

Boosted PRIM with Application to Searching for Oncogenic Pathway of Lung Cancer

Pei Wang
Department of Statistics
Stanford University
Stanford, CA 94305
wp57@stanford.edu

Young Kim, Jonathan Pollack
Department of Pathology
Stanford University
Stanford, CA 94305

Robert Tibshirani
Department of HRP and Statistics
Stanford University
Stanford, CA 94305
tibs@stanford.edu

Abstract

Boosted PRIM (Patient Rule Induction Method) is a new algorithm developed for two-class classification problems. PRIM is a variation of those Tree-Based methods ([4] Ch9.3), seeking box-shaped regions in the feature space to separate different classes. Boosted PRIM is to implement PRIM-styled weak learners in Adaboost, one of the most popular boosting algorithms. In addition, we improve the performance of the algorithm by introducing a regularization to the boosting process, which supports the perspective of viewing boosting as a steepest-descent numerical optimization by Jerry Friedman [3].

The motivation for Boosted PRIM is to solve the problem of "searching for oncogenic pathways" based on array-CGH (Comparative Genomic Hybridization) data, though the algorithm itself is suitable for general classification problems. We illustrate the performance of the method through some simulation studies as well as an application on a lung cancer array-CGH data set.

1. Introduction

Studies have shown that some of the genetic alterations are causally associated with cancer development. The concept "Oncogenic pathway" describes an accumulation of genetic events which are responsible for tumor progression. Obviously, models for tumor progression pathways would be of great value for the early diagnosis or treatment of cancers. Therefore it is worthwhile to develop statistical and computational methods to study such pathway structures [5] [1].

Genomic DNA copy number alterations are key genetic events in the development of human cancers [6]. Normally a human cell contains two copies of each of the 22 non-sex chromosomes; when genetic alterations occur, the DNA

copy numbers will vary away from two. Array Comparative Genomic Hybridization (array-CGH) is an approach to scan for genome-wide differences in DNA copy numbers. In a typical experiment, a tumor sample labelled red (Cy5) is hybridized to a reference normal sample labelled green (Cy3). For each gene/clone (one spot on the chips), the scanner reports the ratio of the red light intensity to the green light intensity, which corresponds to the ratio of the DNA copy number of the gene/clone in the tumor sample to that of the normal sample. A more elaborate introduction to array-CGH can be found in [7].

Based on the array-CGH data sets, the pathway structures can be studied with statistical and computational methods.

Several pioneering studies have been done in this field. Desper and *et al.* proposed a few tree-styled models for oncogenic pathways and developed computational approaches for data analysis [2] [5]. In another study, Michael A. Newton suggested an innovative probability model to describe tumor development and successfully implemented the Monte Carlo Markov Chain (MCMC) for model fitting [1].

However all these methods were designed for old CGH profiles, in which the outcomes are discrete numbers: 0 or ± 1 (+1 stands for amplifications while -1 stands for deletions). In addition, the above methods have some limitations on modelling. R. Desper and *et al.* used specified tree-styled models to represent pathway structures [2] [5], while the true biological process are always more complicated. M. A. Newton's stochastic model can only handle non-overlapping pathways; however, overlapping cases are more common for complex diseases [1].

In this paper, we introduce a new algorithm "Boosted PRIM" for pathway inference, which is designed for continuous measurements of high-resolution CGH array data. Since the method is totally nonparametric, we do not need any heuristic assumptions for the pathway structures. In addition, the method can effectively handle overlapping cases.

The idea of Boosted PRIM is to view the pathway problem from a classification angle. Given the CGH profiles of both the tumor samples (tumor vs. normal hybridization) and the normal samples (normal vs. normal hybridization), the true pathway structure should provide ideal classification rules separating tumor samples from normal samples. Therefore, good classifiers should also conversely cast light on the corresponding pathway structures.

PRIM (Patient Rule Deduction Method) is a two-class classification method ([4], 279), chosen for this specific problem due to the inherent biological properties of array-CGH data. In addition, in order to overcome the difficulty of the limited sample size (around one hundred) in most array-CGH studies, we implement a boosting procedure with PRIM-styled weak learners. Moreover, we propose a new regularization strategy for boosting to improve the performance of the algorithm.

Section 2 describes the statistical model for pathway structures as well as the tumor development. The algorithm of Boosted PRIM is explained in Section 3. The performance of the algorithm on some simulation studies is shown in Section 4. An application of the algorithm to a Lung Cancer array-CGH data set is presented in Section 5, followed by a brief discussion in Section 6.

2. Statistical Model Set Up

Based on M. Newton's Instability-Selection-Network model [1], a similar set up is used to describe the pathway structure as well as the process of cancer development, although it is adapted for continuous measurements in array-CGH experiments.

The outcome of the array-CGH experiment for one sample (person) can be denoted as:

$$\{(Y : X) : Y \in \{0, 1\}; X = (x_1, x_2, \dots, x_p), x_i \in R^1\},$$

where Y is an indicator variable, 1 stands for tumor samples, and 0 for normal samples; x_i is the $\log_2 \left(\frac{\text{red light intensity}}{\text{green light intensity}} \right)$ of the i th gene/clone.

As mentioned earlier, in normal human cells, the DNA copy number of a gene/clone should be exactly 2, which corresponds to the measurement of $x_i = 0$. But when deletion or amplification occurs, the DNA copy number will deviate from 2, corresponding to $x_i < 0$ or $x_i > 0$.

An oncogenic pathway is an accumulation of genetic events, or more specifically, a set of DNA copy number alterations, which are responsible for tumor progression. A pathway will be called "open" if all the genetic events in this set are true. Moreover, because of the heterogeneity of complex diseases, more than one oncogenic pathway may exist for any kind of tumor. Thus, we define "pathway structure" as a set of all possible oncogenic

pathways. It is thus reasonable to assume that a cell appears as a tumor cell if at least one pathway is open in the cell.

Now, with array-CGH data, we can statistically describe above the genetic events and tumor development as follows:

1. **Mutation occurs to the i^{th} gene:**

$$I(x_i \in [-\alpha', \alpha]),$$

where $[-\alpha', \alpha]$ is a neighborhood of zero. $x_i > \alpha$ represents the amplification of the i^{th} gene, while $x_i < -\alpha'$ represents the deletion of the i^{th} gene.

2. **A pathway is open** (all the genes in the pathway have mutations):

$$O(X, PA_j) = \prod_{i_j \in PA_j} I(\{x_i > \alpha_j\} \cup \{x_i < -\alpha'_j\})$$

3. **The sample is selected to be a tumor cell** (at least one pathway is open):

$$Y = S(X, PA) = I\left(\sum_{j=1}^K O(X, PA_j) > 0\right) \quad (1)$$

This model can be illustrated through the following example. Suppose the pathway structure consists of two pathways. One pathway is controlled by genes 1, 2, and 5, while the other is related to genes 3, 4, 5 and 6. We denote the pathway structure as

$$PA = \{\{1, 2, 3\}, \{3, 4, 5, 6\}\}$$

The cell will be observed as a tumor cell if either genes 1,2,3 cannot function normally at the same time, or genes 3,4,5 and 6 cannot function normally at the same time. Thus we have the following equation: (Note: to simplify the notation, we assume that all the "disfunctions" in this example are amplifications)

$$Y = \begin{cases} 1, & \text{if } ((x_1 > \alpha_1) \cap (x_2 > \alpha_2) \cap (x_3 > \alpha_3)) \cup \\ & (x_3 > \alpha_3) \cap (x_4 > \alpha_4) \cap (x_5 > \alpha_5) \cap (x_6 > \alpha_6); \\ 0, & \text{otherwise.} \end{cases} \quad (2)$$

where $\{\alpha_i\}$ are unknown parameters.

Now the task of searching for oncogenic pathway can be represented as a statistical problem: how to recover unknown parameter PA in the selection function S satisfying $Y = S(X, PA)$, where (Y, X) are observed data in CGH experiments.

Since Y is the class label, the above problem is quite similar as a classification problem. The selecting function (1) of pathways provides a perfect classification. In next section, we introduce a new two-class classification method adapted specifically to these pathway selection functions.

3. Boosted PRIM

3.1. PRIM

PRIM stands for “Patient Rule Induction Method”. It is a variation of those Tree-Based Methods([4] Ch9.3). PRIM finds boxes in the feature space in which the response average is high (or low). The algorithm is illustrated in **Figure 2**. The searching procedure is as follows:

1. Starts with a box containing all of the data.
2. Compress the box along one face by a small amount, and peel off the observations falling outside the box; the face chosen for compression is the one resulting in the largest box mean, after the compression is performed.
3. The process is repeated, until the current box contains some minimum number of data points.

As explained in *Section 2*, the “selecting rule” implied by one pathway can be expressed as

$$\{x_{i_1} > \alpha_1\} \cap \{x_{i_2} > \alpha_2\} \cap \dots \{x_{i_k} > \alpha_k\}, \quad (3)$$

(again, to simplify the notation, only the amplification cases are shown here), which is a “corner box” in the X space. PRIM exactly aims at searching such kind of box-shaped region which best separates the two class samples.

However, since the sample size of the array-CGH data is always limited, PRIM alone cannot provide satisfactory results. Therefore, in *Section 3.2* we introduce another powerful tool “Boosting”, a currently very popular method in the field of machine learning, and explain how to combine it together with PRIM.

3.2. Boosting

Boosting is a method for iteratively building an additive model

$$F(x) = \sum_j \alpha_j h_j(x), \quad (4)$$

where $h_j \in \mathcal{H}$ and \mathcal{H} is a large family of candidate predictors or “weak learners”.

3.2.1. Adaboost and PRIM. In [3], Friedman raised a statistical perspective, which connects the boosting methods with steepest-descent minimization. Especially when y belongs to $\{-1, 1\}$ and the loss function $L(y, F)$ is $\exp(-yF)$, the stage-wise steepest-descent strategy is exactly the popular Adaboost (Freund and Schapire 1996) [4]. Following is an outline for the Adaboost algorithm:

Algorithm I: Adaboost

1. Initialize the observation weights $w_i = 1/N$, $i = 1, 2, \dots, N$.
2. For $m=1$ to M
 - (a) Fit a classifier $G_m(x)$ to the training data using weights w_i .
 - (b) Compute

$$err_m = \frac{\sum_{i=1}^N w_i I(y_i \neq G_m(x_i))}{\sum_{i=1}^N w_i}$$

- (c) Compute $\alpha_m = \log((1 - err_m)/err_m)$.
 - (d) Set $w_i \leftarrow w_i \cdot \exp[\alpha_m \cdot I(y_i \neq G_m(x_i))]$, $i = 1, 2, \dots, N$
3. Output $G(x) = \text{sign}[\sum_{m=1}^M \alpha_m G_m(x)]$.

To combine boosting and PRIM, we can simply plug in the PRIM styled weak learner $G_m(x)$ — a box region indicator function — into the above algorithm. However, to make the algorithm more efficient, we suggest following regularization strategy.

3.2.2. Regularization for Adaboost. In the prediction problems, fitting the training data too closely can be counterproductive. Reducing the expected loss on the training data beyond some point causes the population-expected loss. Regularization methods attempt to prevent such “overfitting” by constraining the fitting procedure [3]. Friedman used a shrinkage strategy in his algorithm MART to regularize the stage-wise learning process of Boosting, which simply replace α_m with $\nu\alpha_m$ in the weight updating step 2(d) of *Algorithm I*, where ν is a “learning rate” parameter.

However, we can even be more conservative to set α_m to a fixed ε for all M iterations. Thus, the algorithm only moves a fixed but very small step along the steepest-descent direction in each searching iteration, instead of doing the line search in the original Adaboost. This strategy is clearly more time consuming, but since it learns at a more conservative pace, it can be less vulnerable to outliers or noise in the data.

We illustrate the effect of both “shrink learning rate (ν)” regularization and “fix small step (ε)” regularization through a simulation study. The training sample consists of 572 observations $\{y_i, \mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{i20})\}$, which are randomly generated as described in *Section 4*. 856 more observations are generated as the testing sample set. We then apply the two regularized *Algorithm I* with a PRIM-style “weak learner” to the training sample set. **Figure 3** shows the misclassification rate on the test sample set as a function of the number of iterations M , for different regularization methods and parameters. We can see that the regularization method with fixed moving length $\varepsilon = 0.1$ and

$\varepsilon = 0.05$ greatly outperform the original *Adaboost* as well as the shrinkage regularized *Adaboost*. This supports the idea that a lower learning rate is preferable in the boosting process.

The outline of the final version of Boosted PRIM algorithm is as follows:

Algorithm II: Boosted PRIM

1. Initialize the observation weights $w_i = 1/N$, $i = 1, 2, \dots, N$.
2. For $m=1$ to M
 - (a) Fit a PRIM style classifier $G_m(x)$ to the training data using weights w_i .
 - (b) Begin with a big box containing all the training points.
 - (c) Peels away points along one edge by δ percent in order to maximize the weighted \bar{Y} of the points remaining in the box.
 - (d) Repeat (c) for N_0 times; get a sequence of boxes, from large to small; select the one (B) with the maximum weighted \bar{Y} ; then $G_m(x) = 2I\{x \in B\} - 1$.
 - (e) Set $w_i \leftarrow w_i \cdot \exp[-\varepsilon \cdot y_i \cdot G_m(x_i)]$, $i = 1, 2, \dots, N$
3. Output $G(x) = \text{sign}[\sum_{m=1}^M G_m(x)]$.

3.3. Interpretation

As we stated in *Section 2*, our ultimate goal is to estimate the parameter PA in the selection function

$$Y = S(X, PA) = I \left(\sum_{j=1}^K O(X, PA_j) > 0 \right). \quad (5)$$

From *Section 3.1* we know that the PRIM styled weak learners can be represented as

$$G_m(x) = \begin{cases} 1, & x \in B_m \\ -1, & \text{otherwise} \end{cases}, \quad (6)$$

where B_m is a corner box in the X space (without lose of general, assume only amplifications occur):

$$B_m = \{\mathbf{x} : \{x_{i_1^m} > \alpha_1^m\} \cap \{x_{i_2^m} > \alpha_2^m\} \cap \dots \cap \{x_{i_{k_m}^m} > \alpha_{k_m}^m\}\} \quad (7)$$

Since the "selecting rule" implied by one pathway is right a "corner box" in the X space, comparing (6) with (2), we can see that $G_m(x)$ is exactly the selection function of $PA_m = \{i_1^m, i_2^m, \dots, i_{k_m}^m\}$.

However the output

$$\hat{Y} = G(x) = \text{sign} \left[\sum_{m=1}^M G_m(x) \right]. \quad (8)$$

of Boosted PRIM is a series of box-indicator functions. Thus, in order to recover PA based on (8), we introduce a new statistic "Variable Importance" to summary across all $\{G_m(x)\}_{m=1}^M$.

3.3.1. Variable Importance (VI) Matrix We can write B_m in Equation(7) as a p -dimension box

$$B_m = \{\mathbf{x} : \{x_1 \in I_1^m\} \cap \{x_2 \in I_2^m\} \cap \dots \cap \{x_p \in I_p^m\}\} \quad (9)$$

where for $j \in \{i_1^m, i_2^m, \dots, i_{k_m}^m\}$

$$I_j^m = \{x : x > \alpha_j^m\};$$

otherwise I_j^m is $(-\infty, \infty)$. Then define a $p \times p$ matrix $\{VI_{ij}\}_{i,j=1}^p$ with each entry

$$VI_{ij}^m = \widetilde{VI}_{ij}^m \cdot \min(\widetilde{VI}_{ii}^m, \widetilde{VI}_{jj}^m) / \max(\widetilde{VI}_{ii}^m, \widetilde{VI}_{jj}^m),$$

where

$$\widetilde{VI}_{ij}^m = \frac{\sum_{\{k:(x_{k_i} \in I_i^m) \cap (x_{k_j} \in I_j^m)\}} Y_k}{\sum_{\{k:(x_{k_i} \in I_i^m) \cap (x_{k_j} \in I_j^m)\}} 1}.$$

VI_{ii}^m reflects the contribution of the i^{th} feature to the "separation" ability of this box, while VI_{ij}^m for feature pairs (i, j) . If gene i and gene j are two components of the same pathway, we would expect a large value of VI_{ij} .

Figure 4 is an example of the result VI matrix from a simulation study.

3.3.2. Summary across M iterations For each iteration of the boosting process, we obtain a box-indicator function and then a VI matrix. The most straight forward way to summarize a group of box-indicator functions of the same pathway is to calculate the average VI matrix. However, multiple pathways always exist at the same time for most diseases, which means that iterations from boosting process always correspond to more than one pathway. Thus, we need to do a clustering for boosting iterations first, such that iterations of the same pathway would be grouped to the same cluster, and then calculate the average VI matrix for each cluster, which gives us the possible components of the corresponding pathway.

4. Simulation Study

We show the performance Boosted PRIM through a simple simulation example.

Assume that $PA = \{\{1, 2, 3\}, \{3, 4, 5\}\}$, $p = 20$. Simulate CGH measurement for the 20 genes independently from standard normal

$$X^i = (x_1^i, \dots, x_p^i), x_j^i \sim N(0, 1).$$

Then, for each X^i , we can assign it to class 1 or class 0 through

$$Y^i = \begin{cases} 1, & \text{if } ((x_1 > \alpha) \cap (x_2 > \alpha) \cap (x_3 > \alpha)) \vee \\ & ((x_4 > \alpha) \cap (x_5 > \alpha) \cap (x_6 > \alpha)); \\ 0, & \text{otherwise.} \end{cases}$$

Set $\alpha = 0.3$. Simulate 100 points from class 1 and 100 points from class 0. The results are shown in **Figure 4**. As can be seen, for this simple set-up, Boosted PRIM accurately identifies the true pathway structure PA .

5. Application to a Lung Cancer CGH array data

Array CGH experiments were performed on 48 lung cancer cell lines (Young Kim and Jonathan Pollack, unpublished). For each sample, the $\log_2(\frac{red}{green})$ for around 25000 genes/clones were collected. Since nearby genes/clones tend to have similar amplification or deletion behavior, it is more meaningful to investigate at “region” levels, instead of at “gene/clone” levels. In addition, eliminating the dimension of the data set helps improve the powers of the later statistical analysis.

Thus, each array (a vector of about 25000 elements) is first converted to a vector of 798 elements. Each element is the mean value of the measurement of one cytoband. 789 cytobands are then further combined to 49 regions through the Fixed Order Clustering method. After filtering, 21 interesting regions are selected from the original 49.

Since few arrays of normal samples were available in this study, we simulate 100 normal samples by using an empirical Gaussian $N(0, 0.1)$ distribution, for the measurements of normal arrays can be simply considered random noises.

Apply Boosted PRIM to the selected 21 regions of 48 lung cancer cell line samples as well as the 100 pseudo normal samples. The average VI matrix across all boosting iterations is illustrated in **Figure 1**. We can see that there is a strong interaction between region 5 ($4q11 - 4q35.2$) and region 20 ($20p13 - 20q13.3$). Also the result suggests four other regions (Region 7 : $6q12 - 6q27$; Region 9 : $8p23.3 - 8p11.21$; Region 12 : $10q11.21 - 10q26.3$; Region 13 : $11q12.1 - 11q21$), which may be in the same pathway as $4q11 - 4q35.2$ and $20p13 - 20q13.3$. Further work need to be done to investigate the biological meaning of these interactions.

6. Discussion

Boosted PRIM is a powerful algorithm for a two-class classification, whose prediction error is comparable with MART (as shown in other studies). Since its classification rules have special shape in the feature space, it can be effec-

tively applied to solving the pathway problem with array-CGH data.

However, in reality, oncogenic pathways involve not only DNA copy number alterations, but also many other genetic events that cannot be detected through the array-CGH technique. Therefore, in order to understand the complex biological process of tumor development, other information, such as the sequence data of the genome, the RNA expression level, and the protein synthetic amount, should also be considered.

Acknowledgments Pei Wang was partially supported by the Stanford Graduate Fellowship. Robert Tibshirani was partially supported by National Science Foundation Grant DMS-9971405 and National Institute of Health Contract N01-HV-28183.

References

- [1] M. A. Newton. A statistical method to discover significant combinations of genetic aberrations associated with cancer using comparative genomic hybridization profiles. 2001.
- [2] R. Desper, F. Jiang, O. Kallioniemi, H. Moch, D. Papadimitriou, and A. A. Schäffer. Distance-based reconstruction of tree models for oncogenesis. *J. Comp. Bio.*, (7):789–803, 2000.
- [3] J. H. Friedman. Greedy function approximation: A gradient boosting machine. *IMS*, 1999 Reitz Lecture.
- [4] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. 2001.
- [5] F. Jiang, R. Desper, C. H. Papadimitriou, A. A. Schäffer, O.-P. Kallioniemi, J. Richter, P. Schraml, G. Sauter, M. J. Mihatsch, and H. Moch. Construction of evolutionary tree models for renal zcell carcinoma from comparative genomic hybridization data. *Cancer Research*, 60:6503–6509, 2000.
- [6] C. Lengauer, K. Kinzler, and B. Vogelstein. Genetic instabilities in human cancers. *Nature*, 396, 1998.
- [7] D. Pinkel, R. Seagraves, D. Sudar, S. Clark, I. Poole, D. Kowbel, C. Collins, W. Kuo, C. Chen, Y. Zhai, S. Dairkee, B. Ljung, J. Gray, and D. Albertson. High resolution analysis of dna copy number variation using comparative genomic hybridization to microarrays. *Nature Genetics*, 20, 1998.

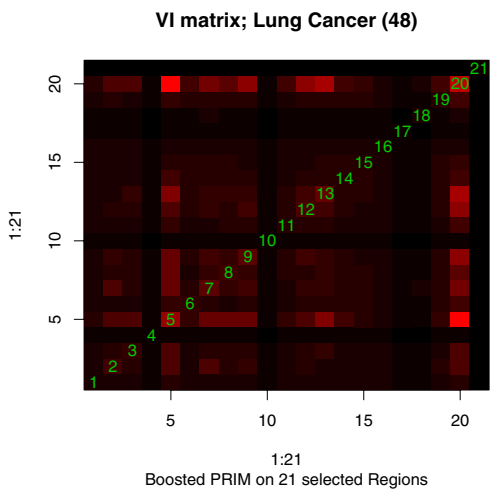


Figure 1. VI matrix for lung cancer study. The suggested pathway is (5, 7, 9, 12, 13, 20).

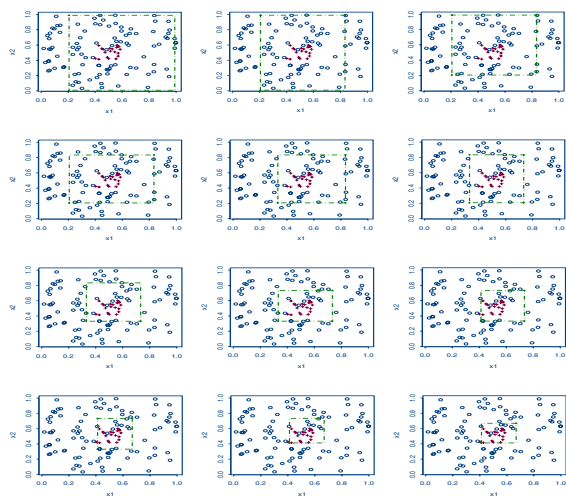


Figure 2. PRIM algorithm. There are two classes: red—class 1; blue—class 0. The procedure begins starting at the top left panel, the sequence of peelings is shown, until a pure red region is isolated in the bottom right panel.

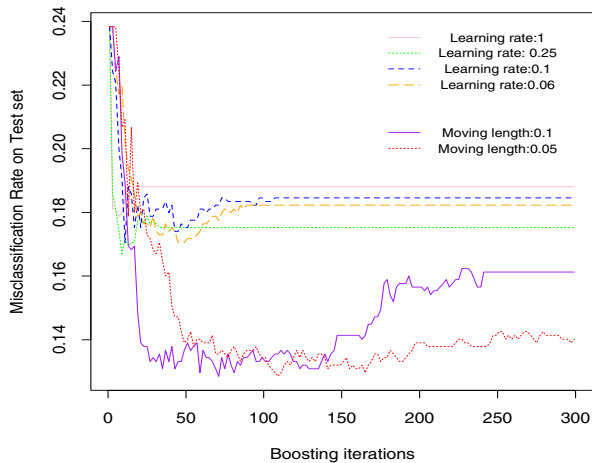


Figure 3. Misclassification rate as a function of number of iterations M for simulate data set. Four curves correspond to shrinkage parameter values of $\nu \in \{1.0, 0.25, 0.1, 0.06\}$; and the other two correspond to fixed moving length $\varepsilon \in \{0.1, 0.05\}$.

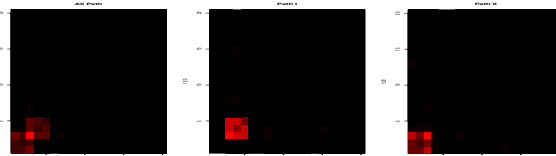


Figure 4. Variable Importance Matrix. Each square represents a 20×20 matrix. In one square, the small block at position (i, j) stands for the (i, j) th entry of the matrix. Color of the small block changes from red to black when the value of the entry changes from large to small. The leftmost square is the average VI matrix of all boosting iterations. The right two squares are the average VI matrix of two different clusters. The first cluster (the middle square) suggests the pathway (3, 4, 5). The second cluster (the right square) suggests the pathway (1, 2, 3).