6
Random and Deterministic Forests

Forest-based classification and prediction is one of the most commonly used nonparametric statistical methods in many scientific and engineering areas, particularly in machine learning and analysis of high-throughput genomic data. In this chapter, we first introduce the construction of random forests and deterministic forests, and then address a fundamental and practical issue on how large the forests need to be.

6.1 Introduction to Random Forests

We have seen that tree-based data analyses are readily interpretable. However, tree-based methods have their limitations. First, tree structure is prone to instability even with minor data perturbations. This is generally the case for all stepwise model selection procedures. Second, thanks to the advancement of genomics and informatics, high-dimensional data are very common. As illustrated in Example 1.7, many studies use tens of thousands of gene expressions to predict an outcome using several tens or hundreds of subjects. This phenomenon with a large number of variables and limited number of observations is commonly referred to as the "large p and small n" problem (e.g., Kosorok and Ma 2007; Zhang et al. 2008). To leverage the richness of a data set of massive size, we need to broaden the classic statistical view of "one parsimonious model" for a given data set. Third, due to the adaptive nature of the tree construction, theoretical inference based on a tree is usually not feasible. Generating more trees may

H. Zhang and B.H. Singer, *Recursive Partitioning and Applications*,
Springer Series in Statistics, DOI 10.1007/978-1-4419-6824-1_6,

provide an empirical solution to statistical inference (Zhang 1998a; also see Chapter 12).

To address these limitations, the method of forests has emerged as an ideal solution. Here, a forest refers to a constellation of any number of tree models. Such an approach is also referred to as an *ensemble*. In general, a forest consists of hundreds or thousands of trees, so it is more stable and less prone to prediction errors as a result of data perturbations (Breiman 1996, 2001). While each individual tree is not a good model, combining them into a committee improves their value. It is important to note that trees in a forest should not be pruned to the "smallest" size level as described in Section 2.3. In fact, as discussed by Breiman (1996, 2001), it would be counterproductive to pool "good" models into a committee.

From a practical point of view, having many trees also provides us with an opportunity to utilize more information (i.e., more variables) in the data set, and hence we can seek more insights into and have a deeper understanding of the data. In some applications, different trees may unravel alternative pathways to disease prognosis or development.

How is a random forest constructed? Suppose that we have n observations and p predictors. The following is the algorithm:

1 Draw a bootstrap sample. Namely, sample n observations with replacement from the original sample.

2 Apply recursive partitioning to the bootstrap sample. At each node, randomly select q of the p predictors and restrict the splits based on the random subset of the q variables. Here, q should be much smaller than p.

3 Let the recursive partitioning run to the end and generate a tree.

4 Repeat Steps 1 to 3 to form a forest. The forest-based classification is made by majority vote from all trees.

If Step 2 is skipped, the above algorithm is called *bagging* (bootstraping and aggregating) (Breiman 1996). Bagging should not be confused with another procedure called *boosting* (Freund and Schapire 1996). One of the boosting algorithms is *Adaboost*, which makes use of two sets of intervening weights. One set, w, weighs the classification error for each observation, and the other, β, weighs the voting of the class label. Boosting is an iterative procedure, and at each iteration, a model (e.g., a tree) is built. It begins with an equal w-weight for all observations. Then, the β-weights are computed based on the w-weighted sum of error, and w-weights are updated with β-weights. With the updated weights, a new model is built and the process continues. Unlike bagging, boosting generally builds a very simple model such as a tree with one split. According to Leo Breiman's Wald Lecture, boosting does not perform as well as bagging. More relevant for the spirit of this book, boosting inhibits interpretation. Indeed, the repeated sampling in bagging facilitates exposure of subpopulations/groups with distinctive characteristics.

In forest construction, several practical questions often arise. Here, we discuss some of those issues. Firstly, how many trees do we need in a forest? Breiman (2001) chose to run 100 trees in several examples and others have used much larger numbers. We will discuss in Section 6.2 as to how large a random forest needs to be. As Breiman (2001) noted, the accuracy of a random forest depends on two key factors: the prediction strength of the individual trees and the correlation of the trees. Thus, we may keep the size of a random forest to the minimal level if the trees can achieve the highest strength and have the weakest correlation.

Secondly, does a random forest overfit the data without pruning the individual trees? Breiman (2001) showed that there is no overfitting issue by the Strong Law of Large Numbers. The prediction error of a random forest converges as the size of the forest increases, and the error has an upper bound that is directly related to the strength and the correlation of the trees in the forest.

Thirdly, selecting the subset of q variables in node splitting is an important feature of random forests. Commonly used choices are $log(p)$ or $\sqrt{p}$. However, there is a caveat with this idea. For example, in genetic studies, we tend to have a huge number of genetic markers (on the order of a million) and some environment variables (ranging from one to hundreds). The environment variables have few chances to be selected in the random forest, not because they are not important, but because there are relatively so few of them. Furthermore, even among genetic markers, not all of them should be treated equally. Thus, in practice, we should be cautious about the fact that the random forest treats all predictors indiscriminately. In Section 6.5, we discuss some approaches to overcoming this issue.

Finally, after a forest is formed, how do we understand the information in the forest, especially if it is too large to examine the individual trees?

6.2 The Smallest Forest

Although the method of forests addresses the two challenges that the tree-based methods face, it also loses some of the advantages that the tree-based methods possess. Most importantly, because of so many trees in a forest, it is impractical to present a forest or interpret a forest. This is what Breiman referred to as a “black-box” in his 2002 Wald lectures presented at the annual meeting of the Institute of Mathematical Statistics. Zhang and Wang (2009) explored whether it is possible to find a common ground between a forest and a single tree so that we retain the easy interpretability of the tree-based methods and avoid the problems that the tree-based methods suffer from. In other words, does a forest have to be large, or how small can a forest be? To answer this fundamental question, the key idea is to shrink the forest with two objectives: (a) to maintain a similar (or even better) level of prediction accuracy; and (b) to reduce the number of the trees in the forest to a manageable level.

To shrink the size of a forest while maintaining the prediction accuracy, we need a criterion to determine the importance of a tree in a forest in terms of prediction performance. Zhang and Wang (2009) considered three options and found that the measure "by prediction" outperformed the others. Specifically, a tree is removed if its removal from the forest has the minimal impact on the overall prediction accuracy. First, calculate the prediction accuracy of forest F, denoted by p_F. Second, for every tree, denoted by T, in forest F, calculate the prediction accuracy of forest F_{-T} that excludes T, denoted by $p_{F_{-T}}$. Let Δ_{-T} be the difference in prediction accuracy between F and F_{-T}:

$$\Delta_{-T} = p_F - p_{F_{-T}}. \tag{6.1}$$

The tree T^p with the smallest $\Delta_{_T}$ is the least important one and hence subject to removal:

$$T^p = \underset{T \in F}{\arg\min}(\Delta_{-T}). \tag{6.2}$$

To select the optimal size subforest, Zhang and Wang (2009) track the performance of the subforests. Let $h(i), i = 1, \ldots, N_f - 1$, denote the performance trajectory of a subforest of i trees, where N_f is the size of the original random forest. Note that $h(i)$ is specific to the method measuring the performance, because there are many subforests with the same number of trees. If there is only one realization of $h(i)$, they select the optimal size i_{opt} of the subforest by maximizing $h(i)$ over $i = 1, \ldots, N_f - 1$:

$$i_{opt} = \underset{i=1,\ldots,N_f-1}{\arg\max}\ (h(i)). \tag{6.3}$$

If there are M realizations of $h(i)$, they select the optimal size subforest by using the 1-se as described by Breiman et al. (1984). That is, they first compute the average $\overline{h}(i)$ and its standard error $\hat{\sigma(i)}$:

$$\overline{h}(i) = \frac{1}{M} \sum_{j=1,\ldots,M} h_j(i), i = 1, \ldots, N_f - 1, \tag{6.4}$$

$$\hat{\sigma}(i) = \text{var}(h_1(i), \ldots, h_M(i)), i = 1, \ldots, N_f - 1. \tag{6.5}$$

Then, find the i_m that maximizes the average $\overline{h}(i)$ over $i = 1, \ldots, N_f - 1$:

$$i_m = \underset{i=1,\ldots,N_f-1}{\arg\max}\ (\overline{h}(i)). \tag{6.6}$$

As discussed by Breiman et al. (1984), the 1-se rule tends to yield a more robust and parsimonious model.

TABLE 6.1. Comparison of prediction performance of the initial random forest, the optimal subforest, and a previously established 70-gene classifier

		Predicted	Observed outcome	
Method	Error rate	outcome	Good	Poor
Initial random forest	26.0%	Good	141	17
		Poor	53	58
Optimal subforest	26.0%	Good	146	22
		Poor	48	53
Published classifier	35.3%	Good	103	4
		Poor	91	71

Finally, they choose the smallest subforest such that its corresponding $\overline{h}$ is within one standard error (se) of $\overline{h}(i_m)$ as the optimal subforest size i_{opt}:

$$i_{opt} = \min_{i=1,\ldots,M}(h(i) > (\overline{h}(i_m) - \hat{\sigma}(i_m)), \tag{6.7}$$

which is the critical point of the performance trajectory.

Using a microarray data set on Breast Cancer Prognosis (van de Vijver et al. 2002), Zhang and Wang (2009) examined several approaches to selecting the smallest forest. To begin the process, an initial forest is constructed using the whole data set as the training data set. As the first approach, one bootstrap data set is used for execution and the out-of-bag (oob) samples for evaluation. As the second approach, the oob samples are used for both execution and evaluation. As the third approach, the bootstrap samples are used for both execution and evaluation. Lastly, bootstrap samples are redrawn for execution and again redrawn for evaluation. It appears that the first approach works well for the Breast Cancer Prognosis data set that includes 288 samples, each of which contains the response variable defined by whether the patients remained disease-free five years after their initial diagnoses or not. Using the first approach and after replicating the bootstrapping procedure 100 times, they found that the sizes of the optimal subforests fall in a relatively narrow range, of which the 1st quartile, the median, and the 3rd quartile are 13, 26, and 61, respectively. This allows them to choose the smallest optimal subforest with the size of 7.

To compare the performance of the initial random forest with this optimal subforest, they used the two forests as classifiers in the original data set. Table 6.1 presents the misclassification rates based on the oob samples. The classifier proposed by van de Vijver et al. (2002) is included in the table as the benchmark.

Table 6.1 demonstrates that the optimal subforest, while much smaller, is comparable to the initial random forest in terms of prediction.

6.3 Importance Score

Unlike a tree, a forest is generally too overwhelming to interpret. One solution is to summarize or quantify the information in the forest, for example, by identifying "important" predictors in the forest. If important predictors can be identified, a random forest can also serve as a method of variable (feature) selection, and we can utilize other simpler methods such as classification trees by focusing on the important predictors. The question is: how do we know a predictor is important? To answer this question, various measures of variable importance have been proposed (e.g., Breiman 2001, Friedman 2001, Chen et al. 2007). In the following, we present several variable importance measures.

6.3.1 Gini Importance

During the course of building a forest, whenever a node is split based on variable k, the reduction in Gini index in (4.4) from the parent node to the two daughter nodes is added up for variable k, and this is done over all trees in the forest, giving rise to a simple variable importance score. Although Breiman noted that Gini importance is often very consistent with the permutation importance measure (`http://www.stat.berkeley.edu/~breiman/RandomForests`), others found it undesirable for being in favor of predictor variables with many categories (see, e.g., Strobl et al. 2007). This phenomenon appears similar to the undesirable end-cut preference problem discussed at the end of Section 4.1.

6.3.2 Depth Importance

Chen et al. (2007) introduced an importance index that is similar to Gini importance score, but considers the location of the splitting variable as well as its impact. Specifically, whenever node t is split based on variable k, let $L(t)$ be the depth of the node and $S(k,t)$ be the χ^2 test statistic from the variable, then $2^{-L(t)}S(k,t)$ is added up for variable k over all trees in the forest. Here, the depth is 1 for the root node, 2 for the offspring of the root node, and so forth. This depth importance measure was found useful in identifying genetic variants for complex diseases, although it is not clear whether it also suffers from the same end-cut preference problem.

6.3.3 Permutation Importance

The third importance index is the permutation importance, referred to as the variable importance. For each tree in the forest, we count the number of votes cast for the correct class. Then, we randomly permute the values of variable k in the oob cases and recount the number of votes cast for the correct class in the oob cases with the permuted values of variable k. The permutation importance is the average of the differences between the number of votes for the correct class in the variable-k-permuted oob data from the number of votes for the correct class in the original oob data, over all trees in the forest.

The permutation importance index is arguably the most commonly used choice. There are a few important issues to note. Firstly, the permutation importance index is not necessarily positive, and does not have an upper limit. Secondly, both the magnitudes and relative rankings of the permutation importance for predictors can be unstable when the number, p, of predictors is large relative to the sample size. This is certainly the case for genomic data. Thirdly, the magnitudes and relative rankings of the permutation importance for predictors vary according to the number of trees in the forest and the number, q, of variables that are randomly selected to split a node. As presented by Genuer et al. (2008), the effect of the number of trees in the forest is relatively minor, although more trees lead to better stability. However, the magnitude of the importance may increase dramatically as q increases, although the rankings may remain the same. To illustrate this, we simulated data based on a microarray data set on Breast Cancer Prognosis (van de Vijver et al. 2002). That study had 295 samples with 24,496 genes. We randomly selected four genes to generate a binary (e.g., normal or abnormal) outcome y. Let x_1, x_2, x_3, and x_4 be the expression intensities of the four selected genes. Then, the response is derived by $y = I(\sum_{i=1}^{4} x_i > 0)$; here recall that $I(\cdot)$ is the indicator function.

Figure 6.1 displays the importance scores of the four selected genes with a range of q's. Before the computation, genes with correlation greater than 0.1 with any of the four selected genes (in terms of the expression level) are removed, to avoid the potential effect of correlation. There are 1000 trees in the forest. Clearly, the importance score tends to increase as the q increases. However, the four genes keep the same order of importance. Without going into detail, we should note that the effect of the forest size on the importance scores is relatively minor.

Finally, there are conflicting numerical reports with regard to the possibility that the permutation importance overestimates the variable importance of highly correlated variables (see, e.g., Strobl et al. 2008 and Díaz-Uriarte and Alvarez de Andrés 2006). Genuer et al. (2008) specifically addressed this issue with simulation studies and concluded that the magnitude of the importance for a predictor steadily decreases when more variables highly correlated with the predictor are included in the data set. We also performed a simulation to examine this issue. We began with the four selected genes. Then, we identified the genes whose correlations with any of the four selected genes are at least 0.4. Those correlated genes are divided randomly in five sets of about same size. Finally, we added one, two, . . . , and five sets of them sequentially together with the four selected

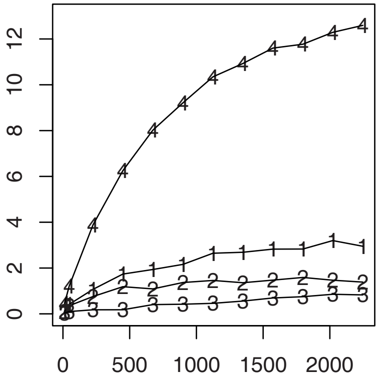

FIGURE 6.1. The dependence of the permutation importance on the choice of q. The x-axis is q and the y-axis the importance score. Each curve is for one of the four selected genes.

genes as the predictors. Figure 6.2 is consistent with the result of Genuer et al. (2008). We can see that the rankings of the predictors are preserved.

Furthermore, let us examine the impact of the correlation from a different point of view. We again began with the four selected genes and then included genes that are correlated with any of the correlated gene at least 0.6, 0.4, and 0.2. We see from Figure 6.3 that the magnitude of the importance for a gene increases as we restrict the correlation to a higher level.

It is reasonable to say that although variable importance is an important concept in random forests, we need to be cautious in the interpretation. In practice, the ranking is more relevant than the magnitude.

6.3.4 Maximum Conditional Importance

To overcome some of the issues raised above, Wang et al. (2010) introduced a maximal conditional chi-square (MCC) importance by taking the maximum chi-square statistic resulting from all splits in the forest that use the same predictor. Through simulation studies, Wang et al. (2010) found that MCC can distinguish causal predictors from noise. In addition, they compared the specificity (true negative probability) and sensitivity (true positive probability) of the importance indices introduced above using various genetic models. All indices have high specificity, i.e., screening out SNPs that are not associated with an underlying trait. However, MCC has the highest sensitivity in identifying the causal SNPs. Another use of MCC is to assess interactions. For example, consider the interaction between two predictors x_i and x_j. For x_i, suppose its MCC is reached in node t_i of a tree within a forest. Whenever x_j splits an ancestor of node t_i, we count one and otherwise zero. The final frequency, f, can give us a measure of interaction between x_i and x_j, and through the replication of the forest construction

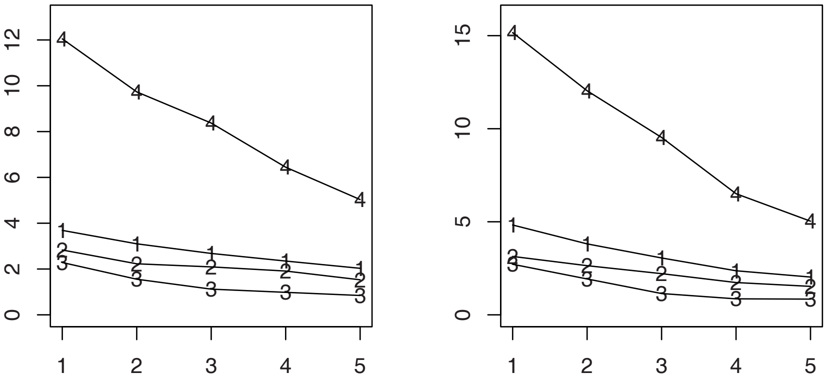

FIGURE 6.2. The dependence of the permutation importance on the number of correlated predictors. The x-axis is the number of correlated sets of genes and the y-axis the importance score. Each curve is labeled with the gene number. The forest size is set at 1000. q equals the square root of the forest size for the left panel and 8 for the right panel.

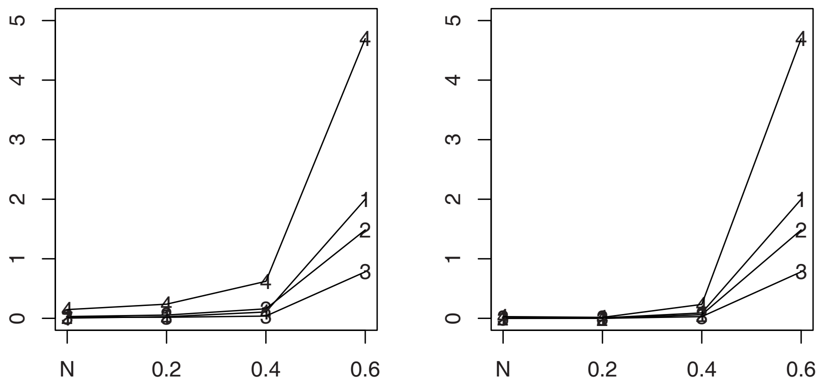

FIGURE 6.3. The dependence of the permutation importance on the correlation among the predictors. The x-axis is the level of correlation and the y-axis the importance score. Each curve is labeled with the gene number. The forest size is set at 1000. q equals the square root of the forest size for the left panel and 8 for the right panel.

we can estimate the frequency and its precision. As an illustration, in Figure 6.4 we present the heat map from the following simulation carried out by Wang et al. (2010).

They generated 100 predictors independently, each of them is the sum of two i.i.d. binary variables (0 or 1). This is to mimic genotypes derived from SNPs in genetic studies. For the first 16 predictors, the underlying binary random variable has the success probability of 0.282. For the remaining 84, they draw a random number between 0.01 and 0.99 as the success probability of the underlying binary random variable. The first 16 predictors will be used as the risk variables in our simulation and the remaining 84 the noise variables. The outcome variable is generated as follows. The 16 risk variables are divided equally into four groups, and without loss of generality, say sequentially. Once these 16 risk variables are generated, we calculate the following probability on the basis of which the response variable is generated:

$$w = 1 - \Pi(1 - \Pi q_k)$$

where the first product is with respect to the four groups, the second product is with respect to the first predictors inside each group, and $q_0 = 1.2 \times 10^{-8}$, $q_1 = 0.79$, and $q_2 = 1$. The subscript k equals the randomly generated value of the respective predictor. For example, if $x_1 = 1$, then $k = 1$ and we use q_1, i.e., 0.79 for the first predictor. The response variable takes the value of 1 with the probability w and 0 otherwise.

Wang et al. (2010) used the foregoing procedure to generate the first 200 possible controls (the response variable equals 0) and the first 200 possible cases (the response variable equals 1). This completes the generation of one data set, and a random forest can be built. Finally, they replicated the entire process 1000 times.

We can see from Figure 6.4 that the interactions within the 4-SNP groups are present and the interactions across the 4-SNP groups are absent. This figure seems to suggest that MCC can be utilized as a mechanism to detect interactions among predictors.

Lastly, to compare MCC with the permutation importance, let us examine the impact of including correlated predictors on MCC. In the same simulation as that for Figure 6.5, we also obtained the result for MCC as presented in Figure 6.5. Clearly, the inclusion of correlated genes has little impact on MCC.

6.4 Random Forests for Predictors with Uncertainties

In general, we base our analysis on predictors that are observed with certainty, or we assume so. However, this is not always the case. For example, to identify genetic variants for complex diseases, haplotypes are sometimes

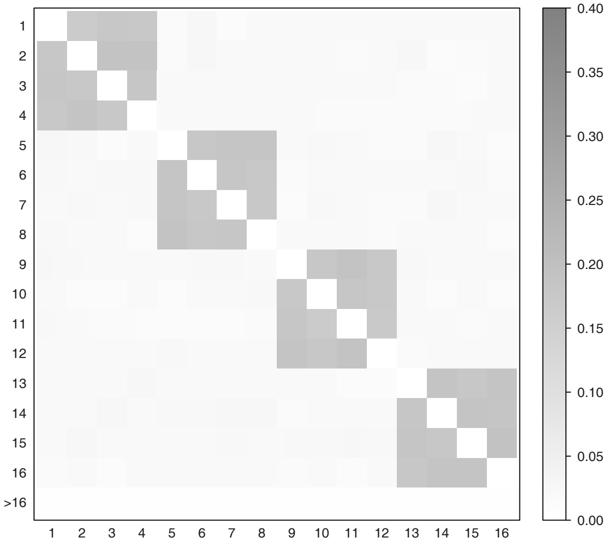

FIGURE 6.4. Interaction heat map. The x-axis is the sequence number of the primary predictor and the y-axis the sequence number of the potential interacting predictor. The intensity expresses the frequency when the potential interacting predictor precedes the primary predictor in a forest.

the predictors. A haplotype consists of alleles in multiple loci that are transmitted together on the same chromosome. In genomewide association studies, a haplotype is a combination of single nucleotide polymorphisms (SNPs) on a chromatid. The current technologies are capable of genotyping the SNPs with a great level of confidence, but not so for haplotypes, which have to be statistically inferred from the SNPs (see, e.g., Lin et al. 2002). As a result, haplotypes are available in frequencies. This issue also arises from other studies. For example, race is a predictor in almost all epidemiological studies. Even though it may be recorded as “White,” “Black,” etc., some subjects really are half white and half black or in other proportions. In the following, we describe the use of the random forest idea proposed by Chen et al. (2007) to address these uncertainties in the predictors.

For clarity, we assume x_1 is the only categorical variable with uncertainties, and it has K possible levels. For the i-th subject, $x_{i1} = k$ with a probability p_{ik} ($\sum_{k=1}^{K} p_{ik} = 1$). In a typical random forest, the “working” data set is a bootstrap sample of the original data set. Here, a “working” data set is generated according to the frequencies of x_1 while keeping the other variables intact. Thus, the data set would be $\{z_{i1}, x_{i2}, \ldots, x_{ip}, y_i\}_{i=1}^{n}$, where z_{i1} is randomly chosen from $1, \ldots, K$, according to the probabilities $(p_{i1}, \ldots, p_{iK})$. Once the data set is generated, the rest can be carried out in the same way as for a typical random forest. The procedure is similar if

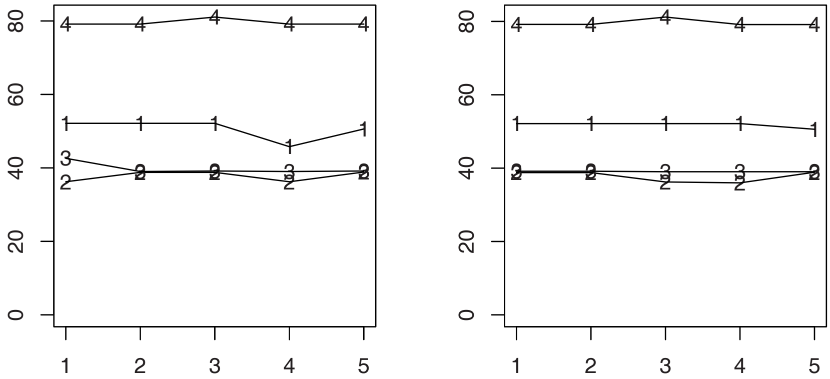

FIGURE 6.5. The dependence of the MCC on the number of correlated predictors. The x-axis is the number of correlated sets of genes and the y-axis the importance score. Each curve is labeled with the gene number. The forest size is set at 1000. q equals the square root of the forest size for the left panel and 8 for the right panel.

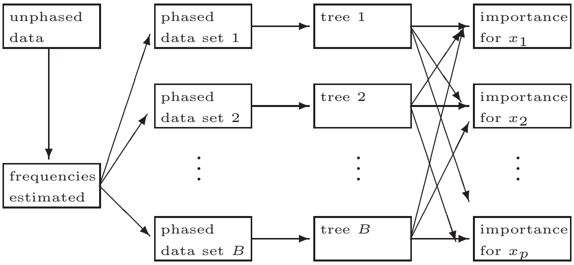

FIGURE 6.6. A schematic diagram to construct a forest for predictors with uncertainties. Predictors $x_1, x_2, \dots, x_p$ are not directly observed, and hence the raw data are referred to as "unphased data." The frequencies of the predictors can be estimated, and these frequencies are used to generate "phased data" in which the values of the predictors are drawn according to the distribution of the predictors. One tree is built for each phased data set. Finally, the importance score for each predictor is computed in the forest.

there are additional predictors with uncertainties, and in fact, this is the case for haplotype-based genetic analysis. We refer to Chen et al. (2007) for details. Figure 6.6 illustrates this process, and a computer program `HapForest` is available from `http://c2s2.yale.edu/software`.

A caveat with the tree- and forest-based method is that it is not feasible to perform theoretically based statistical inference such as the computation of statistical significance and confidence interval. For hypothesis testing, a general, while computationally intensive, approach is to generate data under the null hypothesis and examine the distribution of the critical statistics using the replicated permutation samples. For example, to assess the significance of association between a haplotype and a disease, the null distribution for an importance index can be empirically estimated by randomly permuting the disease status in the raw data and then going through the process in Figure 6.6 to produce one set of importance indices for all haplotypes under the null hypothesis. Repeating this process can estimate empirically the null distribution for all haplotypes.

Chen et al. (2007) and Wang et al. (2009) applied this method to a genetic data set on age-related macular degeneration (AMD), which is a leading cause of vision loss in the elderly. Using a genomewide significance level of 0.05, they confirmed one well-known haplotype, ACTCCG (on chromosome 1), and revealed several novel haplotypes, TCTGGACGACA (on chromosome 7), GATAGT (on chromosome 5), and TCTTACGTAGA (on chromosome 12). Using permutation, these novel haplotypes were associated with AMD beyond chance by a genomewide 5% significance level. The haplotype on chromosome 1 is in the gene called complement factor H (Klein et al. 2005), the one on chromosome 7 is located in the Bardet–

Biedl syndrome 9 gene, the one on chromosome 5 is in the region of the Sarcoglycan delta, and the one on chromosome 12 is in the Ankyrin repeat and sterile alpha motif domain containing 1B (Wang et al. 2009).

6.5 Random Forests with Weighted Feature Selection

The second step in the construction of a random forest is to select a subset of the predictors (features) to split a node. By this random selection, all features are treated with the same chance of being selected. This could be problematic when the number of available predictors is huge such as millions of SNPs in a genomewide association (GWA) study. It would take a large number of trees to give those important predictors enough chances to be selected in the first place. Furthermore, in a GWA study, besides genotypes, there tend to be a few covariates such as demographic variables that must be considered. Consequently, there is a severe imbalance in the number of SNPs and the number of "environmental" covariates. The standard random forest procedure is not effective in identifying potentially important environmental variables, because they are simply overwhelmed by the number of SNPs.

A simple, seemingly effective approach is to perform a univariate test using each predictor, e.g., the allelic χ^2 statistic for each SNP. Then, instead of drawing a subset of the q variables with equal probability for all predictors, the sampling probability is refined as a monotonic function of the χ^2 value. This approach is similar to the enriched random forest approach (Amaratunga et al. 2008) proposed in gene expression analyses.

In a simulation study in which the number of risk-enhancing SNPs is relatively small, Chen et al. (unpublished data) confirmed that the typical random forest is ineffective, as expected (Genuer et al. 2008), in identifying the underlying SNPs and environmental factors. However, the weighted random forest of a similar size yielded a much superior performance in terms of the number of prediction errors and the power of uncovering the important predictors.

6.6 Deterministic Forests

If we examine individual trees in a forest, we tend to find trees with comparable structures that have similar classification performance when the number of features is large relative to the number of samples, particularly evident in the analysis of microarray data. This observation was the motivation behind Zhang et al. (2003) in which the authors proposed a forest with trees of similar structures and similar performance. This forest could

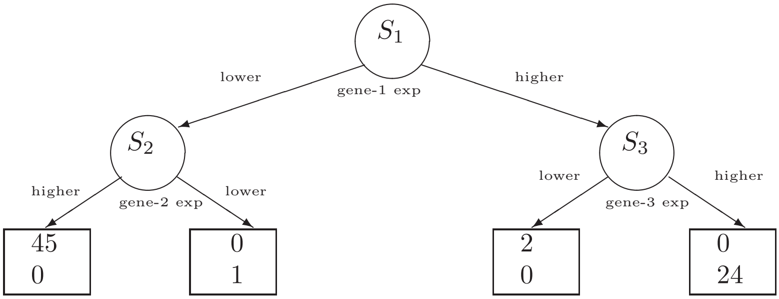

FIGURE 6.7. The frame of the deterministic forest. Each of S_1, S_2, S_3 is one of the top three splits of the corresponding node. Inside the terminal nodes are the results in one of the trees from the analysis of the leukemia data set.

provide more precise and biologically interpretable classification rules than any individual tree, and is reproducible—for this reason, such a forest is called deterministic forest.

Zhang et al. (2003) proposed and examined a simple way of forming a deterministic forest. They selected a prespecified number, say 20, of the top splits of the root node and a prespecified number, say 3, of the top splits of the two daughter nodes of the root node. This use of top nodes gives rise to a total of 180 possible (20 by 3 by 3) trees. When they applied this procedure to a leukemia data set (Golub et al. 1999), they noted that many of the trees are perfect or nearly perfect in classifying the subjects in the learning sample. For example, in Figure 6.7, S_1 is one of the top three splits of the root node, S_2 is one of the top three splits of the second node, and S_3 is one of the top three splits of the third node. Inside the terminal nodes are the results in one of the trees, illustrating a perfect classification in the learning sample.

An alternative, but computationally more challenging, approach is to prespecify a general structure such as “A” trees as the first step. An “A tree”’ (see, e.g., Figure 6.7) is a tree that is symmetric on the left and right. Then, we search for trees of a desired performance for inclusion in the forest. The performance of this procedure warrants further investigation.

6.7 A Note on Interaction

In classical statistical inference, the assessment of interaction requires prespecification of the interaction term. For example, in a linear model involving response Y, and two predictors x_1 and x_2, the product term x_1x_2

is the common representation of the interaction effect. In general, however, interactions refer to any deviation from the additive effects. The trees and forests provide a very flexible framework without prespecifying the interactions. Instead, we can assess interactions after trees and forests are grown. Furthermore, trees and forests can suggest existence of interactions even when the effect of those interactions may be too small to be detected individually. From a theoretical point of view, it would be important to establish a theoretical framework to assess interactions that are difficult to specify *a priori*.