classes (corresponding to locations of cultivation) and 13 features. The tight clusters of each class using the original, PBK, and RFProxIH proximities suggests overfitting to the training data. It seems RF-GAP may show more sensitivity to outliers.

TABLE A.1

Complete results for data imputation using random forest proximities using a single iteration. For each considered dataset, values were removed completely at random in the amounts of 5%, 10%, 25%, 50% and 75%. Missing values were imputed using various proximity definitions (PBK was not used here for computational considerations). The experiment was repeated 100 times for all datasets. The two numbers directly below each dataset indicates the number of observations and number of variables, respectively. The average of the mean-squared errors (MSE) between the original and imputed values are recorded along with the standard errors. For missing value percentages up through 50%, RF-GAP proximities (Definition 3) provided the most accurate imputation across all datasets. At 75% missing values, RF-GAP proximities outperformed across 69% of datasets. Otherwise, the RF-GAP proximities are in second place. Note: some datasets were not compatible with RFProxIH due to continuous response or categorical features vectors.

Data	Proximity	5%	10%	25%	50%	75%
Arrhythmia 452 279	RF-GAP	8.35 ± 0.03	11.76 ± 0.02	18.994 ± 0.02	27.675 ± 0.02	35.489 ± 0.02
	OOB	8.765 ± 0.02	12.303 ± 0.02	19.586 ± 0.02	28.24 ± 0.02	35.302 ± 0.01
	Original	8.724 ± 0.02	12.239 ± 0.03	19.537 ± 0.02	28.196 ± 0.02	35.662 ± 0.02
	RFProxIH	8.75 ± 0.03	12.272 ± 0.02	19.622 ± 0.02	28.333 ± 0.02	35.952 ± 0.02
Balance Scale 645 4	RF-GAP	3.868 ± 0.02	5.468 ± 0.02	8.784 ± 0.02	12.348 ± 0.02	15.007 ± 0.02
	OOB	3.957 ± 0.02	5.633 ± 0.03	8.965 ± 0.02	12.49 ± 0.02	15.054 ± 0.02
	Original	3.942 ± 0.02	5.585 ± 0.02	8.922 ± 0.03	12.433 ± 0.02	15.002 ± 0.03
	RFProxIH	3.913 ± 0.02	5.613 ± 0.03	8.925 ± 0.02	12.45 ± 0.02	15.021 ± 0.03
Banknote 1372 5	RF-GAP	2.503 ± 0.02	3.558 ± 0.01	5.863 ± 0.01	8.881 ± 0.02	11.398 ± 0.01
	OOB	2.635 ± 0.02	3.81 ± 0.02	6.239 ± 0.01	9.283 ± 0.02	11.696 ± 0.01
	Original	2.643 ± 0.01	3.773 ± 0.01	6.214 ± 0.02	9.253 ± 0.02	11.664 ± 0.01
	RFProxIH	2.647 ± 0.01	3.765 ± 0.02	6.202 ± 0.02	9.277 ± 0.01	11.703 ± 0.01
Diabetes 678 8	RF-GAP	2.556 ± 0.02	3.693 ± 0.02	5.949 ± 0.02	8.584 ± 0.01	10.696 ± 0.01
	OOB	2.634 ± 0.02	3.78 ± 0.02	6.145 ± 0.01	8.809 ± 0.01	10.801 ± 0.01
	Original	2.67 ± 0.02	3.808 ± 0.02	6.193 ± 0.02	8.831 ± 0.02	10.797 ± 0.01
	RFProxIH	2.648 ± 0.02	3.833 ± 0.02	6.232 ± 0.02	8.876 ± 0.01	10.808 ± 0.01
Ecoli 336 8	RF-GAP	1.201 ± 0.02	1.737 ± 0.02	2.725 ± 0.02	4.009 ± 0.02	5.097 ± 0.02
	OOB	1.276 ± 0.02	1.762 ± 0.03	2.907 ± 0.02	4.092 ± 0.02	5.173 ± 0.04
	Original	1.202 ± 0.02	1.787 ± 0.02	2.815 ± 0.02	4.029 ± 0.02	5.129 ± 0.03
	RFProxIH	NA	NA	NA	NA	NA
Glass 214 10	RF-GAP	1.188 ± 0.02	1.731 ± 0.02	2.893 ± 0.02	4.286 ± 0.02	5.632 ± 0.02
	OOB	1.278 ± 0.02	1.805 ± 0.02	3.031 ± 0.02	4.397 ± 0.02	5.634 ± 0.01
	Original	1.253 ± 0.02	1.776 ± 0.02	3.013 ± 0.02	4.432 ± 0.02	5.67 ± 0.02
	RFProxIH	1.29 ± 0.02	1.782 ± 0.02	3.02 ± 0.02	4.451 ± 0.02	5.69 ± 0.02
Hill Valley 606 101	RF-GAP	2.381 ± 0.02	7.565 ± 0.05	16.596 ± 0.03	24.879 ± 0.02	31.498 ± 0.02
	OOB	2.887 ± 0.03	9.032 ± 0.03	17.298 ± 0.03	25.883 ± 0.02	32.946 ± 0.02
	Original	3.458 ± 0.03	9.268 ± 0.03	17.397 ± 0.03	26.038 ± 0.02	33.047 ± 0.02
	RFProxIH	3.771 ± 0.03	9.538 ± 0.03	17.538 ± 0.03	26.061 ± 0.02	33.056 ± 0.02
Ionosphere 351 34	RF-GAP	5.467 ± 0.02	7.707 ± 0.02	12.585 ± 0.02	18.765 ± 0.02	24.115 ± 0.01
	OOB	5.842 ± 0.02	8.245 ± 0.02	13.43 ± 0.02	19.683 ± 0.02	24.669 ± 0.01
	Original	5.899 ± 0.02	8.342 ± 0.02	13.563 ± 0.02	19.805 ± 0.02	24.681 ± 0.01
	RFProxIH	NA	NA	NA	NA	NA
Iris 150 4	RF-GAP	0.625 ± 0.01	0.85 ± 0.01	1.365 ± 0.01	1.978 ± 0.01	2.525 ± 0.01
	OOB	0.633 ± 0.01	0.883 ± 0.01	1.405 ± 0.01	1.998 ± 0.01	2.52 ± 0.01
	Original	0.631 ± 0.01	0.875 ± 0.01	1.395 ± 0.01	2.019 ± 0.01	2.528 ± 0.01
	RFProxIH	0.641 ± 0.01	0.884 ± 0.01	1.425 ± 0.01	2.027 ± 0.01	2.543 ± 0.01
Lymphography 148 18	RF-GAP	3.617 ± 0.02	5.288 ± 0.02	8.442 ± 0.02	12.337 ± 0.02	15.692 ± 0.03
	OOB	3.698 ± 0.02	5.432 ± 0.02	8.67 ± 0.02	12.563 ± 0.02	15.713 ± 0.02
	Original	3.638 ± 0.02	5.434 ± 0.03	8.653 ± 0.02	12.512 ± 0.02	15.691 ± 0.02
	RFProxIH	3.706 ± 0.02	5.415 ± 0.02	8.745 ± 0.02	12.495 ± 0.02	15.743 ± 0.02
Optdigits 5620 64	RF-GAP	18.746 ± 0.02	26.864 ± 0.02	44.191 ± 0.02	66.301 ± 0.02	85.627 ± 0.02
	OOB	20.598 ± 0.02	29.437 ± 0.02	48.028 ± 0.02	70.513 ± 0.02	88.248 ± 0.01
	Original	20.629 ± 0.02	29.578 ± 0.02	48.211 ± 0.02	70.84 ± 0.02	88.425 ± 0.01
	RFProxIH	20.685 ± 0.02	29.63 ± 0.02	48.453 ± 0.02	71.023 ± 0.02	88.483 ± 0.01
Parkinsons 197 23	RF-GAP	2.084 ± 0.02	3.044 ± 0.02	4.926 ± 0.02	7.456 ± 0.02	9.593 ± 0.01
	OOB	2.258 ± 0.02	3.275 ± 0.02	5.318 ± 0.01	7.801 ± 0.01	9.76 ± 0.01
	Original	2.271 ± 0.02	3.297 ± 0.02	5.392 ± 0.02	7.898 ± 0.01	9.797 ± 0.01
	RFProxIH	NA	NA	NA	NA	NA
Seeds 210 7	RF-GAP	0.924 ± 0.01	1.306 ± 0.01	2.159 ± 0.01	3.211 ± 0.01	4.144 ± 0.01
	OOB	1 ± 0.01	1.409 ± 0.01	2.279 ± 0.01	3.302 ± 0.01	4.177 ± 0.01
	Original	1.01 ± 0.01	1.442 ± 0.01	2.343 ± 0.01	3.382 ± 0.01	4.232 ± 0.01
	RFProxIH	1.035 ± 0.01	1.46 ± 0.01	2.358 ± 0.01	3.416 ± 0.01	4.267 ± 0.01
Sonar 208 60	RF-GAP	4.392 ± 0.01	6.488 ± 0.01	10.586 ± 0.02	15.723 ± 0.01	19.951 ± 0.01
	OOB	4.638 ± 0.02	6.78 ± 0.02	10.981 ± 0.01	16.055 ± 0.01	20.117 ± 0.01
	Original	4.673 ± 0.02	6.857 ± 0.02	11.075 ± 0.01	16.167 ± 0.01	20.167 ± 0.01
	RFProxIH	4.723 ± 0.02	6.914 ± 0.02	11.202 ± 0.02	16.169 ± 0.01	20.168 ± 0.01
Waveform 5000 21	RF-GAP	12.915 ± 0.01	18.328 ± 0.01	29.267 ± 0.01	42.239 ± 0.01	52.91 ± 0.01
	OOB	13.313 ± 0.01	18.93 ± 0.01	30.316 ± 0.01	43.544 ± 0.01	53.753 ± 0.01
	Original	13.342 ± 0.01	18.969 ± 0.01	30.366 ± 0.01	43.584 ± 0.01	53.768 ± 0.01
	RFProxIH	NA	NA	NA	NA	NA
Wine 178 13	RF-GAP	1.525 ± 0.01	2.216 ± 0.01	3.503 ± 0.01	5.058 ± 0.01	6.38 ± 0.01
	OOB	1.603 ± 0.01	2.254 ± 0.01	3.544 ± 0.01	5.097 ± 0.01	6.38 ± 0.01
	Original	1.577 ± 0.01	2.273 ± 0.01	3.545 ± 0.01	5.095 ± 0.01	6.379 ± 0.01
	RFProxIH	1.593 ± 0.01	2.271 ± 0.01	3.559 ± 0.01	5.103 ± 0.01	6.38 ± 0.01

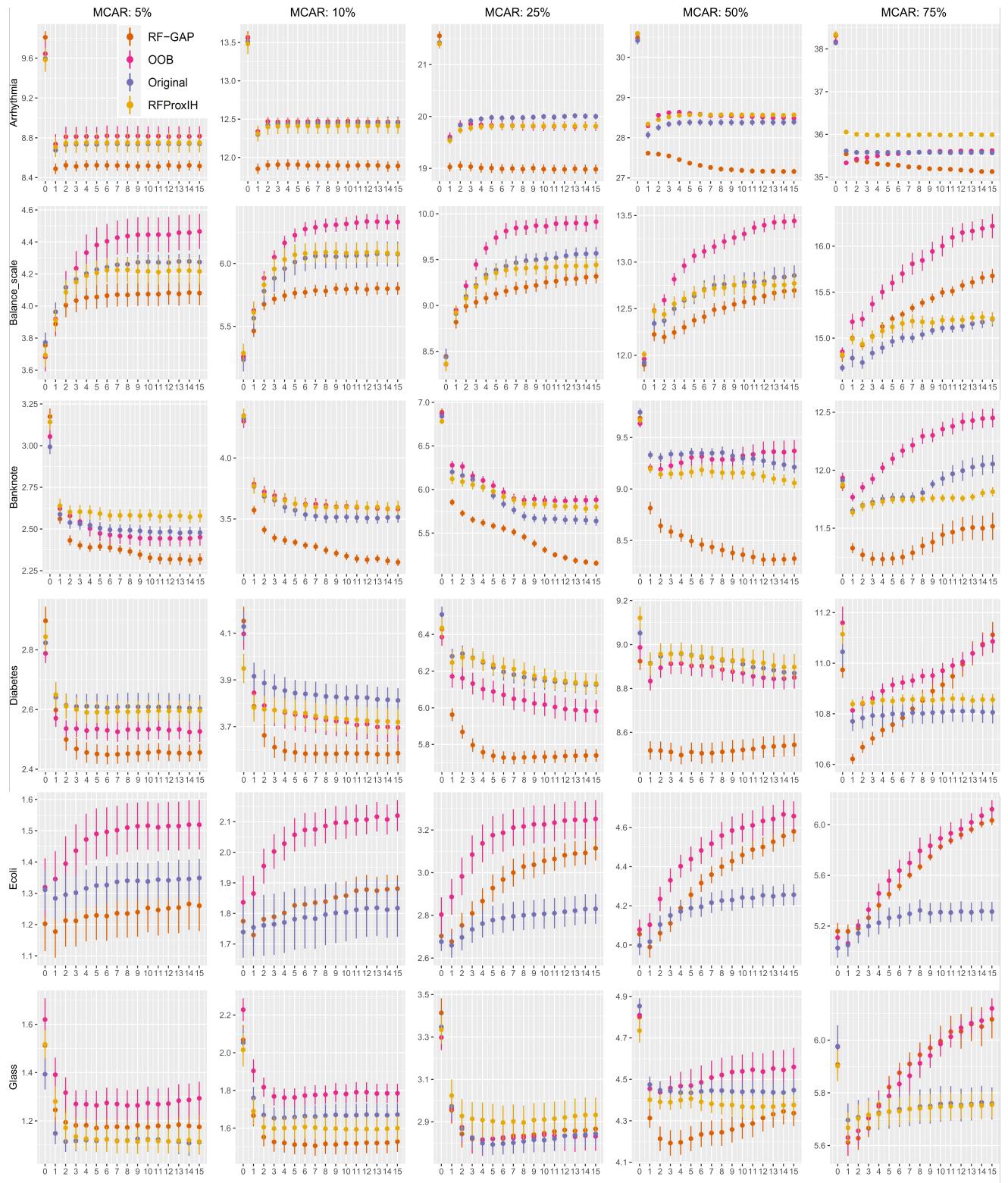


Fig. A.7. Additional imputation results. See Fig. A.9 for more details.

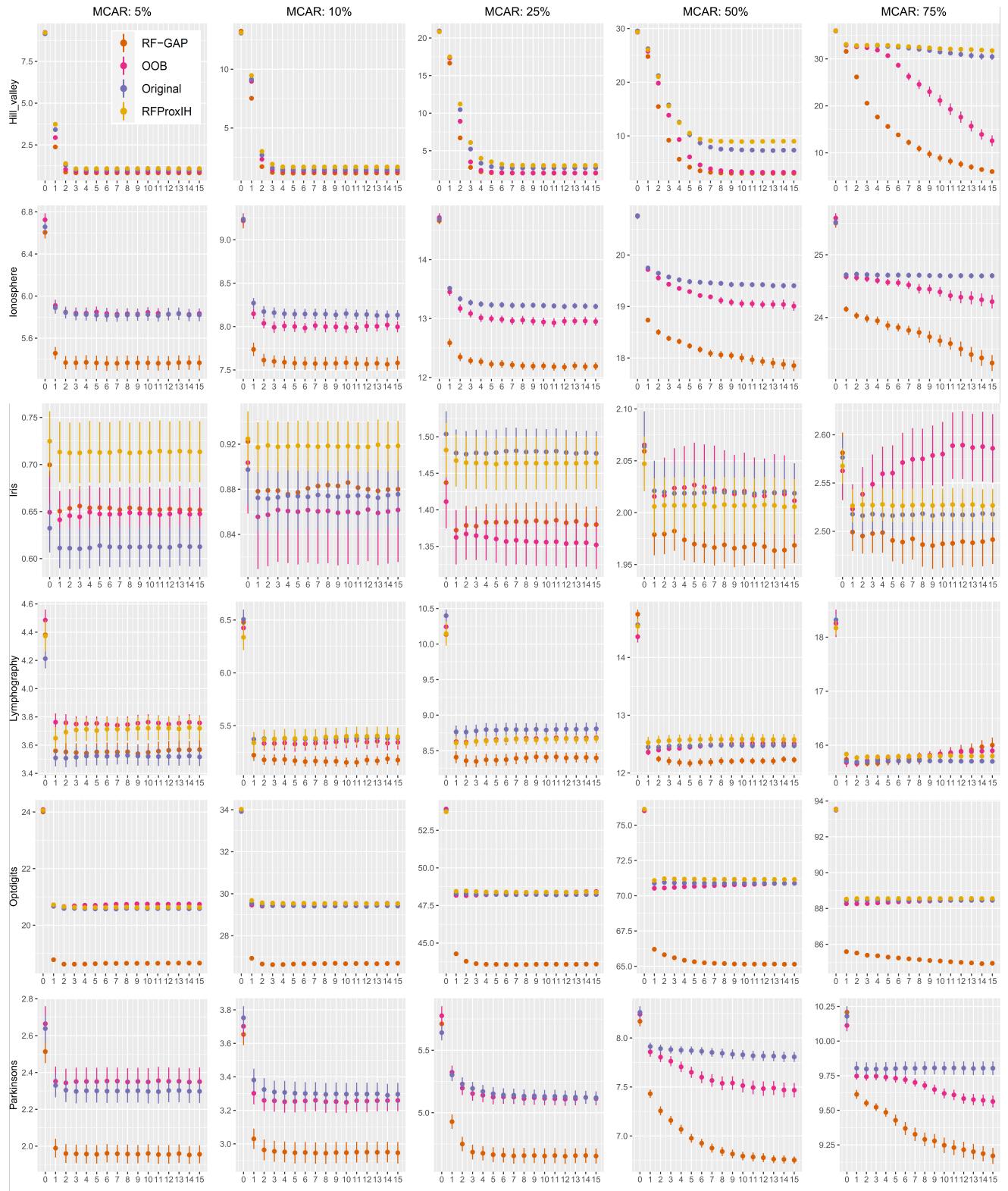


Fig. A.8. Additional imputation results. See Fig. A.9 for more details.

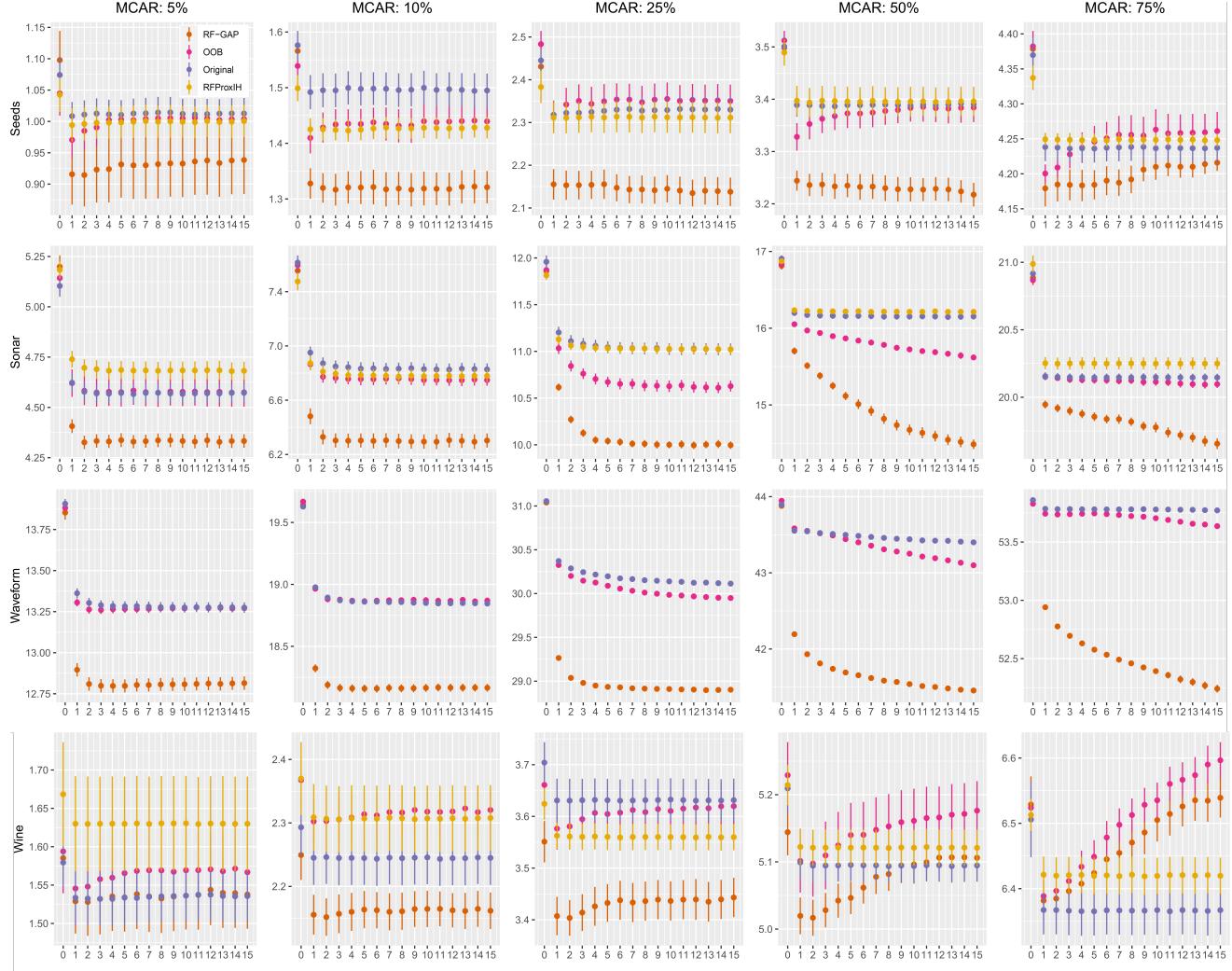


Fig. A.9. This figure, Fig. A.7 and A.8 give the mean-squared error (MSE) between the original and imputed values using four random forest proximity measures. All data variables were scaled from 0 to 1 for comparability. The scores were compared using 1 to 15 imputation iterations as described in Section 6.2 and each experiment was conducted over 10 repetitions using the original proximities, OOB proximities, RF-GAP proximities, and RFProxiH. Imputation 0 provides the MSE for the median-filled imputation. Four different percentages of values missing completely at random (MCAR) were used (5%, 10%, 25%, and 50%) across several datasets from the UCI repository [61]. In general, RF-GAP outperforms the other proximity-weighted imputations. See additional results in Table A.1 for a single iteration.