Learning a Complex Metabolomic Dataset using Random Forests and Support Vector Machines

Young Truong
Department of Biostatistics
University of North Carolina
Chapel Hill, NC 27599-7420
truong@bios.unc.edu

Xiaodong Lin
Statistical and Applied Mathematical Sciences Institute
Research Triangle Park, NC 27709
linxd@samsi.info

Chris Beecher
Metabolon
Research Triangle Park, NC 27709
Beecher@metabolon.com

ABSTRACT

Metabolomics is the omics science of biochemistry. The associated data include the quantitative measurements of all small molecule metabolites in a biological sample. These datasets provide a window into dynamic biochemical networks and conjointly with other omic data, genes and proteins, have great potential to unravel complex human diseases. The dataset used in this study has 63 individuals, normal and diseased, and the diseased are drug treated or not, so there are three classes. The goal is to classify these individuals using the observed metabolite levels for 317 measured metabolites. There are a number of statistical challenges: non-normal data, the number of samples is less than the number of metabolites; there are missing data and the fact that data are missing is informative (assay values below detection limits can point to a specific class); also, there are high correlations among the metabolites. We investigate support vector machines (SVM), and random forest (RF), for outlier detection, variable selection and classification. We use the variables selected with RF in SVM and visa versa. The benefit of this study is insight into interplay of variable selection and classification methods. We link our selected predictors to the biochemistry of the disease.

Categories and Subject Descriptors: H.2.8 [Database Management]: Database applications - Data Mining; I.2.6 [Artificial Intelligence]: Learning

General Terms: Measurement

Keywords: Metabolomics, Random Forest, Support Vector Machines, Missing Data

1. INTRODUCTION

Complex data sets are becoming available in the post-genomic era; there is an increasing interest in the analysis of genetic information (genomics), and their transcription (transcriptomics) and subsequent translation into protein (proteomics). The recently emerging science of metabolomics completes the primary omics ladder of DNA, RNA, proteins and metabolites. As such, it provides another window into cellular function. Metabolomics is the biochemical profiling of all small molecules, biochemicals or metabolites, in an organism. Even though the dynamic nature of metabolites makes them difficult to measure, recent advances in technology allows robust quantification of the concentrations of hundreds to thousands of metabolites from a biological sample [11]. This new omics science offers potential insight into the dynamic interactions in metabolic pathways and an unambiguous representation of cellular physiology [2]. Furthermore, patterns of metabolites can be used to identify biomarkers of specific disease, understand pathological development, and propose targets for drug intervention. We might even track metabolites returning to normal under treatment.

Data obtained through a metabolomic experiment posses a number of challenges to statistical modeling. The number of metabolites measured ($p = 317$), usually in the hundreds is much larger than the number of biological samples ($n = 63$) typically available. Additionally, there may be severe distributional difficulties such as non-normal distributions, outliers (unusual data values), missing values, and high correlations among metabolites. Common objectives are finding "patterns" in the data, in particular, clustering of the biological samples (rows) into groups with similar metabolic expression profiles; and clustering the metabolic concentrations (columns) into groups where the pattern of metabolic expression is similar in the samples.

Due to the challenges of modeling this dataset, many classification and clustering techniques may produce sub-optimal results. For example, most techniques are influenced by outliers and can not accommodate missing values. A disadvantage of hierarchical clustering techniques is that the dendrograms produce orderings of rows (biological samples) that are not unique [8]. To address these issues, we previously used a method called robust singular value decomposition (rSVD) [8] (see also [6]) to detect the cluster. This method was compromised by another method based on recursive partitioning (RP) [1]. RP uses information on the classification of the objects whereas rSVD uses only the information on the metabolites, learning with supervision versus learning without supervision.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
KDD'04, August 22–25, 2004, Seattle, Washington, USA.
Copyright 2004 ACM 1-58113-888-1/04/0008 ...$5.00.

In this paper, we explore two more techniques on this metabolomic dataset (see Section 2). The dataset contains biological samples in which there are three different groups: diseased on medication, diseased not on medication and healthy individuals. One of the main scientific interests is to identify other groups within these major groups based on the metabolites data. Even though the samples were analyzed using a detector that is better than conventional mRNA, the dataset contains blanks where the metabolic concentrations were below detection limits. Subject to these conditions, the goal of this study is to use support vector machines and random forests in an exploratory manner to identify metabolites that classify samples into consistent groups that correspond to their known biological classification by experts. Moreover, it is of scientific interest to examine which or how many of these metabolites are important in classifying these cases.

2. DATA

In this paper we use a data set generated from blood sample of people with and without a particular disease. We know from the medical diagnosis which people have the disease and which do not. There is evidence that some of the people with the disease self-medicate so one or more of the blood metabolites could identify those people. The data set was collected to test the analytical equipment and as a proof of concept to support additional studies. The concept is that a sample of blood will be taken from a control and a target population. The separation and mass spec process should give a profile of metabolic products that is distinct for the target population. If this can be done, then there is an opportunity for better diagnosis. There is also the possibility of dynamically tracking either disease progression or how treatment modifies metabolites. Just as in genomic studies, there will be a need to track back to annotated data bases. The big advantage of mass spec is the very high precision it allows for the identification of metabolic products. There are many statistical challenges with data of this sort.

The data was acquired by an established global electrical chemical array profiling method. It is expected that addition of massspec capabilities for assaying equivalent numbers of different classes of compounds will significantly enhance the categorical separation and mechanistic insights in such data sets [7]. The data contains the metabolic profiles of 63 samples: 32 healthy subjects, 22 subjects diagnosed with a specific disease state and not taking medication, and 9 subject diagnosed with the disease who are taking medication. Each profile contains the intensity (abundance) levels of 317 compounds. Blanks in the data indicate the metabolic concentrations were below detection limits.

3. METHODS

3.1 Random Forest (RF) Methodology

As its name suggests, a random forest is a collection of identitically distributed trees. Each tree is constructed via a tree classification algorithm such as CART [4]. A random forest is formed by using bootstrap samples from the learning set. That is, each tree is grown on a bootstrap replicate of the learning sample. The simplest random forest with random features is formed by selecting randomly, at each node, a small group of input variables to split on. The size of the group is fixed throughout the process of growing the forest. Each tree is grown by using the CART methodology without pruning. After the forest is formed, drop a case with input x into the forest for each tree to classify x. The forest chooses the class for x having the majority vote. Specifically, for each case, the proportion of votes for each class is recorded. For each member of a test set (with or without class labels), these proportions are also computed. They contain useful information about the case. The margin of a case is the proportion of votes for the true class minus the maximum proportion of votes for the other classes. The size of the margin gives a measure of how confident the classification is. A very useful reference to RF is [3].

Since an individual tree is unpruned, the terminal nodes will contain only a small number of instances. Run all cases in the training set down the tree. If cases i and j both land in the same terminal node, increase the proximity between i and j by one. At the end of the run, the proximities are divided by twice the number of trees in the run and proximity between a case and itself set equal to one.

3.1.1 Features of RF

Features of RF can be highlighted as follows (see also [3]):

- It is an excellent classifier – comparable in accuracy to support vector machines.
- It generates an internal unbiased estimate of the generalization error as the forest building progresses.
- It has a method for balancing error in unbalanced class population data sets.
- Generated forests can be saved for future use on other data.
- It gives estimates of what variables are important in the classification and generates information about the relation between the variables and the classification.
- It computes proximities between pairs of cases that can be used in clustering, locating outliers, or by scaling, give useful views of the data.
- The above capabilities can be extended to unlabeled data, leading to unsupervised clustering, data views and outlier detection. The missing value replacement algorithm also extends to unlabeled data.

3.1.2 Random Forests Tools

The design of random forests is to give the user a good deal of information about the data besides an accurate prediction. The information includes:

- *Test set error rate.* In random forests, there is no need for cross-validation or a separate test set to get an unbiased estimate of the test set error. It is obtained internally, during the run, as follows: Each tree is constructed using a different bootstrap sample from the original data. About one-third of the cases are left out of the bootstrap sample and not used in the construction of the kth tree. These are called out-of-bag (OOB). These OOB cases turn out to be useful. Put each OOB case in the construction of the k-th tree down the k-th tree to get a classification. In this way, a test set classification is achieved for each case in about

one-third of the trees. Let the final test set classification of the forest be the class having the most popular votes. Comparing this classification with the class label present in the data gives an estimate of the test set error. Moreover, if an individual variable in the OOB cases is randomly permuted before being put back into the tree, then the decrease in the estimated margins (see below) is an indication of how important that variable is.

- *Variable importance.* There are a number of different ways of measuring variable importance. For example, to estimate the importance of the mth variable, in the left out cases for the kth tree, randomly permute all values of the mth variable. Put these new variable values down the tree and get classifications. For the i-th case in the data, its margin at the end of a run is the proportion of votes for its true class minus the maximum of the proportion of votes for each of the other classes. The measure of importance of the mth variable is the average lowering of the margin across all cases when the mth variable is randomly permuted.

- *Intrinsic proximities between cases.* After each tree is grown, the entire training set is run down the tree. If two cases i and j turn up in the same terminal node, then their proximity measure $\text{prox}(i,j)$ is increased by one. At the end of the forest construction, these are normalized by dividing by the number of trees. The proximities give an intrinsic measure of similarities between cases. They are used in replacing missing values by estimating each missing value by a proximity weighted sum over the non-missing values. Then using the replaced values, run RF again to get new proximities and repeat. The proximities are also used to locate outliers–using the definition of an outlier as a case that only has weak similarities to the other cases. The most useful property is that $1 - \text{prox}(i,j)$ form Euclidean distances in a high dimensional space. They can be projected down onto a low dimensional space using metric scaling. This gives informative views of the data.

- *Scaling coordinates based on the proximities.* The proximities between cases i and j form a matrix with entries $\text{prox}(i,j)$. From their definition, it is easy to show that this matrix is symmetric, positive definite and bounded above by 1, with the diagonal elements equal to 1. It follows that the values $1 - \text{prox}(i,j)$ are squared distances in a Euclidean space of dimension not greater than the number of cases. Let $\text{prox}(i,\cdot)$ be the average of $\text{prox}(i,j), j = 1,\ldots,n$, and $\text{prox}(\cdot,\cdot)$ the average over both coordinates. Then the matrix: $\text{cv}(i,j) = .5[\text{prox}(i,j) - \text{prox}(i,\cdot) - \text{prox}(\cdot,j) + \text{prox}(\cdot,\cdot)]$ is the matrix of inner products of the distances and is also positive definite symmetric. In metric scaling, the idea is to approximate the vectors $x(i)$ by the first few scaling coordinates. This is done in random forests by extracting the number of the largest eigenvalues and corresponding eigenvectors of the cv matrix. The two dimensional plots of the ith scaling coordinate vs. the jth often gives useful information about the data. The most useful is usually the graph of the second vs. the first.

- *Outlier detection.* Outliers are being treated as cases having small proximities to all other cases. Since the data in some classes is more spread out than others, outlyingness is defined only with respect to other data in the same class as the given case. For each case i, a measure of outlyingness is computed according to $\text{out}(i) = [\sum\{\text{prox}(i,j)\}^2]^{-1}$, where the sum is over all j in the same class as case i. This quantity will be large if the proximities $\text{prox}(i,j)$ from i to the other cases j in the same class are generally small. Next, normalize $\text{out}(i), i = 1,\ldots,n$ by subtracting the median from each $\text{out}(i)$, divided by the the mean absolute deviation from the median. The values less than zero are set to zero. Generally, a value above 10 is reason to suspect the case is profound enough of being outlying.

- *Variable Effect on Classes.* For each case i, the proportion of votes that i gets in the forest for class k is $q(k,i)$. The higher that $q(k,i)$ is for a class k, the more "confident" is its classification. Ordinarily, a case is classified into the class k that maximizes $q(k,i)$. The methods of distributing classification errors and increasing coverage of small classes are based on finding suitable thresholds of the $q(k,i)$. To see the differences in variables that drive the classification, RF extracts for each class k, those cases having the highest values of $q(k,i)$. These are then contrasted with each other to see the values of the variables that are discriminating between the classes.

3.2 Support Vector Machine (SVM)

Support vector machine (SVM) is a supervised learning technique for classification and regression. The key to the success of SVM is the kernel function which maps the data from the original space into a high dimensional (possibly infinite dimensional) feature space. By constructing a linear boundary in the feature space, the SVM produces nonlinear boundaries in the original space. When the kernel function is linear, the resulting SVM is a maximum-margin hyperplane. Given a training sample, a maximum-margin hyperplane splits a given training sample in such a way that the distance from the closest cases (support vectors) to the hyperplane is maximized. Typically, the number of support vectors is much less than the number of the training sample. Nonlinear kernel functions such as the polynomial kernel and the Gaussian (radial basis function) kernel are also commonly used in SVM. The computational complexity of the SVM depends on the training sample, thus it avoids the traditional problem of "Curse of dimensionality". One of the most important advantage for the SVM is that it guarantees generalization to certain extend. Namely, the decision rules reflect the regularities of the training data rather than the incapabilities of the learning machine. Because of the many nice properties of SVM, it has been widely applied to virtually every research field. More detailed discussion of SVM and kernel methods can be found in [9].

4. EXAMPLE

4.1 Random Forest

In our analysis, we use RandomForest 4.0-7 for R and the soon to be released version of the fortran code developed by Breiman and Cutler.

Table 1: Confusion matrix and the error rate. The OOB estimate of error rate: 12.7%

	predicted			
observed	blue	green	yellow	class.error
blue	21	0	1	0.045
green	0	3	6	0.667
yellow	1	0	31	0.031

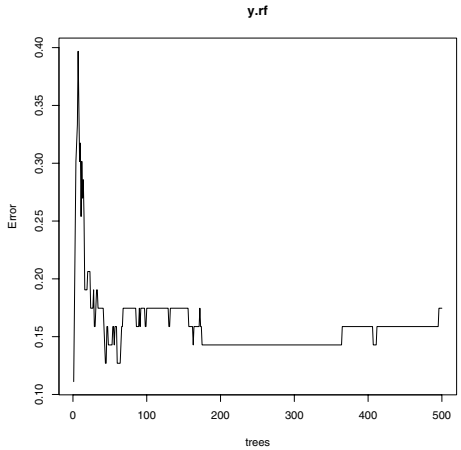

Figure 1: Error Rate.

The current version of random forests contains some modifications and major additions to earlier versions. The additions are:

- Replacement of missing values.
- A method to balance error in class unbalanced data sets.
- Data that can be used to see how the variables relate to the classification.
- Efficient handling of categorical data with a large number of values.

To highlight these features, we grow a forest using 500 trees. The size of the selected group of input variables has a default value of square root of the total number of inputs ($F = [\sqrt{317}] = 17$).

Test set error rate

The bootstrap training sample on which each tree is grown has about 2/3 of the cases. The remaining 1/3 cases are the out-of-bag (OOB) samples. By putting back into the associated tree they form a test sample that gives the ongoing OOB estimate of test set error. If an individual variable in the OOB cases is randomly permuted before being put back into the tree, then the decrease in the estimated margins is an indication of how important that variable is. In building the forest for the metabolite data, the number of variables tried at each split is 17. The OOB estimate of error rate is 12.7%. The result is summarized in Table 1. The error rate is depicted in Figure 1.

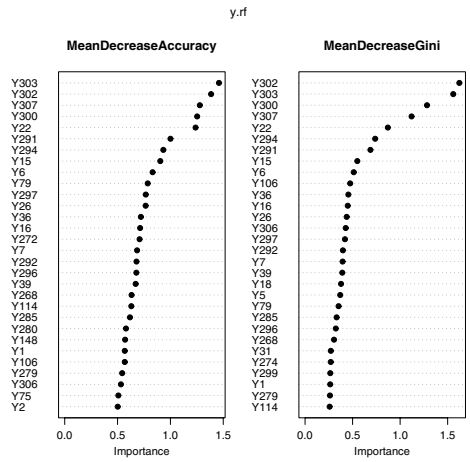

Figure 2: Variable importance.

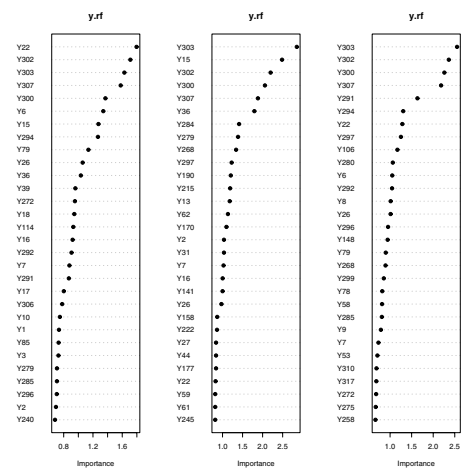

Figure 3: Variable importance according to group (Healthy=left, Diseased-treated=middle, Diseased-untreated=right).

Variable Importance

What was also interesting from a scientific viewpoint was an estimate of the importance of each of the 317 metabolites. From the result of random forest, Figure 2 shows the mean decrease accuracy and the mean decrease Gini (as an indication of node impurity) for all three classes, sorted from the most decrease to the least. Figure 3 contains plots of the mean decrease Gini in each class, sorted according to the magnitude of decrease. To get further insight about these variables or metabolites, we used a tool (still being developed) by putting the three classes together, and distinguish them by color as in Figure 4. Useful information on variable importance can also be highlighted by blurring the less important ones. This indicates a significant step in feature selection without losing the accuracy in predicting the class membership. These figures suggest a set of less than 15 (out of 317) metabolites would provide a very reasonable classification rule.

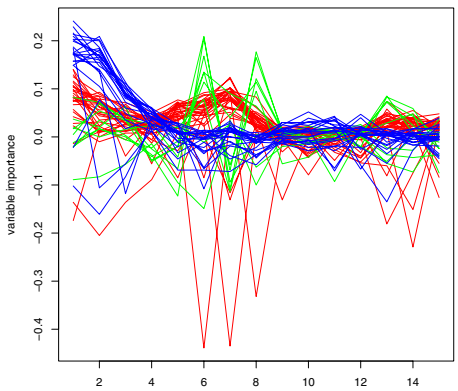

Figure 4: Alternative plot of variable importance. Important variable are number 6, 7, and 8.

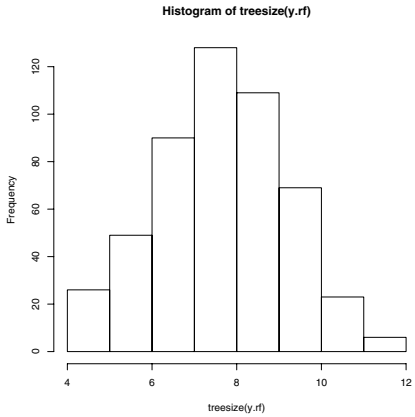

Figure 5: Distribution of tree sizes in RF.

Size of Trees

RF allows the user to see the number of variables employed in building each tree as well as the size of the tree in the forest. A histogram of the size of the tree in our forest is given in Figure 5. It is observed that this forest has many small trees with a mean about 8. The range of the tree sizes is 4 to 12.

Outlier Location

From a study conducted based on the technique rSVD, it is known that there are outliers in this data set [10]. RF defines outliers as cases having small proximities to all other cases. Figure 6 is a graph of outlyingness for the metabolite data using the normalized measure of outlyingness described above. There are at least three large values, which are profound enough to be considered as outliers. The finding here is consistent with method based on rSVD in [10].

MDS Plot and subgroup detection

According to the clinical information, we suspect that there are two normal subgroups and they were not easily detected

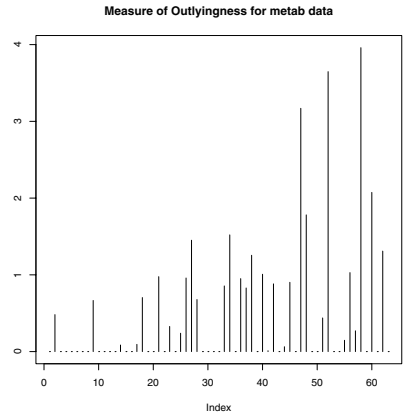

Figure 6: Outliers identification.

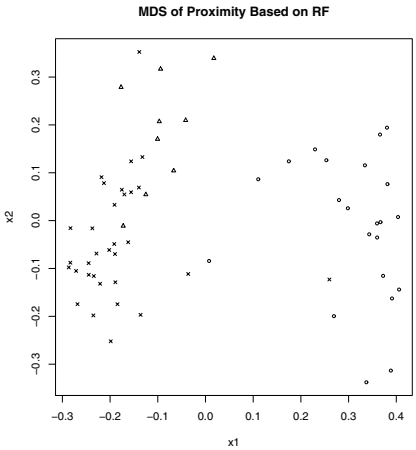

Figure 7: Multidimensional scaling plot. Diseased-untreated=circle, Diseased-treated=triangle, Healthy=cross.

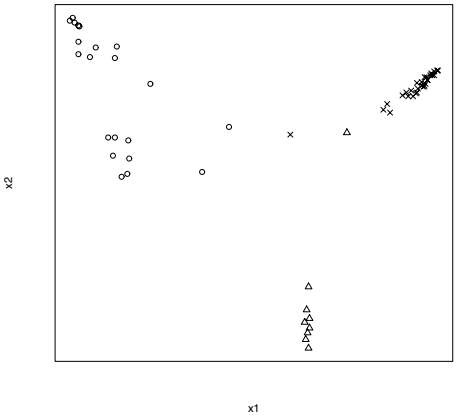

Figure 8: MDS plot combined with orthogonal projections. Diseased-untreated=circle, Diseased-treated=triangle, Healthy=cross. There are two healthy subgroups.

Table 2: Confusion matrix and the error rate for imputed data. The OOB estimate of error rate: 14.0%.

	predicted			
observed	blue	green	yellow	class.error
blue	20	0	2	0.090
green	2	1	6	0.889
yellow	1	0	32	0.031

Table 3: Comparison of different SVM techniques on the metabolonmic data set.

	Linear	Polynomial	Gaussian	LPSVM
class.error	19.0%	23.8%	28.6%	17.4%
Features	N/A	N/A	N/A	32

using methods such recursive partitioning or rSVD reported previously [10]. Figure 7 shows the first two multidimensional scaling coordinates obtained by using the proximity matrix.

RF is currently under a major revision, we used a graphical interface developed in JAVA to view the MDS plot through orthogonal projections or rotations. Figure 8 shows a rotated multidimensional scaling plot. The two normal subgroups are clearly displayed.

Missing values

In a previous study[10], we applied rSVD and recursive partitioning methods to classify the outcomes, but did not look into the issue of imputing the missing values. In the current paper, RF was used to impute the missing value values. The test set error rate based on the imputed data is summarized in Table 2. The error rate is higher than that given in Table 1, which was obtained by replacing the missing values by zeros. The missing values occurred when the machine could not detect the intensity level of the metabolite, and so it may not be a good idea to treat below-detection values as missing at random.

4.2 Support Vector Machines

We perform several SVM analysis on the metabolomic data set. For the classical SVM, we used the OSU SVM package available at http://www.ece.osu.edu/ maj/osu_svm/. We also applied the LPSVM [5] for the SVM with feature selection. In this study, we divide the samples into the disease and non-disease groups and perform binary classifications. For the classical SVM, we apply both the linear kernel and the nonlinear kernel (second degree polynomial and Gaussian). In all these studies, leave-1-out cross validation is used to choose the optimal tuning parameters.

Table 3 describes the error rates for these methods. Apparently, the error rate obtained from the LPSVM: 0.174 is the best among all the five methods. Moreover, this superior performance is achieved by using only 32 out of the 317 metabolites.

5. CONCLUSIONS

In this study, we observed that the SVM has the tendency to overfit the data. That is, it will be difficult to apply the tool to select important metabolites to classify cases. It also has a higher error rate. On the other hand, random forests do not overfit and the method provides many useful tools for detecting hidden structures in the data. We have used the outlyingness measure to search for the profound observations, and the newly implemented graphical procedure to detect the hidden subgroups in the healthy cohort. Furthermore, the variable importance feature has been effective in choosing the relevant metabolites, this is important for understanding the nature and the biochemistry of the disease.

6. ACKNOWLEDGMENTS

The authors gratefully acknowledge Dr.s Bruce Kristal, Steve Rozen, Rima Kaddurah-Daouk, Wayne Matsen, Misha Bogdanov, Flint Beale, and Dr.s Merit Cudkowicz and Robert Brown at the Massachussetts General Hospital for their contributions to the organization of the original dataset and for access to the clinical samples. We also wish to thank David Banks and Leanna House for many helpful discussions and ideas.

7. ADDITIONAL AUTHORS

Additional authors: Adele Cutler (Department of Mathematics, Utah State University, Logan UT 84332-3900, email: `adele@math.usu.edu`) and S. Stanley Young (National Institute of Statistical Sciences, Research Triangle Park, NC 27709, email: `genetree@bellsouth.net`).

8. REFERENCES

[1] *FIRMPlus*. Golden Helix Inc, Bozeman, MT, USA.

[2] C. Beecher. Metabolomics: The newest of the 'omics' sciences. *Innovations of Pharmaceutical Technology*, pages 57–64, 1995.

[3] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[4] L. Breiman, J. Friedman, R. Olshen, and C. J. Stone. *Classification and Regression Trees*. Wadsworth, Belmont, CA, 1984.

[5] G. Fung and O. L. Mangasarian. A feature selection newton method for support vector machine classification. *Computational Optimization and Applications*, 28:185–202, 2004.

[6] D. M. Hawkins, L. Liu, and S. Young. Robust singular value decomposition. *National Institute of Statistical Science Technical Report*, 122, 2001.

[7] B. S. Kristal, K. Vigneau-Callahan, and W. R. Matson. Simultaneous analysis of multiple redox-active metabolites from biological matrices. *Methods Mol Biol*, (186):185–94, 2002.

[8] L. Liu, D. Hawkins, S. Ghosh, and S. Young. Robust singular value decomposition analysis of microarray data. *Proceedings of The National Academy of Sciences*, 23:13167–13172, 2003.

[9] B. Scholkopf and A. Smola. *Learning with Kernels*. MIT Press, Cambridge, MA, USA, 2002.

[10] S. J. Simmons, X. Lin, C. Beecher, Y. Truong, and S. S. Young. Active and passive learning to explore a complex metabolism data set. *IFCS*, to appear, 2004.

[11] M. Stitt and A. R. Fernie. From measurements of metabolites to metabolomics: an 'on the fly' perspective illustrated by recent studies of carbon-nitrogen interactions. *Current Opinion in Biotechnology*, 14:136–144, 2003.