Appendix B Random Forests Example

The following example has been designed to illustrate the steps of the Random Forests (RF) clustering approach used in this investigation. The data used for this purpose were randomly generated to represent measurements of boron and calcium for samples from both impacted and unimpacted locations. The real data are measured on a much larger list of variables, and do not necessarily exhibit such clear separation between impacted and unimpacted samples. However, this simple example offers a clearer view of the RF clustering process than the real, and far more complicated, data could. The example data are illustrated in **Figure B-1**:

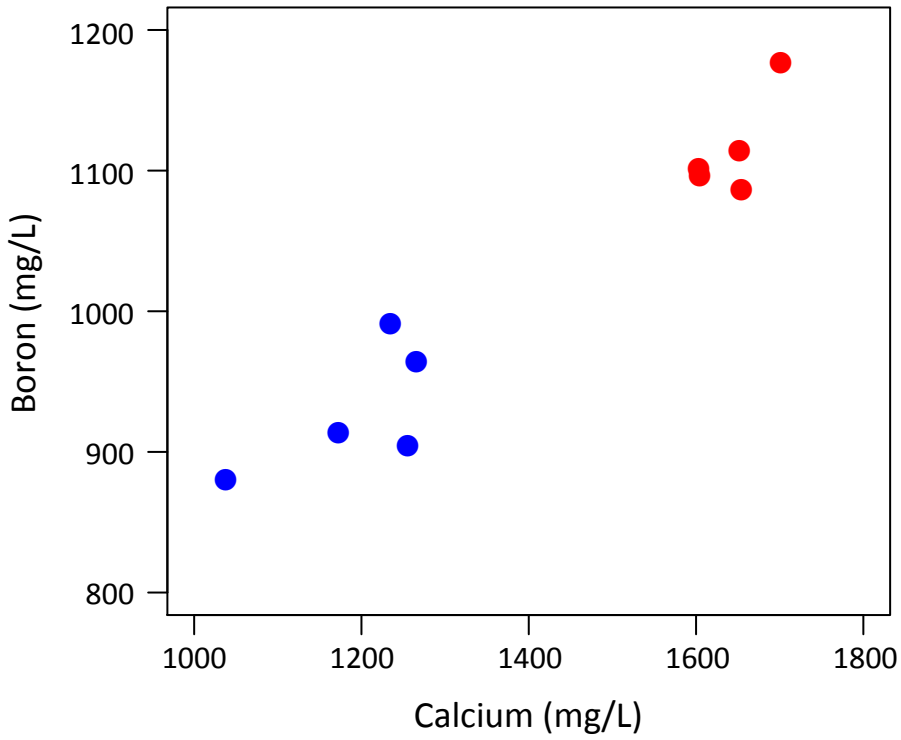


Figure B-1. Example dataset randomly generated to represent water samples from "impacted" (red) and "unimpacted" (blue) wells, measured for Boron and Calcium. These data were made up for the purpose of the illustration—results from this example should not be taken to imply anything about true values of Boron or Calcium in impacted versus unimpacted water samples.

Note that the samples have been color-coded based on the "true" grouping prescribed for the purpose of the example. However, the crux of the clustering problem in this study is that, in fact, the true impacted status of the samples is unknown. To reflect this, in subsequent figures the data are not distinguished by color.

The first step in the RF clustering algorithm is to generate a synthetic dataset by making random draws from the values of each variable in the real data[6]. An example is shown in **Figure B-2**.

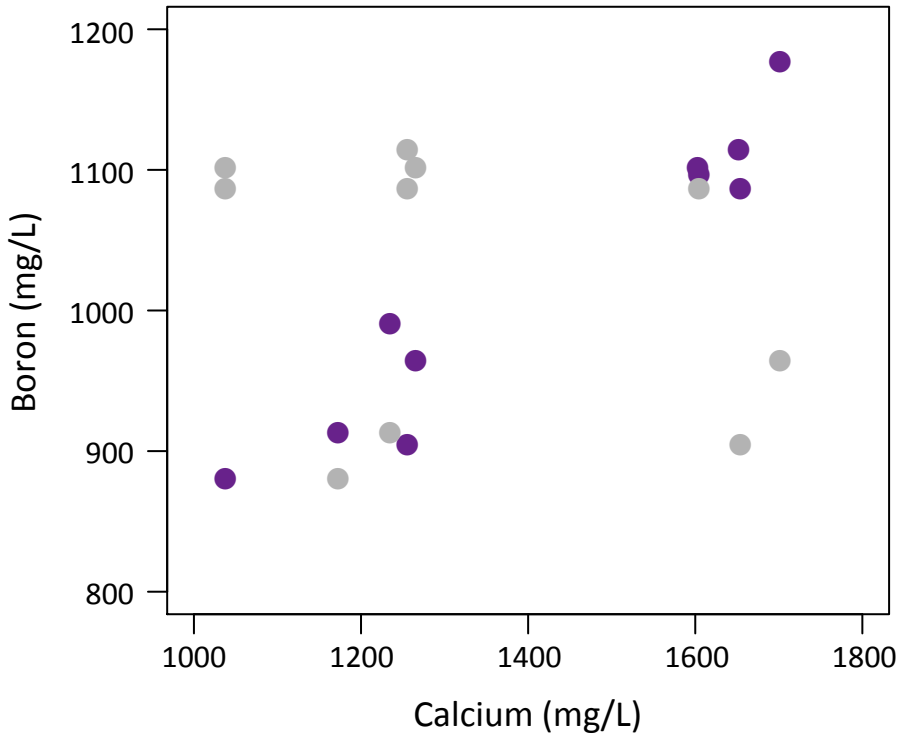


Figure B-2. The first step in the Random Forests clustering algorithm is to generate a synthetic dataset (grey), based on the real data of interest (purple). Note that the latter are the same data as in Fig. 1, recolored to reflect that the true impacted versus unimpacted status of the samples is unknown during a typical analysis.

As can be seen here, the RF clustering algorithm calls for generating the same number of synthetic samples as there are in the real dataset (in this case, 10). Because the synthetic data were drawn randomly from each variable *independently*, they do not exhibit the same correlation structure as the real data. In this example, values for Boron and Calcium are correlated in the real data—that is, high values for one variable tend to be associated with high values for the other, and vice versa. By contrast, the synthetic data are just as likely to have high values of one variable paired with low values of the other. Ultimately, it is this distinction—structure in the real data, versus none in the synthetic—that allows the RF analysis to "learn" about the nature of the real data.

The next step in the method is to construct an RF model that tries to separate the real and synthetic data. An RF is comprised of hundreds or thousands of decision trees, mathematical

[6] In this appendix, the 10 data points shown in Fig. B-1 are considered the "real" data. While these data were in fact fabricated for this illustrative example, they represent real samples that might be collected from the field. Most importantly, they exhibit correlation structure among the measured variables (in this case, high values of Boron are associated with high values of Calcium, and vice versa), as would be typical for real field data. By contrast, the "synthetic" data generated as part of the RF clustering algorithm (grey dots in Fig. B-2) show no such relationship.

devices that attempt to classify a group of samples by splitting it repeatedly based on the values of measured variables. An example of one such tree is shown here:

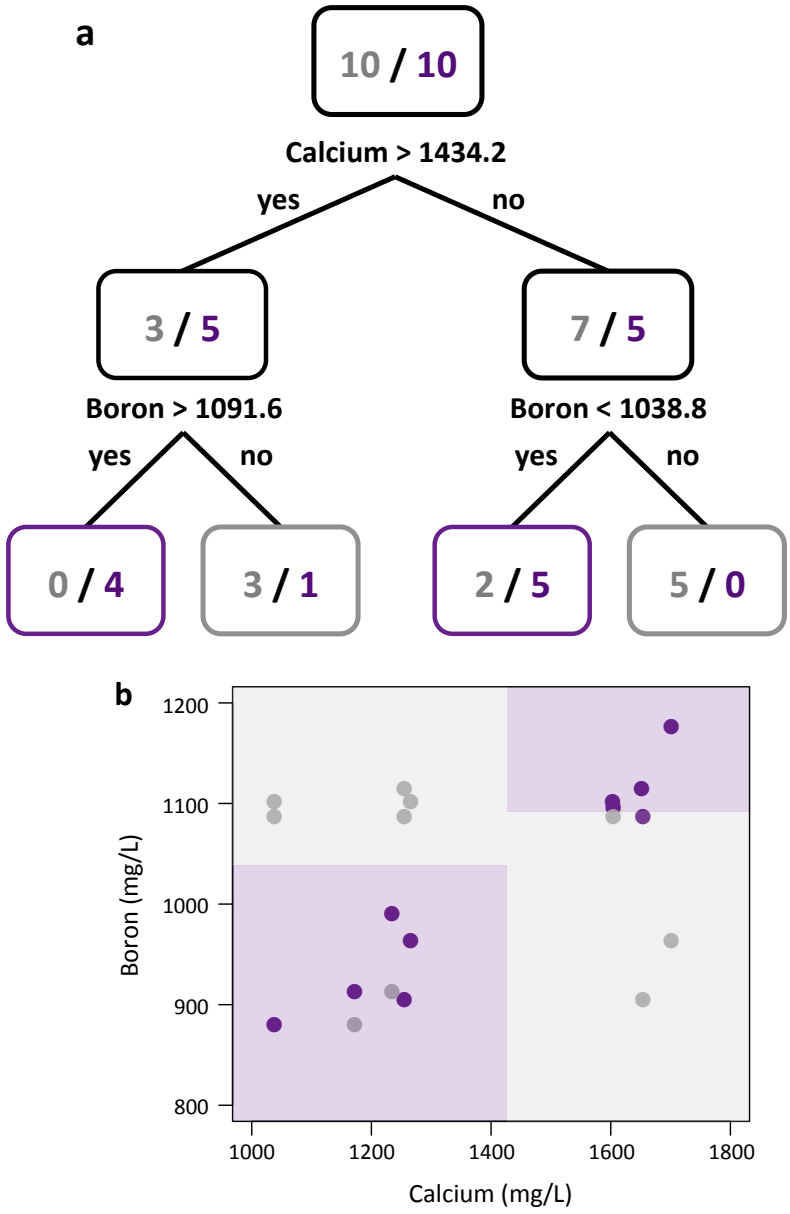


Figure B-3. A Random Forests analysis consists of a collection of many individual decision trees. One such tree is illustrated here. (a) The tree, in which boxes represent *nodes* containing real and/or synthetic samples (shown by purple and grey numbers, respectively). The tree is built by splitting each node based on a particular value for one of the variables in the dataset (criteria beneath boxes). The final *leaf* nodes are assigned to whichever class they predominantly contain (colored outlines); these are the classes the tree would predict for samples classified into each leaf based on the splitting criteria. (b) An alternative representation of the tree. The splits in (a) are translated into divisions along the calcium and boron axes, which together define regions in which the tree predicts samples are either real (purple) or synthetic (grey). In both (a) and (b), samples whose color matches the leaf / region in which they fall have been correctly classified by the tree (17/20, in this example); the remaining samples (3/20) have been misclassified.

The tree in **Figure B-3**a was produced in steps. In the first step, there are 10 samples labeled "real" and 10 labeled "synthetic". The decision tree algorithm considered all possible "splits" along the Calcium and Boron axes, and found that the best[7] possible split was on Calcium, at a value of 1434.2 mg/L. Defining this split produced two new groups, each of which is somewhat more segregated than the original. The process was then repeated to split each of the resulting groups further, leading to four total groups or "leaves" that are fairly homogeneous in terms of their real versus synthetic makeup[8]. In **Figure B-3**a, the leaves are color-coded to match the class to which the majority of their observations belong. For each leaf, this is the tree's prediction of class membership for any sample that belongs in that particular leaf based on the classification rules defined by the tree.

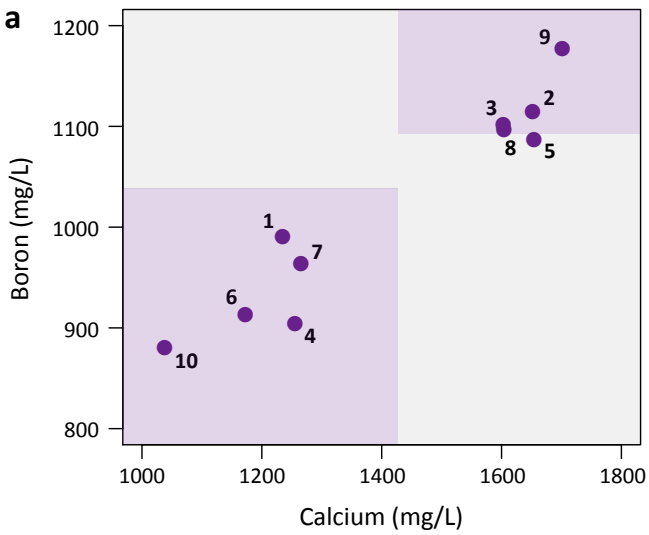
For this simple example with only two measured variables, the decision tree can also be illustrated in the scatterplot of the data. As shown in **Figure B-3**b, each split in the decision tree corresponds to dividing the data plane horizontally or vertically, and the four shaded regions defined by these divisions correspond to the four leaves of the tree. In both views, it is clear that classification is not perfect—two synthetic samples are predicted to be real in the lower left quadrant, and one real sample is identified as synthetic in the lower right. However, there is enough structure in the real data that the RF predictions are accurate for the majority of samples, which is all that is required to obtain meaningful clustering results.

The next step in the RF clustering algorithm is to construct a proximity matrix from the decision trees used to classify the real and synthetic data. Again, the single tree from above (**Figure B-3**) is used for illustration. In **Figure B-4**a, the data are again illustrated with shading to indicate the tree's real and synthetic class predictions. Labels have been added to the real data for clarity, and the synthetic data have been removed as they are not considered further in the analysis once their role in constructing the decision trees has been fulfilled.

Figure B-4b shows the proximity matrix for the example decision tree. In this matrix, each row and column represents one of the samples, such that every entry in the matrix corresponds to a pair of samples (e.g., the entry in the third row and fourth column represents Sample 3 paired with Sample 4). Entries are defined to be *1* if the pair of samples they represent are classified into the same leaf in the decision tree, and *0* otherwise. This is again easy to illustrate in the scatterplot of the data in **Figure B-4**a—sample pairs receive a *1* if they are in the same shaded region, and a *0* if not. Note that every diagonal entry of the matrix must contain a *1* by this definition of proximity, since every sample is in the same leaf/shaded region as itself. Note also that such a proximity matrix is symmetrical (that is, the entry [i, j] is equal to entry [j, i] for any row/column numbers *i* and *j*), but for clarity only half of the example matrix is filled in.

[7] In terms of making each of the resulting groups as homogeneous as possible.

[8] In practice, the tree-building algorithm actually proceeds further, until all leaves contain only one class of observation. For the purpose of this example, however, this partial tree provides a better illustration of key concepts.



b

	1	2	3	4	5	6	7	8	9	10
1	1									
2	0	1								
3	0	1	1							
4	1	0	0	1						
5	0	0	0	0	1					
6	1	0	0	1	0	1				
7	1	0	0	1	0	1	1			
8	0	1	1	0	0	0	0	1		
9	0	1	1	0	0	0	0	1	1	
10	1	0	0	1	0	1	1	0	0	1

Figure B-4. Decision trees are used to construct a proximity matrix. (a) The data from Fig. 3b, with synthetic samples removed for clarity (they are used only for tree construction, and discarded thereafter), and sample numbers added for reference. (b) The proximity matrix for the data, in which row and column numbers correspond to the sample numbers in (a). Each entry in the matrix is *1* (shaded green) if it corresponds to a pair of samples that are classified in the same leaf node in the decision tree (i.e., fall into the same shaded region in (a)), and *0* otherwise (shaded light grey). For clarity, only half of the symmetric proximity matrix is filled in here.

As noted above, an RF analysis consists of repeating the tree-building and proximity-calculating steps many times[9]. To obtain a final proximity matrix the matrices from all trees are averaged together. A *1* or *0* in the final matrix would indicate samples that were classified together in all or none of the trees, respectively. In practice, most entries in the final matrix will be somewhere in between, and thus represent the degree of similarity (that is, *proximity*) between each pair of samples. The final RF proximity matrix for this example dataset is shown here:

	1	2	3	4	5	6	7	8	9	10
1	1									
2	0.00	1								
3	0.03	0.70	1							
4	0.63	0.00	0.03	1						
5	0.08	0.44	0.29	0.06	1					
6	0.74	0.00	0.01	0.69	0.02	1				
7	0.76	0.01	0.09	0.68	0.15	0.37	1			
8	0.03	0.62	0.93	0.03	0.44	0.01	0.09	1		
9	0.00	0.81	0.36	0.00	0.48	0.00	0.01	0.31	1	
10	0.20	0.00	0.00	0.37	0.00	0.71	0.06	0.00	0.00	1

Figure B-5. A Random Forests proximity matrix is the mean across proximity matrices generated by many decision trees, which differ from one another due to random factors introduced into the analysis. The result for this example is shown here. Shading indicating same (green) or different (light grey) leaf node classification is retained from the single tree's proximity matrix shown in Fig. 4b. In most cases, the Random Forests outcome agrees with the individual tree, as indicated by the former showing a high value for sample pairs that were in the same leaf node in the example tree, and vice versa. Some samples that were classified together in the single tree, however, have relatively low (<0.5) proximity according to the Random Forests output (yellow circles). In general, the latter is more reliable because it is based on many trees, and thus avoids overfitting that may be associated with any one of them.

[9] As described in Appendix A, each tree in the forest has randomness added to it in several ways. This prevents the trees from being identical, and in general leads to better predictive power for a Random Forests analysis than for any one decision tree alone. Incidentally, it is also the origin of the name *Random* Forest.

This matrix is shaded to match the proximity scores from the single decision tree shown above. Thus it can be seen that most of the samples that were classified together in the example tree have high proximity overall in the RF results, and vice versa. There are exceptions however (circled in yellow), reflecting that a single tree may have a tendency to conform too closely to the particular dataset at hand (i.e., to be "overfit"). In general, where the RF differs from the single tree, the former can be expected to yield more robust information about the dataset. For example, the individual tree classified Sample 10 together with Samples 1 and 7, whereas the RF indicates that Sample 10 is only weakly related to these samples. Visual inspection of Fig. 4a would seem to support the RF interpretation—Samples 1 and 7 are not much closer to Sample 10 than they are to the cluster in the upper right quadrant of the plot.

The final step in RF clustering is to use the proximity matrix as input to the Partitioning Around Medoids (PAM) algorithm. PAM searches among a set of samples with known proximity to one another, to identify the samples that are most representative of natural groupings in the data. In this example, for instance, PAM was run to identify two clusters, which might be conceptualized to represent impacted and unimpacted samples (the rationale used in this study). The algorithm considered different pairs of samples to serve as the central points or *medoids* of these clusters, and for each possible pair calculated an overall score representing how similar all the other samples are to their nearest medoid.

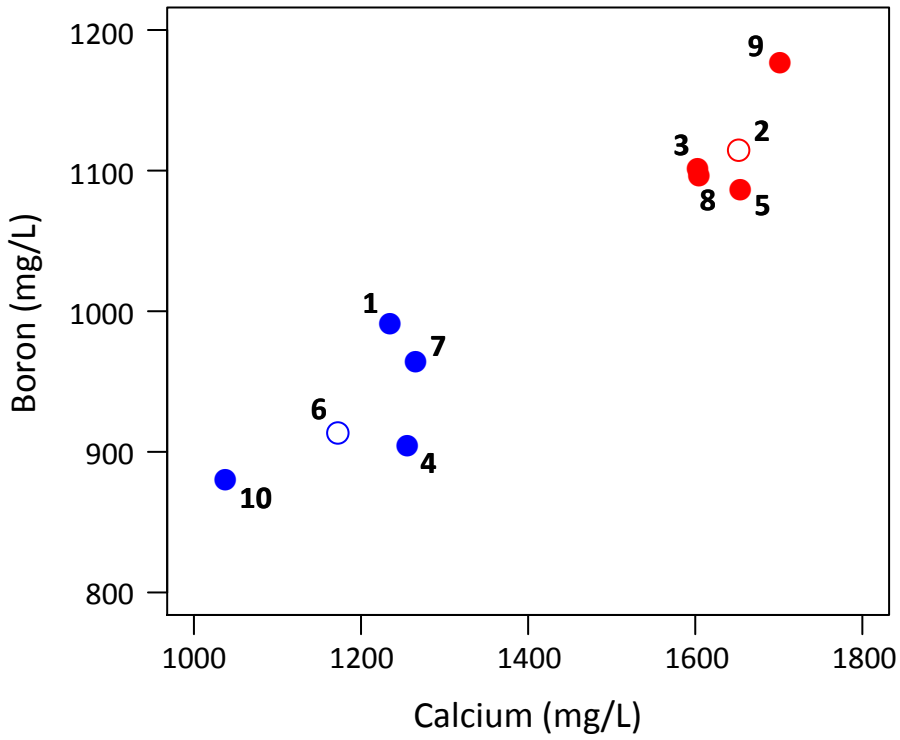


Figure B-6. In the final step of the clustering analysis, the Random Forests proximity matrix (Figure B-5) is used as input to a Partitioning Around Medoids (PAM) algorithm. PAM divides a dataset into a specified number of clusters (two, in this example) by identifying samples to serve as the center points ("medoids") of each cluster. Once medoids are chosen, samples are assigned to the cluster whose medoid they are nearest to, according to the proximity matrix. The algorithm searches for the optimal medoids around which to cluster (open circles), defined as those that maximize the total proximity between each sample and its assigned medoid. In this example, PAM identifies the same two clusters (red and blue dots) that were actually imposed in this artificial dataset (Figure B-1).

As shown in **Figure B-6**, for this example Samples 2 and 6 were identified as the best medoids to represent a two-cluster partitioning of the data. Assigning the remaining samples to their nearest medoid results in a visually intuitive clustering that in fact corresponds to the known model used to generate the data for this example (**Figure B-1**). Thus, the completed RF clustering algorithm was able to obtain an accurate clustering pattern for the data in this test case where the true clusters were known. While it is impossible to be sure of clustering outcomes in the real analysis, where true impacted versus unimpacted status is unknown, reliable performance in test cases such as the one illustrated here builds confidence that the approach is robust.

Finally, it is worth acknowledging that the example shown here is intentionally contrived to illustrate the RF clustering method. In this simple case, the clusters identified by the algorithm are quite distinct, and could easily have been obtained by various other, less complicated techniques. However, in a real dataset, with perhaps thousands of samples measured on dozens of variables, results will not be so obvious. Careful analysis has shown RF clustering to have distinct advantages over alternative methods in these cases, but demonstrating those advantages is beyond the scope of this simple example.

Appendix C Boxplots for Initial Stratigraphic Layers