WALD LECTURE II

LOOKING INSIDE THE BLACK BOX

Leo Breiman
UCB Statistics
leo@stat.berkeley.edu

ORIGIN OF BLACK BOXES

Statistics uses data to explore problems.

Think of the data as being generated by a black box .

A vector of input variables **x** (independent variables) go into one side.

Response variables **y** come out on the other side.

Inside the black box, nature functions to associate the input variables with the response variables, so the picture is like this:

All we see are a sample of data

$$(\mathbf{y}_n, \mathbf{x}_n)\, n = 1, \ldots, N)$$

From this, ststisticians want to draw conclusions about the mechanism operating inside the black box.

STOCHASTIC DATA MODELING

starts with assuming a stochastic data model for the inside of the black box.

A common data model is that data are generated by independent draws from:

response variables
f(predictor variables,random noise, parameters)

Parameters are estimated from the data and the model then used for information and / or prediction. The black box is filled in like this:

Model validation is yes-no using goodness-of-fit tests and residual examination

Inferring a mechanism for the black box is a highly risky and ambiguous venture.

Nature's mechanisms are generally complex and cannot summarized by a relatively simple stochastic model, even as a first approximation. The attraction: a deceptively simple picture of the inside.

FITTING THE DATA

An important principle

The better the model fits the data, the more sound the inferences about the black box are.

Goodness-of-fit tests and residual analysis are not reliable. They accept a multitude of badly fitting models.

Suppose there is a model $f(\mathbf{x})$ that outputs an estimate $\hat{\mathbf{y}}$ of the true $\mathbf{y}$ for each value of $\mathbf{x}$.

Then a measure of how well f fits the data is given by how close $\hat{\mathbf{y}}$ is to $\mathbf{y}$. This can be measured as follows: given an independent test set

$$(\mathbf{y'}_n, \mathbf{x'}_n)\ n = 1, \ldots, N')$$

and a loss function $L(\mathbf{y}, \hat{\mathbf{y}})$, define the estimated prediction error as

$$PE = av_{n.'} L(\mathbf{y}'_n, f(\mathbf{x}'_{n'}))$$

If there is no test set, use cross-validation to estimate PE.

The lower the PE, the better the fit to the data

RANDOM FORESTS

A random forest (RF) is a collection of tree predictors

$$f(\mathbf{x}, \mathbf{T}, \Theta_k), k = 1, 2, \ldots, K)$$

where the Θ_k *are i.i.d random vectors.*

In regression, the forest prediction is unweighted average over the forest: in classification, the unweighted plurality.

Unlike boosting, the LLN insures convergence as $k \to \infty$.

The key to accuracy is low correlation and bias. To keep bias low, trees are grown to maximum depth.

To keep correlation low, the current version uses this randomization.

1) Each tree is grown on a bootstrap sample of the training set.

2) A number m is specified much smaller than the total number of variables M. At each node, m variables are selected at random out of the M, and the split is the best split on these m variables.

(see Random Forests , Machine Learning(2001) 45 5-320)

RF AND OUT-OF-BAG (OOB)

In empirical tests, RF has proven to have low prediction error. On a variety of data sets, it is more accurate than Adaboost (see my paper)

It handles hundreds and thousands of input variables with no degeneration in accuracy

An important feature is that it carries along an internal test set estimate of the prediction error.

For every tree grown, about one-third of the cases are out-of-bag (out of the bootstrap sample). Abbreviated oob.

Put these oob cases down the corresponding tree and get response estimates for them.

For each case n, average or pluralize the response estimates over all time that n was oob to get a test set estimate $\hat{y}_n$ for y_n.

Averaging the loss over all n give the test set estimate of prediction error.

The only adjustable parameter in RF is m. The default value for m is $\sqrt{M}$. But RF is not sensitive to the value of m over a wide range.

HAVE WE PRODUCED ONLY A GOLEM?

With scientific data sets more is required than an accurate prediction, i.e. relevant information about the relation between the inputs and outputs and about the data--

looking inside the black box is necessary

Stochastic data modelers have criticized the machine learning efforts on the grounds that the accurate predictors constructed are so complex that it is nearly impossible to use them to get insights into the underlying structure of the data.

They are simply large bulky incoherent single purpose machines.

The contrary is true

Using RF we can get more reliable information about the inside of the black box than using any stochastic model.

But it is not in the form of simple equations.

RF & LOOKING INSIDE THE BLACK BOX

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

Much of this information comes from using the oob cases in the training set that have been left out of the bootstrapped training set.

The information includes:

i) *Variable importance measures*

ii) *Effects of variables on predictions*

iii) *Intrinsic proximities between cases*

iv) *Clustering*

v) *Scaling coordinates based on the proximities*

vi) *Outlier detection*

I will explain how these work and give applications, both for labeled and unlabeled data.

VARIABLE IMPORTANCE.

Because of the need to know which variables are important in the classification, RF has three different ways of looking at variable importance.

Sometimes influential variables are hard to spot--using these three measures provides more information.

Measure 1

To estimate the importance of the mth variable, in the oob cases for the kth tree, randomly permute all values of the mth variable

Put these altered oob **x-**values down the tree and get classifications.

Proceed as though computing a new internal error rate.

The amount by which this new error exceeds the original test set error is defined as the importance of the mth variable.

Measures 2 and 3

For the nth 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 2nd measure of importance of the mth variable is the average lowering of the margin across all cases when the mth variable is randomly permuted as in method 1.

The third measure is the count of how many margins are lowered minus the number of margins raised.

We illustrate the use of this information by some examples.

AN EXAMPLE--HEPATITIS DATA

Data: survival or non survival of 155 hepatitis patients with 19 covariates.

Analyzed by Diaconis and Efron in 1983 Scientific American.

The original Stanford Medical School analysis concluded that the important variables were numbers 6, 12, 14, 19.

Efron and Diaconis drew 500 bootstrap samples from the original data set and used a similar procedure, including logistic regression, to isolate the important variables in each bootstrapped data set.

Their conclusion , "Of the four variables originally selected not one was selected in more than 60 percent of the samples.

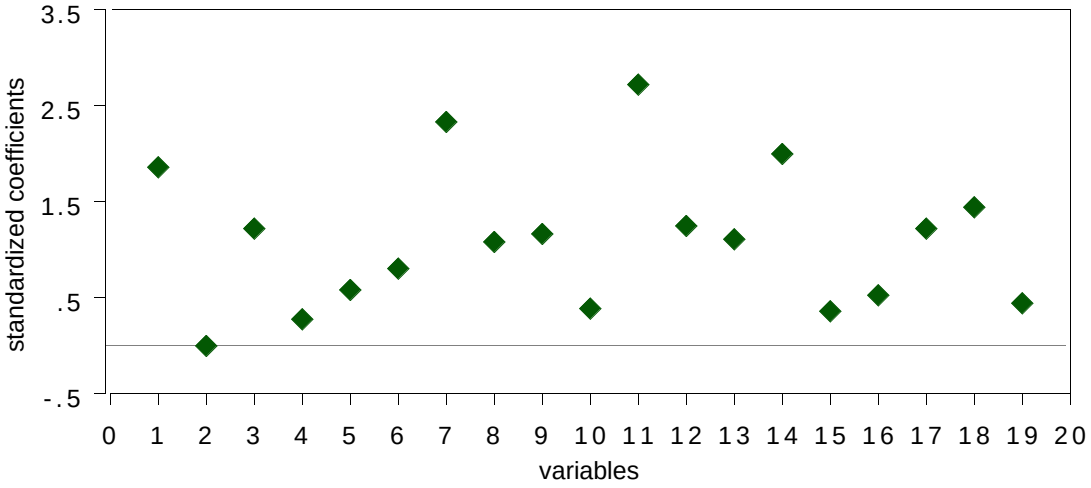
Hence the variables identified in the original analysis cannot be taken too seriously."

LOGISTIC REGRESSION ANALYSIS

Error rate for logistic regression is 17.4%.

Variables importance is based on absolute values of the coefficients of the variables divided by their standard deviations.

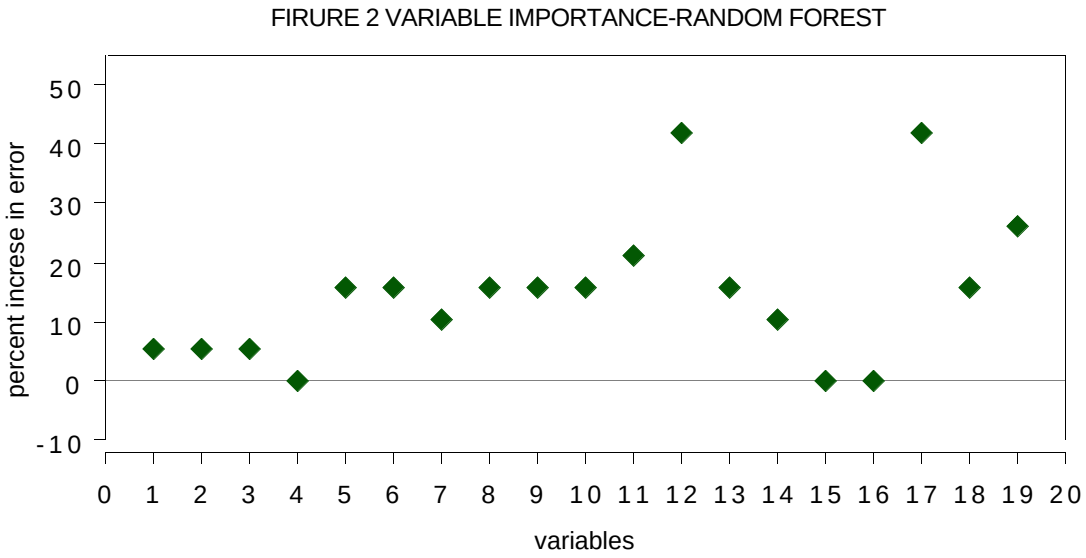
FIGURE 1 STANDARDIZED COEFFICIENTS-LOGISTIC REGRESSION



The conclusion is that variables 7 and 11 are the most important covariates. When logistic regression is run using only these two variables, the cross-validated error rate rises to 22.9% .

ANALYSIS USING RF

The error rate is 12.3%--30% reduction from the logistic regression error. Variable importances (measure 1) are graphed below:



Two variables are singled out--the 12th and the 17th The test set error rates running 12 and 17 alone were 14.3% each.

Running both together did no better. Virtually all of the predictive capability is provided by a single variable, either 12 or 17. (they are highly correlated)

REMARKS

There are 32 deaths and 123 survivors in the hepatitis data set. Calling everyone a survivor gives a baseline error rate of 20.6%.

Logistic regression lowers this to 17.4%. It is not extracting much useful information from the data, which may explain its inability to find the important variables.

Its weakness might have been unknown and the variable importances accepted at face value if its predictive accuracy is not evaluated.

The standard procedure when fitting data models such as logistic regression is to delete variables.

Diaconis and Efron (1983) state , "...statistical experience suggests that it is unwise to fit a model that depends on 19 variables with only 155 data points available."

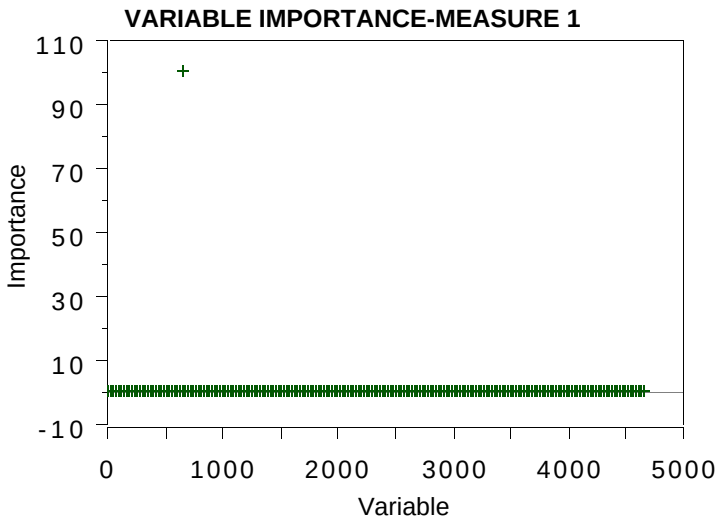
RF thrives on variables--the more the better. There is no need for variable selection ,On a sonar data set with 208 cases and 60 variables, the RF error rate is 14%. Logistic Regression has a 50% error rate.

MICROARRAY ANALYSIS

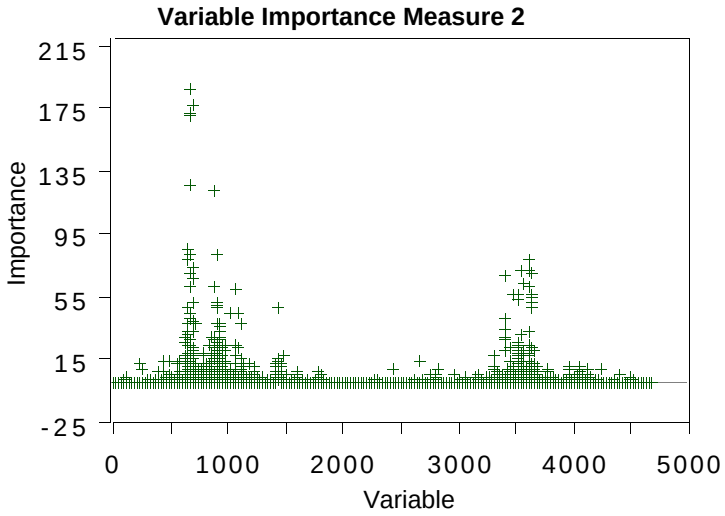
RF was run on a microarray lymphoma data set with three classes, sample size of 81 and 4682 variables (genes) without any variable selection. The error rate was low (1.2%).

What was also interesting from a scientific viewpoint was an estimate of the importance of each of the 4682 gene expressions.

RF was run and the measures of importance computed. Here are the results for the first measure of importance.



Next are the results for the second measure



The graphs show that measure 1 has the least sensitivity, showing only one significant variable.

Measure 2 has more, showing not only the activity around the gene singled out by measure 1 but also a secondary burst of activity higher up.

Measure 3 (not shown) has too much sensitivity, fingering too many variables.

EFFECTS OF VARIABLES ON PREDICTIONS

Besides knowing which variables are important, another piece of information needed is how the values of each variable effects the prediction.

Each time case n is oob it receives a vote for a class from its associated tree.

At the end of the run, there are available the proportions of the vote for each class and for each case. Call these the cpv's

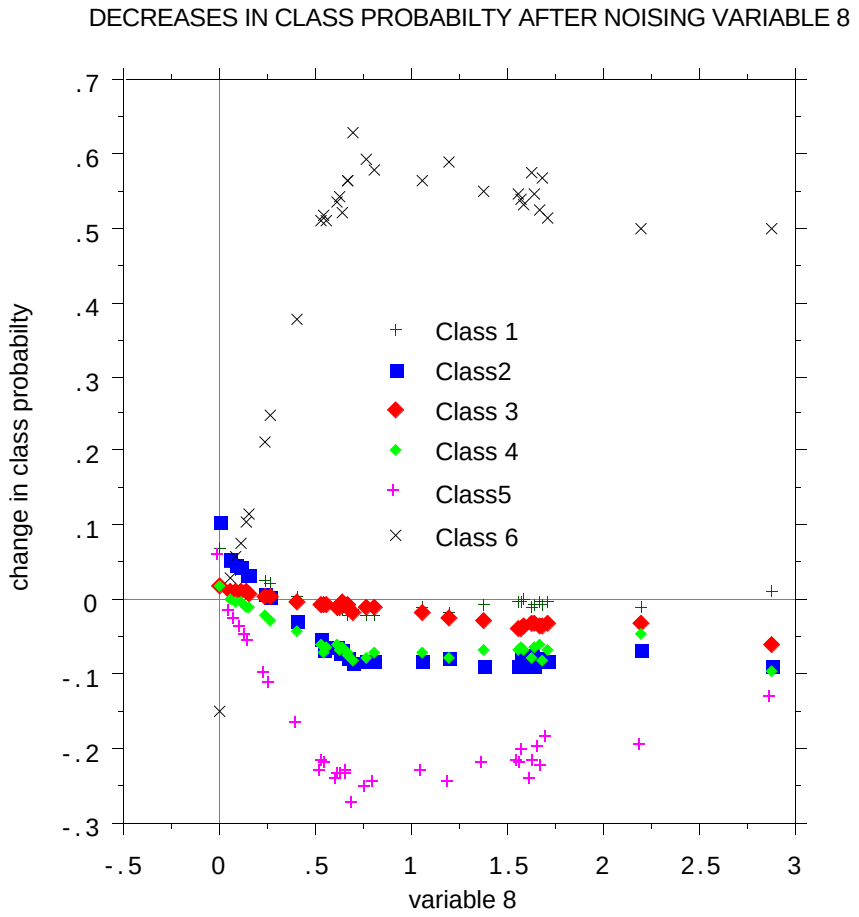
For each class and each variable m, compute the cpv for the jth minus the cpv with the mth variable noised.

Plot this against the values of the mth variable and do a smoothing of the curve.

To illustrate, we use the glass data set. It's six class with 214 samples and nine variables consisting of chemical proportions.

The figure below is a plot of the decreases in cpv's due to noising up the 8th variable in the glass data.

EFFECT OF THE 8TH VARIABLE



The sixth class cpv is significantly decreased, implying that the eighth variable is important for singling out this class.

The other class cpv's increase somewhat, implying that the other classes can be predicted more accurately with the eighth variable removed.

A PROXIMITY MEASURE AND CLUSTERING

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 case i and case 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 the number of trees in the run and proximity between a case and itself set equal to one.

This is an intrinsic proximity measure, inherent in the data and the RF algorithm.

To cluster-use the above proximity measures.

EXAMPLE-BUPA LIVER DISORDERS

This is a two-class biomedical data set consisting of six covariates, the last being alcohol consumption per day.

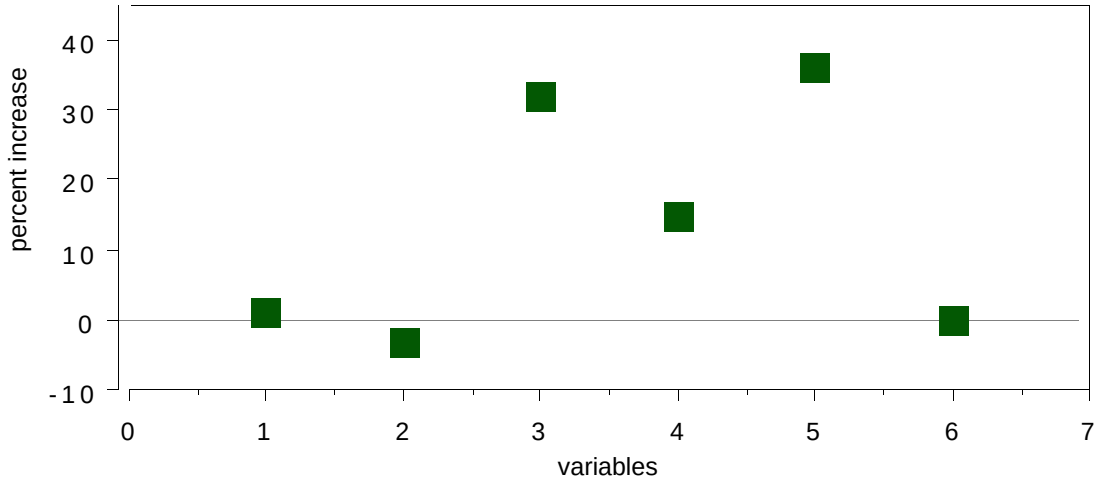
The first five attributes are the results of blood tests thought to be related to liver functioning. The 345 patients are classified into two classes by the severity of their liver disorders.

What can we learn about this data?

The misclassification error rate is 28% in a Random Forests run.

A) *Variable Importance (method 1)*

FIGURE 2 VARIABLE IMPORTANCE-BUPA LIVER



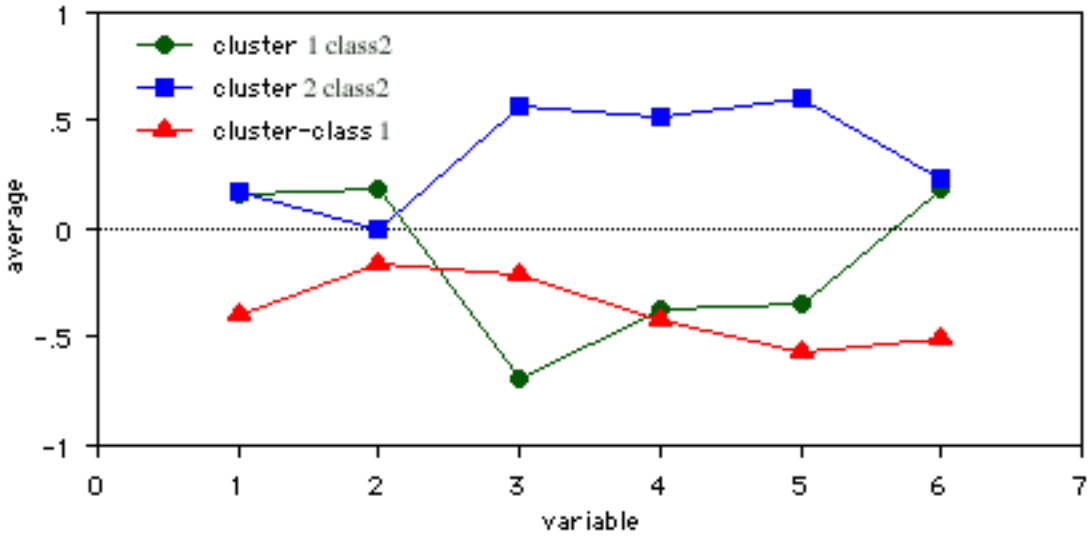
Blood tests 3 and 5 are the most important, followed by test 4.

B) *Clustering*

Using the proximity measure outputted by Random Forests to cluster, there are two class #2 clusters.

In each of these clusters, the average of each variable is computed and plotted:

Figure 3 Cluster Variable Averages



Something interesting emerges. The class two subjects consist of two distinct groups:

Those that have high scores on blood tests 3, 4, and 5 Those that have low scores on those tests.

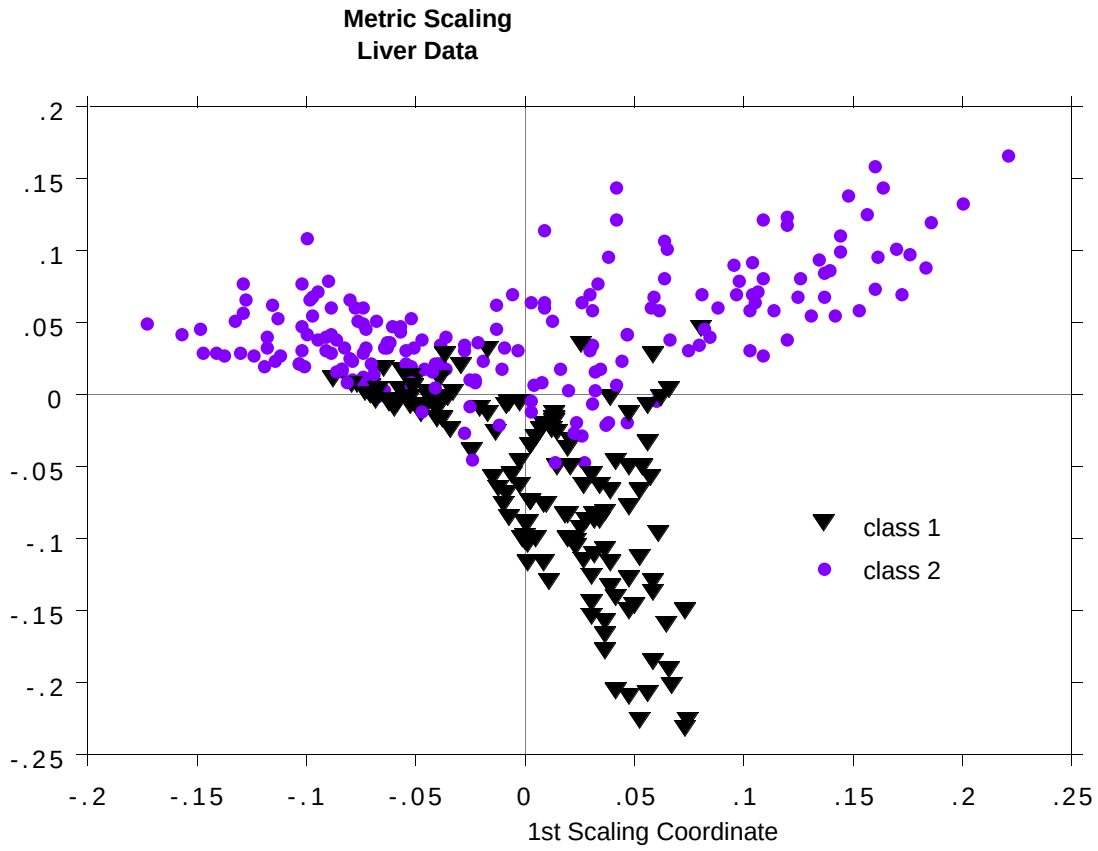
We will revisit this example below.

SCALING COORDINATES

The proximities between cases n and k form a matrix {prox(n,k)}. From their definition, it follows that the values 1-prox(n,k) are squared distances in a Euclidean space of high dimension.

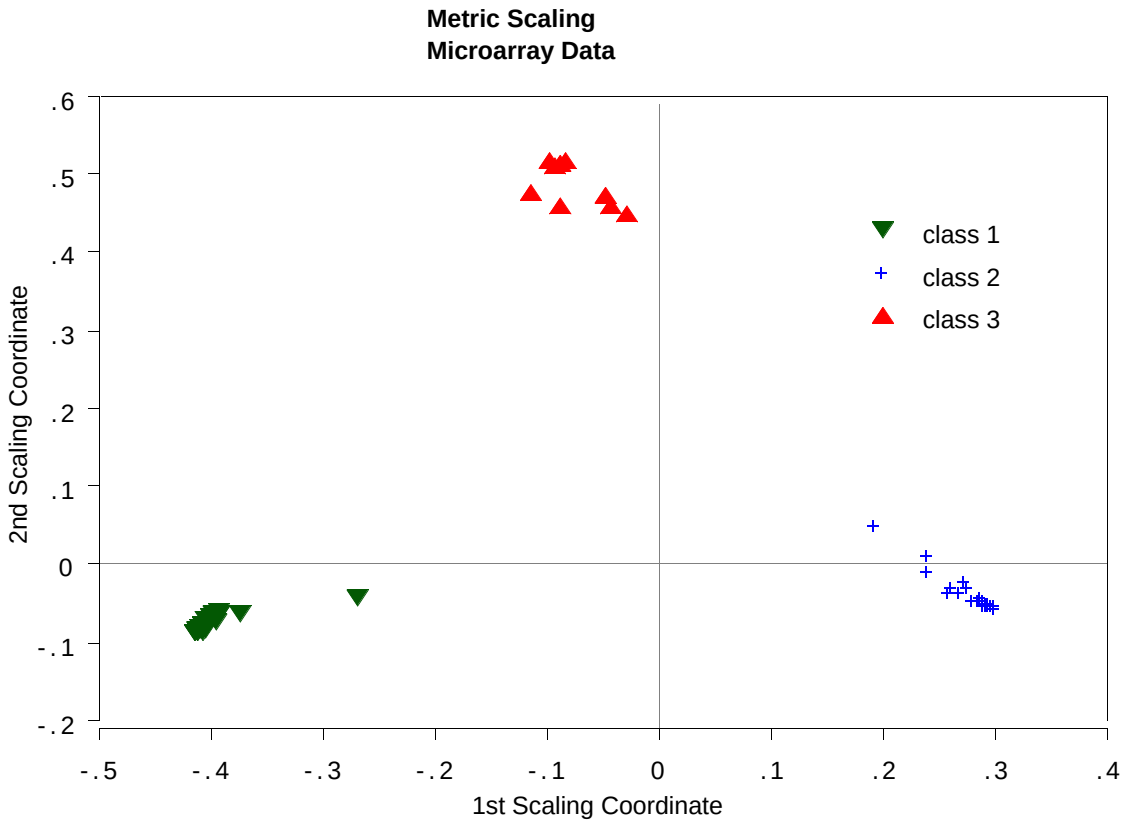
Then, one can compute *scaling coordinates* which project the data onto lower dimensional spaces while preserving (as much as possible) the distances between them.

We illustrate with three examples. The first is the graph of 2nd vs. 1st scaling coordinates for the liver data



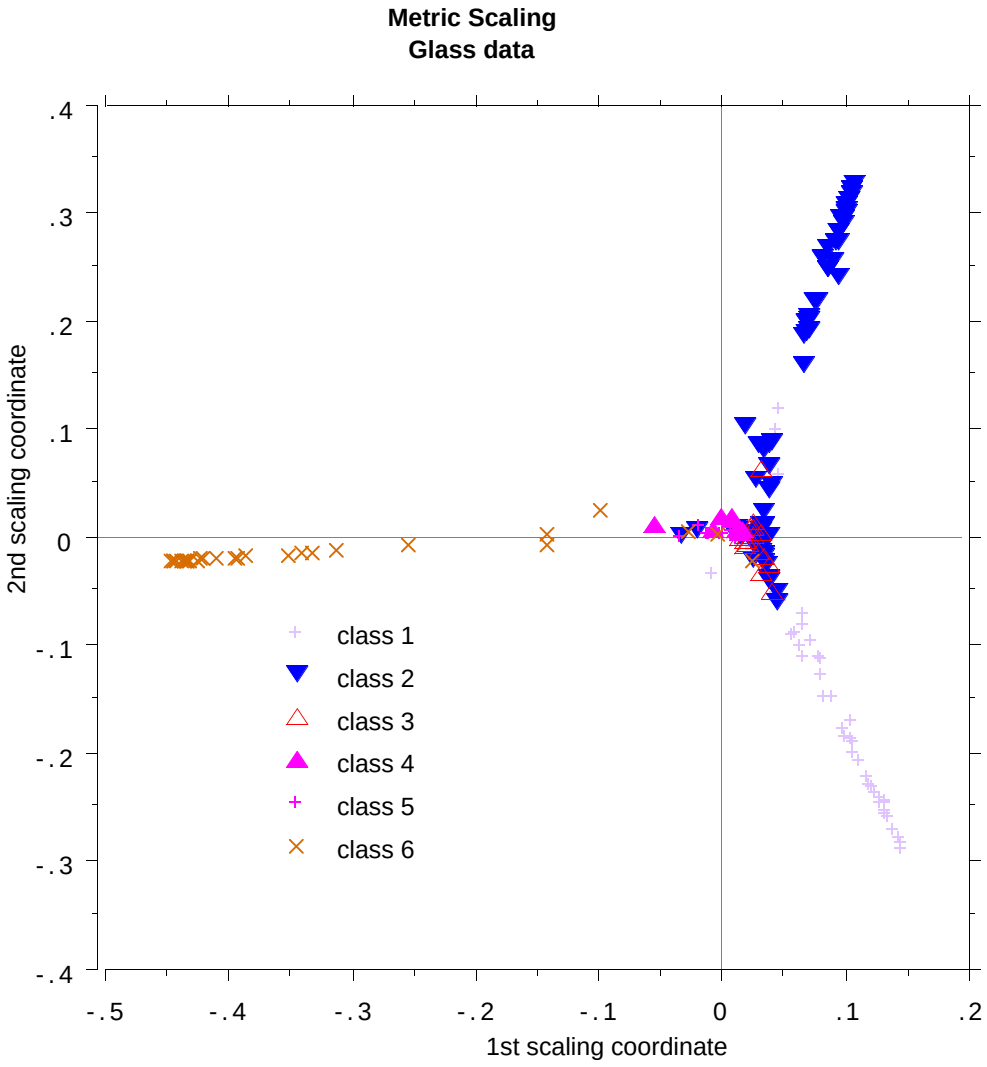
The two arms of the class #2 data in this picture correspond to the two clusters found and discussed above.

The next example uses the microarray data. With 4682 variables, it is difficult to see how to cluster this data. Using proximities and the first two scaling coordinates gives this picture:



Random forests misclassifies one case. This case is represented by the isolated point in the lower left hand corner of the plot.

The third example is glass data with 214 cases, 9 variables and 6 classes. This data set has been extensively analyzed (see Pattern recognition and Neural Networkks-by B.D Ripley). Here is a plot of the 2nd vs. the 1st scaling coordinates.:



None of the analyses to data have picked up this interesting and revealing structure of the data--compare the plots in Ripley's book.

We don't understand its implications yet.

OUTLIER LOCATION

Outliers are defined 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.

To define a measure of outlyingness,
we first compute, for a case n, the sum of the squares of prox(n,k) for all k in the same class as case n.

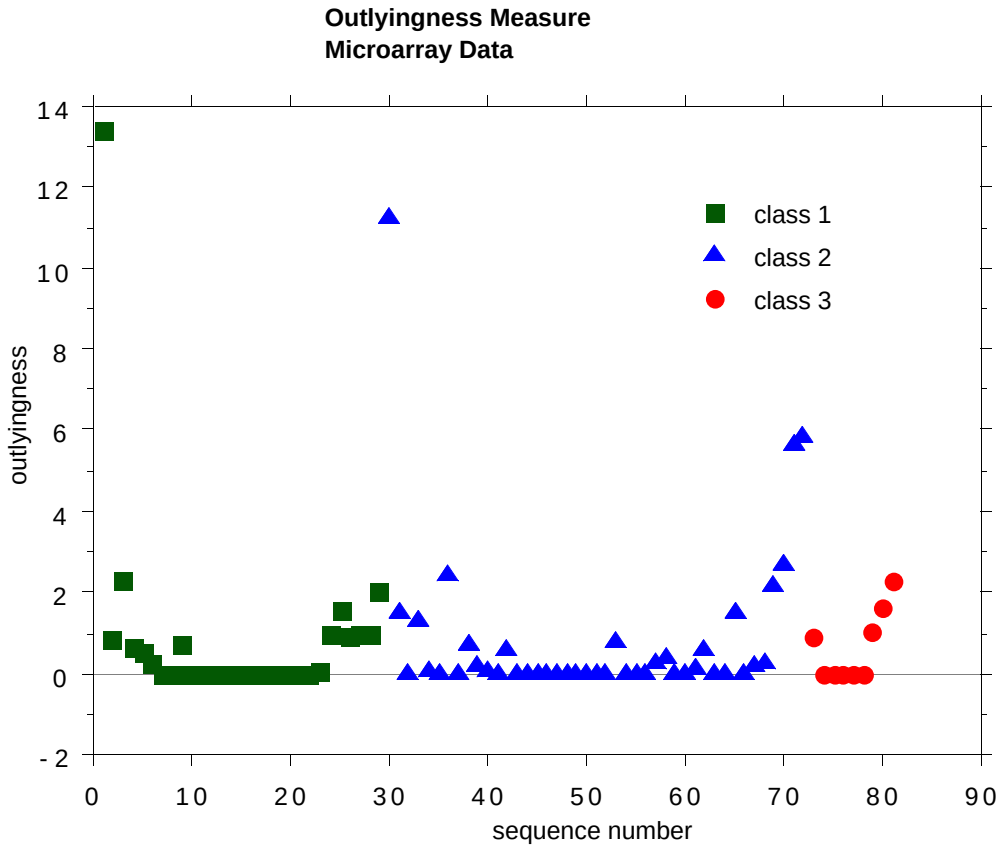
Take the inverse of this sum--it will be large if the proximities prox(n,k) from n to the other cases k in the same class are generally small.

Denote this quantity by out(n).

For all n in the same class, compute the median of the out(n), and then the mean absolute deviation from the median.

Subtract the median from each out(n) and divide by the deviation to give a normalized measure of outlyingness.

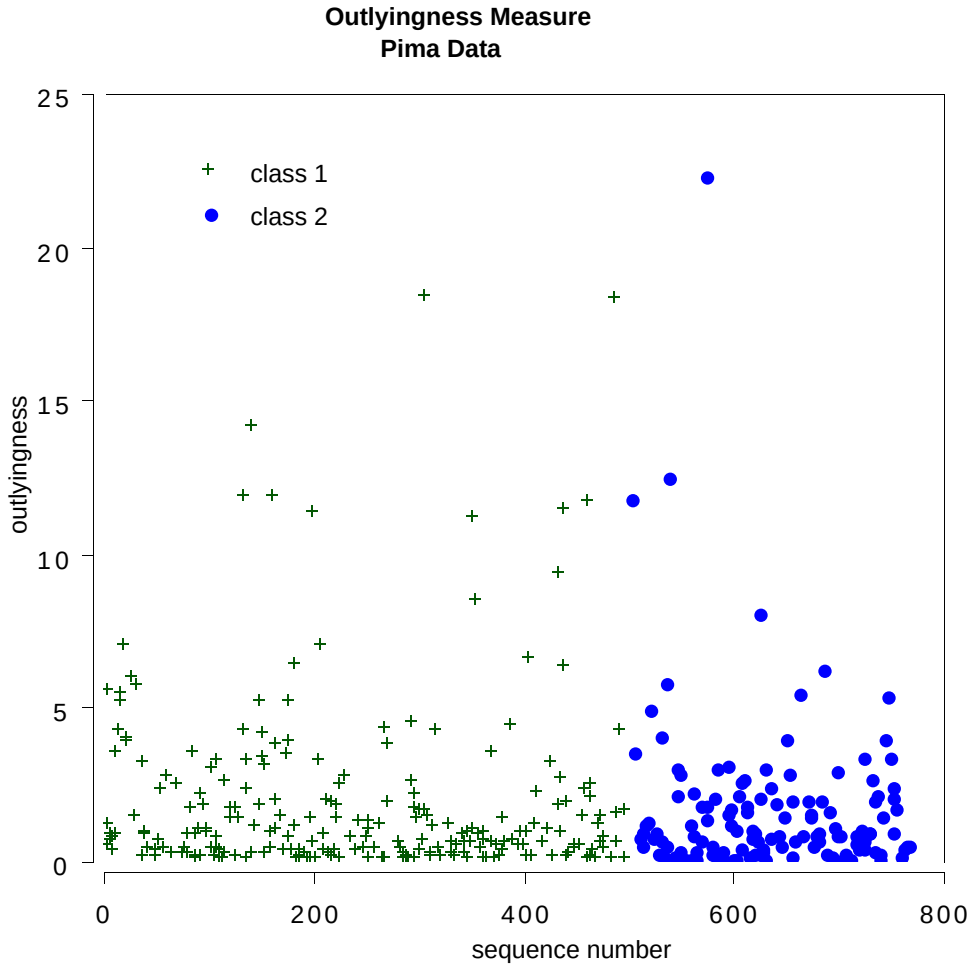
The values less than zero are set to zero. Generally, a value above 10 is reason to suspect the case of being outlying. Here is a graph of outlyingness for the microarray data



There are two possible outliers--one is the first case in class 1, the second is the first case in class 2.

As a second example, we plot the outlyingness for the Pima Indians hepatitis data.

This data set has 768 cases, 8 variables and 2 classes.
It has been used often as an example in Machine Learning research and is suspected of containing a number of outliers.



If 10 is used as a cutoff point, there are 12 cases suspected of being outliers.

ANALYZING UNLABELED DATA

Using an interesting device, it is possible to turn problems about the structure of unlabeled data (i.e. clusters, etc.) into a classification context.

Unlabeled date consists of N vectors {**x**(n)} in M dimensions. These vectors are assigned class label 1.

Another set of N vectors is created and assigned class label 2.

The second synthetic set is created by independent sampling from the one-dimensional marginal distributions of the original data.

For example, if the value of the mth coordinate of the original data for the nth case is x(m,n), then a case in the synthetic data is constructed as follows:

Its first coordinate is sampled at random from the N values x(1,n), its second coordinate is sampled at random from the N values x(2,n), and so on.

Thus the synthetic data set can be considered to have the distribution of M independent variables where the distribution of the mth variable is the same as the univariate distribution of the mth variable in the original data.

RUN RF

When this two class data is run through random forests a high misclassification rate--say over 40%, implies that there is not much dependence structure in the original data.

That is, that its structure is largely that of M independent variables--not a very interesting distribution.

But if there is a strong dependence structure between the variables in the original data, the error rate will be low.

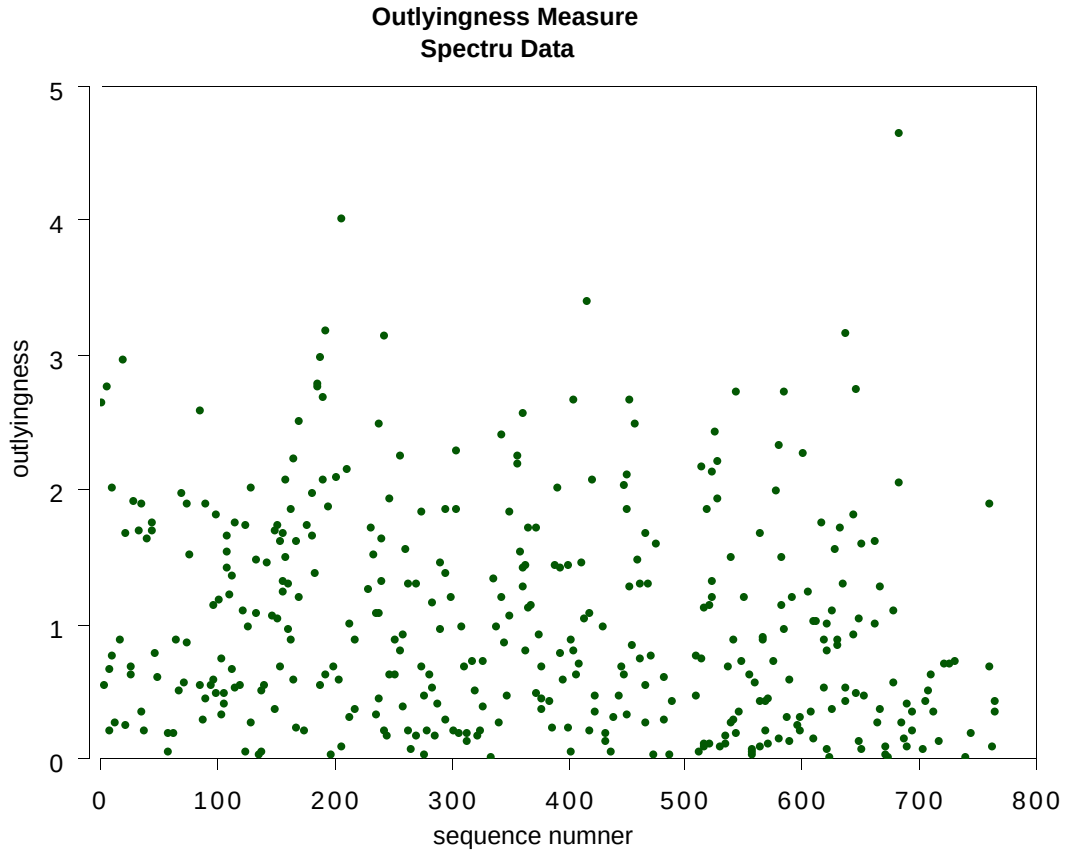
n this situation, the output of random forests can be used to learn something about the structure of the data.

The following is an example that comes from data supplied by Merck.

APPLICATION TO CHEMICAL SPECTRA

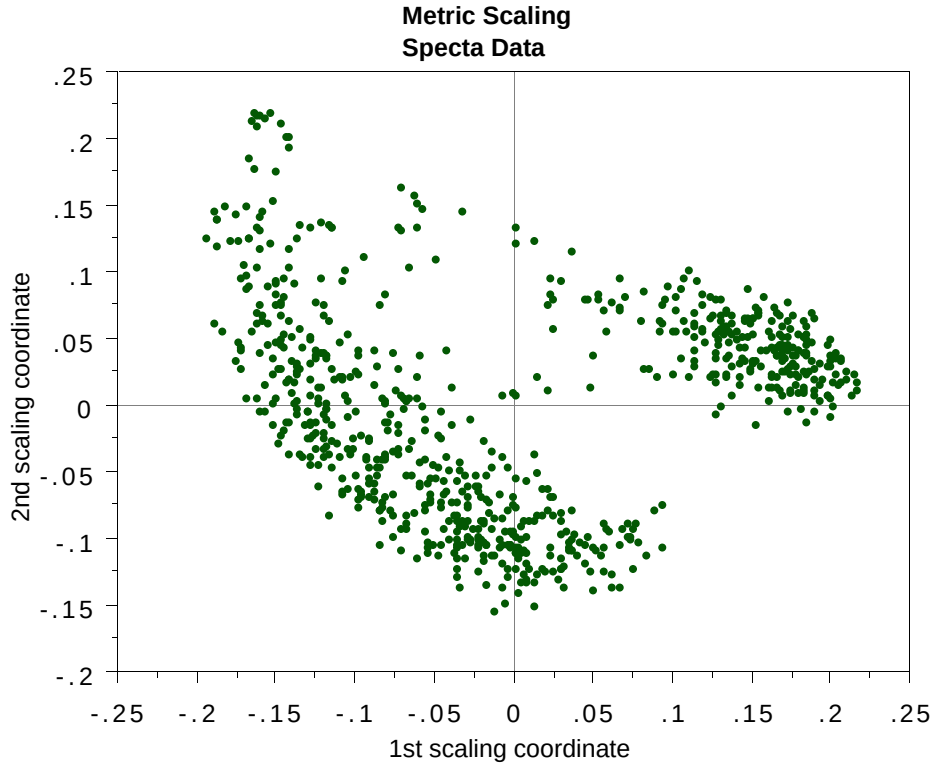
Data supplied by Merck consists of the first 468 spectral intensities in the spectrums of 764 compounds. The challenge presented by Merck was to find small cohesive groups of outlying cases in this data.

Creating the 2nd synthetic class there was excellent separation with an error rate of 0.5%, indicating strong dependencies in the original data. We looked at outliers and generated this plot.



USING SCALING

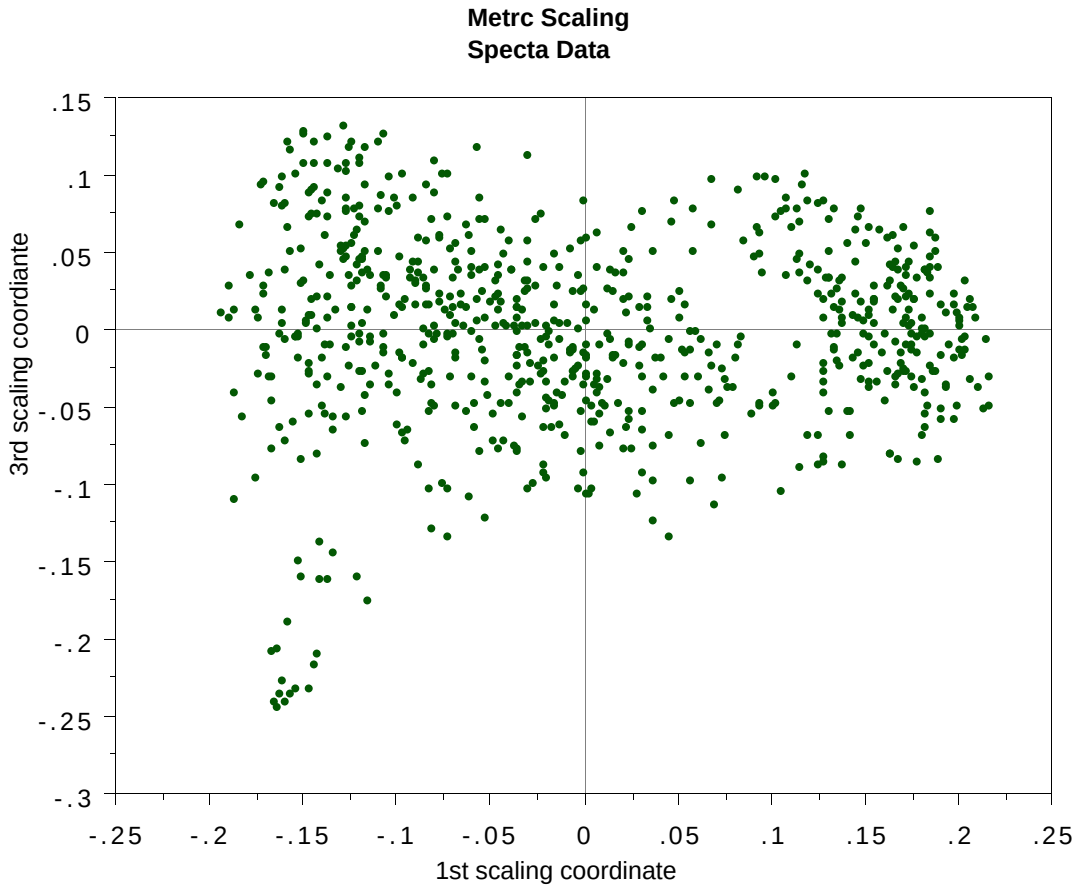
This plot gives no indication of outliers. But outliers must be fairly isolated to show up in the outlier display. To search for outlying groups scaling coordinates were computed. The plot of the 2nd vs. the 1st is below:



This shows, first, that the spectra fall into two main clusters. There is a possibility of a small outlying group in the upper left hand corner.

To get another picture, the 3rd scaling coordinate is plotted vs. the 1st.

ANOTHER PICTURE



The group in question is now in the lower left hand corner and its separation from the body of the spectra has become more apparent.

TO SUMMARIZE

i) With any model fit to data, the information extracted is about the model--not nature.

ii) The better the model emulates nature, the more reliable our information.

iii) A prime criterion as to how good the emulation is the error rate in predicting future outcomes.

iv) The most accurate current prediction algorithms can be applied to very high dimensional data, but are also complex.

v) But a complex predictor can yield a wealth of "interpretable" scientific information about the prediction mechanism and the data.

CURTAIN!

Curtain Call:

Random Forests is free software.

www.stat.berkeley,edu / users / breiman