Identifying pair-wise gene functional similarity by multiplex gene expression maps and supervised learning

Li An
Data Engineering Laboratory,
Center for Data Analytics and
Biomedical Informatics, Temple
University
415 Wachman Hall,1805 N. Broad
St., Philadelphia, PA 19122
215-204-9260
anli@temple.edu

Haibin Ling
Center for Data Analytics and
Biomedical Informatics, Temple
University
305 Wachman Hall, 1805 N. Broad
St.
Philadelphia, PA 19122
215-204-6973
hbling@temple.edu

Zoran Obradovic
Center for Data Analytics and
Biomedical Informatics, Temple
University
303 Wachman Hall, 1805 N. Broad
St.
Philadelphia, PA 19122
215-204-6265
zoran.obradovic@temple.edu

Desmond J. Smith
Department of Molecular and
Medical Pharmacology, David
Geffen School of Medicine, UCLA
CHS, Los Angeles, CA 90095
2nd line of address
310-206-0086
dsmith@mednet.ucla.edu

Vasileios Megalooikonomou
Data Engineering Laboratory,
Center for Data Analytics and
Biomedical Informatics, Temple
University
314 Wachman Hall,1805 N. Broad
St.,
Philadelphia, PA 19122
215-204-5774
vasilis@temple.edu

ABSTRACT

Research has been done to explore the relationships between the Gene Ontology-based similarity and gene expression profiles in the mammalian brain. However, little attention has been paid to the location information of a gene's expressions. Gene expression maps, which contain spatial information regarding the expression of genes in mice's brain, are obtained by combining voxelation and microarrays. Based on the hypothesis that genes with similar gene expression maps may have similar gene functions, we propose an approach to identify pair-wise gene functional similarities by gene expression maps. By considering pairs of genes from an original dataset as samples whose features are extracted from expression maps and labels are the functional similarities of pairs of genes, we explore the relationship between similarities of gene maps and gene functions. We restrict the dataset to genes that are associated with previously detected functional expression profiles to strengthen the relationship. We use AdaBoost, coupled with our proposed weak classifier, to analyze the dataset and predict the functional similarities. The experimental results show that with the increasing similarities of gene expression maps, the functional similarities are increased too. The boosting analysis can predict the functional similarities between genes to a certain degree. The weights of the features in the model indicate which features are significant for this prediction. These findings can potentially assist the biologists by providing helpful clues in predicting gene functions.

Categories and Subject Descriptors

G.4 [**Programming Languages**]: H.2.8 Database Applications - *Data mining, Image databases;* I.5 PATTERN RECOGNITION I.5.2 Design Methodology - *Classifier design and evaluation, Feature evaluation and selection;* I.5.4 Applications; J.3 LIFE AND MEDICAL SCIENCES, *Biology and genetics*.

General Terms

Algorithms, Experimentation.

Keywords

Functional similarity of genes, gene expression maps, boosting, weak classifiers, voxelation.

ACM-BCB '11, August 1-3, Chicago, IL, USA
Copyright © 2011 ACM 978-1-4503-0796-3/11/08... $10.00

1. INTRODUCTION

The Gene Ontology (GO) represents an important knowledge resource for describing the function of genes [1], and has been widely used for identifying similarities between gene functions based on the GO structure [2, 3]. Recently, research has been done to explore the relationships between the GO-based similarity and gene expression profiles [3-7] and the relationships between gene function annotation and gene sequence [8].

However, little research has taken into account the location of a gene's expressions in the mammalian tissue. Voxelation is a new approach that involves dicing the brain into spatially registered voxels (cubes). It produces multiple volumetric maps of gene expression analogous to the images reconstructed in biomedical imaging systems [9-10]. Related research suggests that voxelation is a useful approach for understanding how the genome constructs the brain [11]. Gene expression patterns (maps) obtained by voxelation show good agreement with known expression patterns [12].

Our previous analysis of the gene expression maps [12] focused on studying the relationship between the gene functions and gene expression maps. We determined the similarity of gene expression maps using the wavelet transform and the similarity between gene functions using the GO structure and appropriate distance measures. Clustering analysis was done to detect a number of gene clusters that have both similar gene expression maps and similar gene functions. These clusters were denoted as significant clusters. The study confirmed that the hypothesis that genes with similar gene expression maps have similar gene functions holds for a certain set of genes. We also successfully predicted gene functions with high accuracies by our proposed method - functional expression profiles, which are obtained from specific gene expression maps that are associated with given functions [13].

In this study, we identify pair-wise gene functional similarities from gene expression maps by employing learning-based techniques. In particular, we first form a new dataset by considering pairs of genes from the original dataset as samples. For each sample gene pair, the similarities or distances between the corresponding gene expression maps are used as features to describe it. The labels for gene pairs are the functional similarities between the pairs of genes. Consequently, we formulate the problem of identifying functional similarity between genes as a supervised learning problem.

We use AdaBoost as the basic framework for our learning and predicting task. In order to fit the dataset which has huge number of samples and limited number of features, we propose a novel weak classifier that efficiently captures the distribution of individual features. We further restrict the dataset to the genes which are associated with previously detected functional expression profiles to strengthen the relationship between gene functions and gene maps. The experimental results show that the pair-wise gene functional similarities are increased with increasing similarities of gene expression maps. In addition, the boosting analysis classifies, with a high accuracy, the gene pair samples into two classes: the pairs of genes with similar functions and those without. The analysis of feature selection in the learning process indicates which features are significant for identifying the functional similarity from gene expression maps. Those features can be located and visualized in the original image of mice's brain. These findings can be potentially used for predicting gene functions and providing helpful clues to biologists.

The rest of this paper is organized as follows. In the data and methods section, after describing the pair-wise samples of multiple gene expression maps and briefly discussing how to extract features from the original gene expression maps and indentify gene functions distance, we present approaches for identifying functional similarities of pair-wise samples by Boosting and our own weak classifier. In the results section, we present the experimental results of identifying relationship between similarity of gene expression maps and their functions, and the boosting analysis. The discussion section provides an analysis of the obtained results and ideas for future applications of this methodology.

2. DATA AND METHODS

2.1 Multiplex gene expression maps

Researchers at the David Geffen School of Medicine at UCLA used voxelation in combination with microarrays for acquisition of genome-wide atlases of expression patterns in the brain [11]. They acquired multiplex gene expression maps for 20,847 genes following the procedure below. A fresh brain is removed from a sacrificed mouse, and then a 1mm slice of the brain at the level of striatum is obtained. The coronal slice is cut by a matrix of blades that are spaced 1 mm apart, thus resulting in cubes (voxels) that are 1mm^3. These voxels are located in the slice as Figure 1 shows. A1, A2, B1 … are in red color because these voxels are empty cubes that are assigned to maintain a rectangular. So, each gene is represented by the gene expression values of 68 voxels that compose a gene expression map of a mouse brain. By using different colors to show different values of gene expression, the expression map for a certain gene can be visualized as in Figure 2.

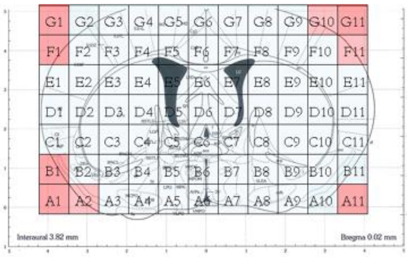
G1 G2 G3 G4 G5 G6 G7 G8 G9 G10 G11
F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 F11
E1 E2 E3 E4 E5 E6 E7 E8 E9 E10 E11
D1 D2 D3 D4 D5 D6 D7 D8 D9 D10 D11
C1 C2 C3 C4 C5 C6 C7 C8 C9 C10 C11
B1 B2 B3 B4 B5 B6 B7 B8 B9 B10 B11
A1 A2 A3 A4 A5 A6 A7 A8 A9 A10 A11
Interaural 3.82 mm
Bregma 0.02 mm

Figure 1 Voxels of the coronal slice

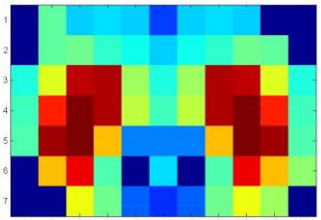

Figure 2 A visualized gene expression map

The data set we consider in this study is a 20,847 by 68 matrix, in which each row represents the log2 ratio of 68 expression values of a particular gene, and each column represents the expression values for all the probes (genes) at a given voxel. To reduce the effects of noise in the original dataset, we discard genes whose gene expression values fall in the range [-1,1].

Then, the remaining 13,576 genes' IDs are imported into the SOURCE [14] database [15] to retrieve their Gene Ontology (GO) annotation information. Out of the 13,576 genes, 7,883 genes are known genes and are annotated with at least one GO term. Our analysis is based on these genes. We denote the set of genes as G={g_1, g_2, …, g_N}, where N=7,883.

2.2 Pair-wise samples

Gene expression maps can be viewed as samples that can be analyzed using data mining techniques. However, the targets or labels associated with each sample are not always available, such as it is the case in our study. Instead, we reconstruct a new dataset by taking each pair of genes as a sample, and calculating the functional similarity of the gene pair which then becomes the label for that sample. As a result, the problem of identifying the relationship between gene expression maps and gene functions is formulated as a regression problem.

In the new dataset, each sample is a pair of gene expression maps. A sample is associated with the distance between the functions of its two genes such that the distance can be viewed as the label for the pair of gene maps. That is, a sample is defined as

(g_1, g_2) with a "label" $d_F(g_1,g_2)$,

where (g_1,g_2) is a pair of genes, and $d_F(g_1,g_2)$ is the desired function distance measure (defined in Section 2.3) for this pair of genes which we intend to approximate.

Suppose g_1, g_2, g_3, … are gene maps and $d_F(g_1, g_2)$, $d_F(g_2, g_3)$, $d_F(g_1, g_3)$ are gene function distances between the pairs (g_1, g_2), (g_2, g_3), (g_1, g_3) respectively, we have samples for all the gene pairs:

g_1, g_2	$d_F(g_1, g_2)$
g_1, g_3	$d_F(g_1, g_3)$
g_2, g_3	$d_F(g_2, g_3)$
…	…

Given the dataset and labels, the problem boils down to finding the relation between the gene expression maps similarity and the functional similarity between two genes. In our previous analysis on gene expression maps [12], we define the similarity between two gene expression maps as the Euclidean distance between their wavelet representations, and calculate the similarity (distance) between two gene functions based on gene ontology structures using Lin's method [16]. We have shown that the similarity between gene expression maps is positively correlated to the similarity between gene functions, which encourages the study of the relationship between gene maps and functional similarity of pairs of genes.

2.3 The gene function distance

We perform the analysis with respect to each one of the three gene ontologies, i.e., cellular component, molecular function and biological process. For example, in the category of biological process, if gene g_1 has functions $F(g_1)$ ={f_{11}, $f_{12, \ldots,}$ f_{1n}} and gene g_2 has functions $F(g_2)$ = {f_{21}, $f_{22, \ldots,}$ f_{2m}}, we define the function similarity (or distance) value between these two genes as the averaged functional distance of pairs of functions between the two genes. This is calculated using the following formula:

$$d_F(g_1,g_2)=\begin{cases}\frac{1}{\Gamma}\sum\limits_{f_1\in F(g_1)}\sum\limits_{f_2\in F(g_2)} d_{func}(f_1,f_2) & if\ \Gamma>0\\ 0 & if\ \Gamma=0\end{cases},$$

where

$$\Gamma=\#\{d_{func}(f_1,f_2)>0,\ \ \forall f_1\in F(g_1),\ f_2\in F(g_2)\}$$

counts the number of function pairs with non-zero distances and d_{func}(.,.) is the gene function distance.

2.4 The features of samples

First, in order to reduce the noise in microarray experiments and improve the signal, we average the left and right hemispheres by taking advantage of the inherent bilateral symmetry of the mice brain. Mice do not have "handedness" or speech-centers in the brain, which are known to be localized to one hemisphere in humans. In the process of averaging, for each row of the map, we average the framed cells, as shown in Figure 3. Then, we replace B1 with B11, A2 with A10, and the averaged gene expression map is obtained as shown in Figure 4.

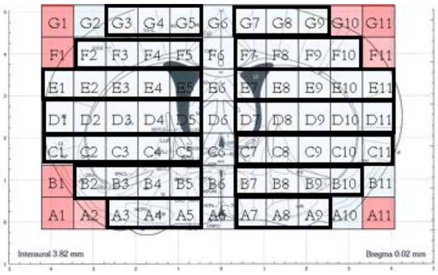
G1 G2 G3 G4 G5 G6 G7 G8 G9 G10 G11
F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 F11
E1 E2 E3 E4 E5 E6 E7 E8 E9 E10 E11
D1 D2 D3 D4 D5 D6 D7 D8 D9 D10 D11
C1 C2 C3 C4 C5 C6 C7 C8 C9 C10 C11
B1 B2 B3 B4 B5 B6 B7 B8 B9 B10 B11
A1 A2 A3 A4 A5 A6 A7 A8 A9 A10 A11
Interaural 3.82 mm
Bregma 0.02 mm

Figure 3 Averaging left and right hemispheres

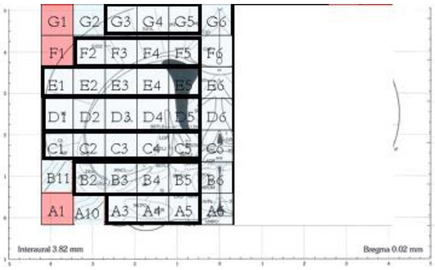

Figure 4 Averaged gene expression map

Next, we use the wavelet transform to extract features from the original expression values of each gene expression map. In the data set that we study, each row represents the gene expression values corresponding to the 68 voxels in the selected slice of mice brain. Intuitively, if an expression value is similar to other values in its spatial neighborhood, it is more reliable. However the original vectors of gene expression values ignore the spatial information. In order to measure the spatial consistency of expression values with others in their spatial neighborhood and to take into account the spatial factors of voxels in the brain map, we employ the wavelet transform to extract new features. We use the discrete wavelet transform (DWT) with single-level two-dimensional wavelet decomposition to extract features based on the gene expression matrix (see Figure 1). The Daubechies wavelet function is used in this study. The output of the wavelet transformation consists of approximation coefficients, which are the average of gene expression values in neighborhood voxels, and detail coefficients, which indicate the difference of each voxel from the average. For the averaged map of 6 by 7 cells, by employing multilevel 2-D wavelet decomposition at level 3, we obtain 42 coefficients (combining approximation and detail coefficients to approach the best results).

For each gene map, we concatenate its 42 wavelet coefficients and the 68 gene expression values, resulting in a descriptor of 110 dimension. Given two genes g_1 and g_2, let V_1 and V_2 be their feature vectors respectively. We derive the feature vector of the gene pair (g_1,g_2) as

$$|V_1 - V_2|.$$

Therefore, a gene pair sample can be represented as:

$$(|V_1\text{-}V_2|,\ d_F(g_1,g_2)) .$$

2.5 Identifying functional similarities of pair-wise samples by Boosting

2.5.1 Why boosting

Having the features and the samples ready, we need to choose a learning technique for our task. We face the challenge of dealing with a huge sample dataset. There are in total 31,066,903(=7883×7882/2) samples of gene pairs. Each sample has 110 features. The dataset is too large to be directly handled by many popular machine learning methods, such as the Support Vector Machine. Boosting [17], however, solves this problem because by loading and computing samples and features (weak learners) sequentially. Another advantage of using boosting is that it provides a way to investigate the roles of features in the learned classifier or regressor. In our particular task, this helps understanding the importance of each individual feature in predicting the gene similarities.

Since boosting is usually used to solve classification problems, we need to transform the regression problem to a classification problem by setting a threshold. The threshold is used to classify the continuous values of function distances into two classes: one that includes the samples (pairs of genes) with similar functions, and another that includes the samples with non-similar functions. So the classification problem with the continuous output in the range [-1,1] is transformed to a problem with two classes {-1, 1}, through a predefined threshold.

There are several variants of boosting algorithms that are widely used in the fields of data mining and pattern recognition. We choose AdaBoost [18] due to its excellent performances observed in many applications and its flexibility in weak classifier design. Intuitively, AdaBoost uses a weighted additive model to fit the training data. The model, which is named a *strong* classifier, is a weighted summation of a set of *weak* classifiers. The weight and weak classifiers are iteratively estimated or selected until convergence.

In our task, for an input feature vector V, a strong classifier denoted as H(V) is formulated as a combination of weak classifiers $h_1(V)$, $h_2(V)$, …, $h_K(V)$:

$$H(V) = \sum_{k=1}^{K} c_k h_k(V) ,$$

where c_k is the weight for the k-th weak classifier h_k. The task of the learning process is, in the k-th iteration, to either fit h_k or to pick h_k from a candidate set of weak classifiers. The fitting or selection is based on the classification performance on the training samples weighted by the current weights.

2.5.2 Designing the weak classifiers

One popular way of designing weak classifiers is to associate with each weak classifier a threshold to create a binary classifier, i.e., a stump function. In particular, for the i-th feature V(i) in our feature vector V and a threshold τ, a weak classifier has the form

$$h^{i,\tau}(V) = \begin{cases} 1 & if\ V(i) \le \tau \\ -1 & if\ V(i) < \tau \end{cases} .$$

The learning process is to find i, τ, and c_k for each one of the weak classifiers. In this case, a weak classifier is associated with only one feature. As a result, the weight c_k can be used to evaluate the importance of the feature in the strong classifier, i.e., the ultimate model used for prediction.

The binary classifier is very simple and easy to implement. However, for a complex learning task such as the one we are dealing with, more efficient weak classifiers often help with improving the learning and predicting efficiency by reducing the number of weak classifiers needed. In addition, in our study we have a huge set of training samples, which enables us to use better but more complex weak classifiers. Motivated by this observation, we extend the simple stump classifier by modeling the weak classifier with uniformly spaced bins. Specifically, our weak classifier for the i-th feature contains an *indicating* vector $L\in\{-1,1\}^M$, where M is a predefined number of bins. A classifier has the following form

$$h^{i,L}(V) = L(index(V(i))) ,$$

where index(V(i)) is the index of the bin V(i) falls into.

In the learning stage, the task at each iteration is to select the best feature i that estimates the indicating vector L. This is done by building a cumulated weight for each feature followed by a voting. The stump weak classifier can be viewed as a simplified case where M=2.

Figure 5 shows an example of a weak classifier learned from one of the features of the training data. It shows that the range of features is divided into small regions. The intervals of weak classifiers depend on the range of each feature (i.e. the max and min of values of the feature). We divide the range uniformly with fixed sizes. The label for each region is the sum of weighted labels of samples within the region. When the weak classifier is used to predict, the sample is assigned the label of the region in which this specific feature falls.

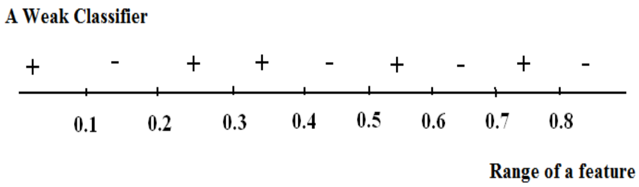

Figure 5: An example of proposed weak classifiers

2.5.3 Applying the method to MFEP specific subset

The relationship between the gene functions and gene expression maps does not hold for all genes but only for a set of certain genes. For this reason, we take advantage of the results obtained using *multiple functional expression profiles* (MFEPs) to perform the boosting analysis. For a given gene function or a set of gene functions, there might be a specific gene expression map (profile) associated with it. Genes that have similar gene expression maps to a specific profile may hold similar gene functions. We call this specific gene expression profile for a set of functions, Multiple Functional Expression Profile (MFEP). Genes associated to an MFEP have the same set of gene functions and also have very similar gene expression maps. The detected MFEPs can be used to predict gene functions with high accuracy [13]. In order to explore the strong relationship between gene functions and expression maps, we use a subset of genes instead of the whole dataset. This subset is created by calculating the expression features and labels of pairs of genes that are associated with the detected MFEPs, so we call it MFEP specific subset. Within this specific subset, there is supposed to be a regression relationship between the similarity of gene pairs (expression features) and gene function similarity (labels), which means that with the increasing similarities of pairs of gene expression maps, the gene functional similarities should be increased.

3. RESULTS

3.1 Identifying relationship between similarity of gene expression maps and their functions

First, we analyze the MFEP specific subset which contains all the genes associated with the MFEPs. We use the correlation coefficients between the 42 wavelet features to identify the similarity between gene expression maps. By the Matlab command *corrcoef*, we get two results: [R P]. R are the correlation coefficients between genes, and P are the p-values for the hypothesis of no correlation. R is taken as the similarity between gene maps to analyze the subset of genes within MFEPs. For a given interval of R, for example [0.1, 0.2], we select the set of samples falling into the interval and average the functional similarities of the samples in this set.

All 345 genes associated to the MFEPS are used in the experiment, resulting in 59,340 samples. The distribution of the correlation coefficients and the corresponding averaged functional similarities of samples are shown in Figure 6. The figure shows that when the similarities of gene maps are increasing, the function similarities are also increasing. The trend is very obvious for the samples with high correlation coefficients (larger than 0.6).

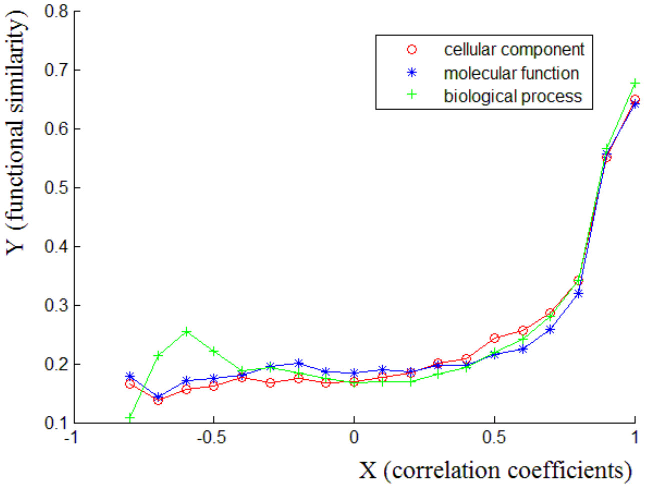

Figure 6: The distribution of correlation coefficients of gene maps similarity to functional similarity

X (-1:0.1:1) are the correlation coefficients for pairs of gene expression maps. Y is the averaged functional similarities of the samples whose correlation coefficients (X) are within a certain interval, for example, between [0.4, 0.5].

3.2 Boosting analysis on the MFEP specific subset

We conducted the boosting analysis on the MFEP specific subset of the 59,340 samples using our proposed weak classifiers and AdaBoost. The dataset was randomly split into two disjoint dataset: a training set (29,670 samples) and a test set (29,670 samples). The functional similarities between two genes are continuous values in the range of [0, 1], where “0” indicates no functional similarity between two genes and “1” indicates that the two genes have exactly same functions. In the experiment, we set a threshold 0.3 to cut the similarity values. If the value was larger than 0.3, we set the label to 1, otherwise we set the label to -1. With this threshold, there were 33.5% training samples that were assigned label 1, and 33.3% control (test) samples that were assigned label 1. The model was learned based on the training set, and was then used to predict the labels of samples in the test set.

There are totally 113 features for each sample in the experiment. In addition to the 42 wavelet features and the 68 original expression values, three new features were included: the correlation coefficient, the p-value of the correlation coefficients, and the Euclidean distance between pair-wise gene maps for each

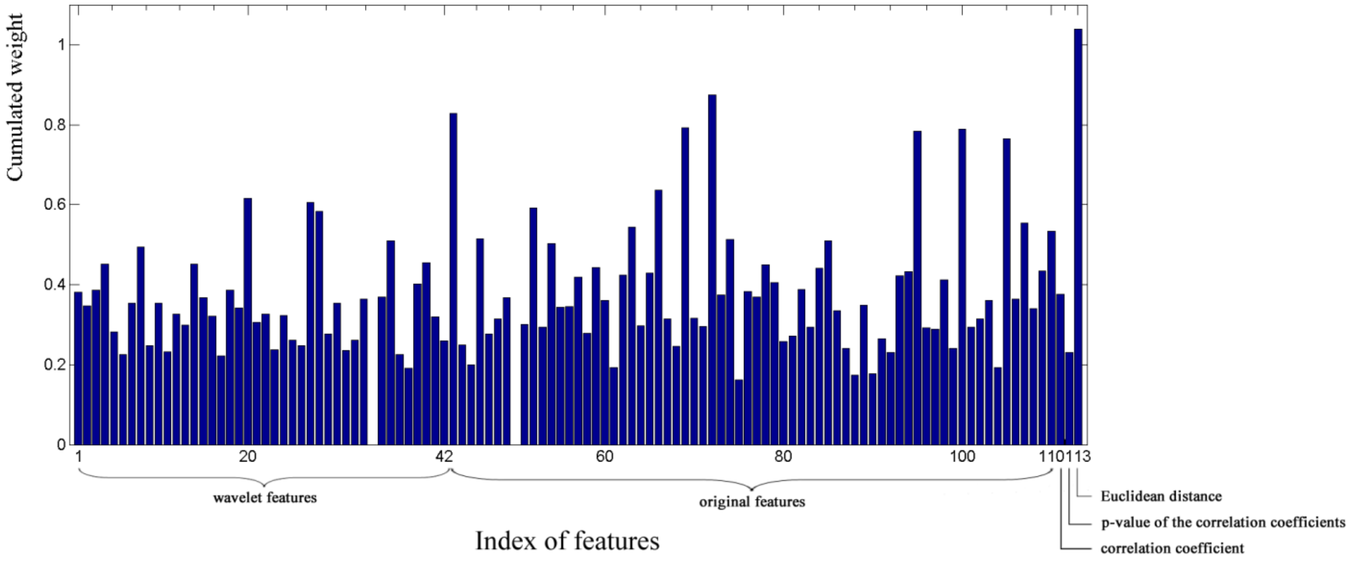

Figure 7 Cumulated weights of selected features

sample. Each p-value is the probability of getting a correlation as large as the observed value by random chance, when the true correlation is zero. If P(i,j) is small, e.g., less than 0.05, then the correlation R(i,j) is significant.

For the weak classifier, we choose 100 as the number of bins. For the AdaBoost algorithm, the boosting repeated for 4000 iterations to reach a stable performance on the prediction. With these settings, we got the prediction error on the training and control samples. The minimum error on training data is 22.03%, and the minimum error on control data is 30.77%. With the number of iterations performed, the error converged to a certain value. By changing the parameters, such as the number of regions and iterations, the error rate can vary.

Boosting selects the best feature (weak classifier) at each iteration and gives a weight to the feature. Figure 7 shows the cumulated weights of selected features over the 4000 iterations. The column of a certain features is the sum of the weights of the feature which are selected during the 4000 iterations. For example, if a feature is selected *m* times with weights w_1, w_2, …, w_m, the sum of weights of the feature is $\sum_{i=1}^{m} w_i$.

In the following we analyze the selected features separately.

Top selected features:

The top 10 selected features are: 113rd, 72nd, 43rd, 69th, 100th, 95th, 105th, 66th, 20th, and 27th features. We notice that the most selected feature is the Euclidean distance between pair-wise gene maps. Since the Euclidean distance directly reflects the appearance similarity of two gene maps, this observation strongly supports our conjecture that gene map similarities correlate closely with the gene functional similarities.

The original 68 expression values:

Because the 68 original features are gene expression values in the 68 voxels (Figure 1), we can visualize and locate these features in the mouse brain. These voxels are shown in Figure 8 as D1, A3, C9(C3), F8(F4), F3, G4, and C6. In the figure, the darker mark means that the voxel is selected more frequently (in terms of sum of weights) and that is more significant in predicting the functional similarity of genes from the gene expression maps.

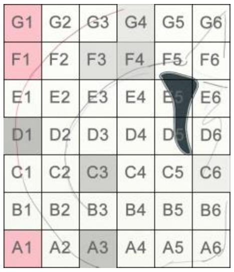

Figure 8 The most selected original voxels (better viewed in color)

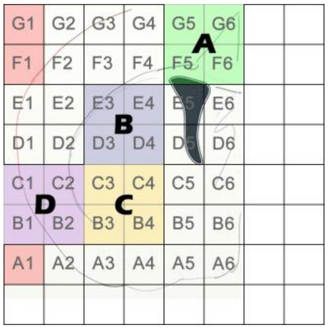

Figure 9 The most selected wavelet features (better viewed in color)

The 42 wavelet features:

The boosting experiment also selected wavelet features which are extracted from the averaged mouse brain. The top selected wavelet features are: the 20th, 27th, 28th, and 36th features. As figure 9 shows, the 20th feature is the horizontal detail coefficient extracted from area A, the 27th and 28th feature are the vertical detail coefficients extracted from area B and C, and the 36th feature is the diagonal detail coefficient extracted from area D.

3.3 Boosting analysis on the restricted subset

From the figure 7 we know that there are still noise weakening the relationship between functional similarities and correlation coefficients within the MFEP specific subset. So here we restrict the MFPE set to a more specific one in which the samples have the correlation coefficients bigger than 0.7, as the square in figure 10 shows.

Similarly, we apply the boosting analysis on the restricted subset of 612 samples using our proposed weak classifiers and AdaBoost. The dataset was randomly split into two disjoint dataset: a training set (441 samples) and a test set (171 samples). We set a threshold 0.67 to cut the similarity values, and there were 35.2% training samples that were assigned label 1, and 30.3% control (test) samples that were assigned label 1.

There are also totally 113 features for each sample in the experiment. For the weak classifier, we choose 20 as the number of bins. For the AdaBoost algorithm, we repeated 2000 iterations to reach the best performance of the prediction. With these settings, we got the minimum error on training data is 0%, and the minimum error on control data is 21.7%.

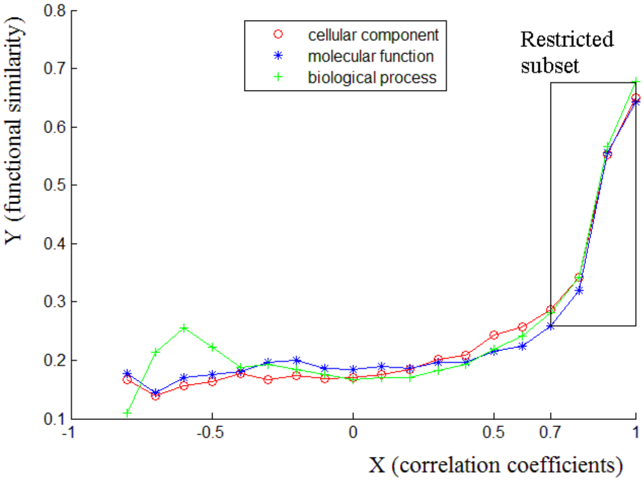

Figure 10 The restricted subset with correlation coefficients bigger than 0.7

Figure 11 shows the cumulated weight of selected features. The column of a certain feature is the sum of the weights of the feature which are selected during the 4000 iterations. For example, if a feature is selected *m* times with weights w_1, w_2, …, w_m, the sum of weights of the feature is $\sum_{i=1}^{m} w_i$.

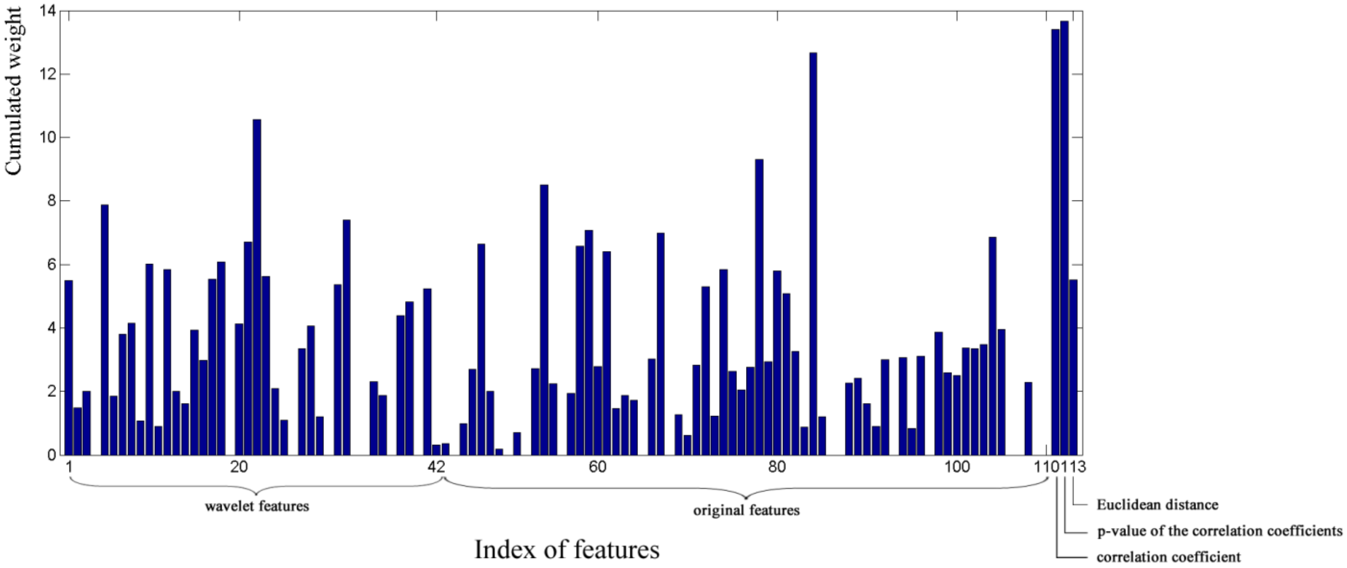

Figure 11 Cumulated weights of selected features

Top selected features:

The top 10 selected features are: the 111st, 104th, 76th, 54th, 35th, 13rd, 82nd, 78th, 86th, and 60th features. We notice that the most selected feature is the correlation coefficient between pair-wise gene maps.

The original 68 expression values:

The top 10 selected original features (voxels) are shown in Figure 13 as G3, D5, B5, D11(D1), D7(D5), E4, B11(B1), F9(F3), C10(C2), and D2. In figure 12, the darker mark means that the voxel is selected more frequently.

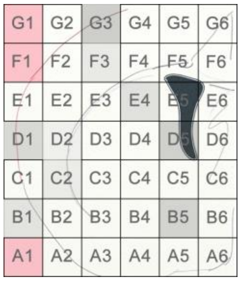

Figure 12 The most selected original voxels (better viewed in color)

The 42 wavelet features:

The top selected wavelet features are: the 35th, 13rd, 20th, 23rd, 15th, 3rd, and 21st features. As figure 13 shows, the 35th feature is the diagonal detail coefficient extracted from area A, 13rd, 20th, 15th features are horizontal detail coefficient extracted from area B, C, D, and 23rd feature is the vertical detail coefficient extracted from area A.

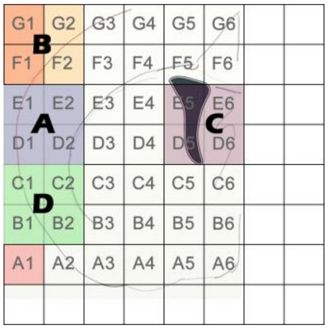

Figure 13 The most selected wavelet features (better viewed in color)

4. DISCUSSION

In this study, we identify the pair-wise gene functional similarities by multiplex gene expression maps. This is based on the hypothesis that genes with similar gene expression maps share similar gene functions. This hypothesis was confirmed for a number of genes in our previous analysis [12]. Since the original dataset only contained the gene expression maps, it was hard to use supervised learning to analyze the data, so, instead, we built a new dataset in which each sample represented a pair of genes. The features for these samples were the similarity or distance values between two gene expression maps, and the labels were the functional similarities between genes. The wavelet transformation was used to extract features from the original expression values of the averaged hemispheres of the mouse brain. We used the absolute difference between each pair of features of the two genes. In addition, the correlation coefficients, the p-value of the correlation coefficients and the Euclidean distance were included in the calculation of the difference between gene expression maps. We define the functional similarities of two gens were the averaged function distances for each pair of functions included in the two genes. The similarity (distance) of two gene function was obtained by Lin's method based on GO structures. We also built the MFEP specific subset by multiple functional profiles so that the genes in the subset had strong relationship between gene functions and gene expression maps. Based on the MFEP specific subsets, we applied AdaBoost and propose our own weak classifier to fit the characteristics of the dataset. We further restricted the dataset to a more specific one and tested our proposed methods on this restricted subset.

From the experiment on identifying the relationship between similarity of gene expression maps and functional similarity, we observed that with increasing similarities of gene expression maps, the pair-wise genes' functional similarities were also increased, especially for samples with correlation coefficients between pairs of gene maps larger than 0.8. From the boosting analysis, we were able to predict functional similarities of pairs of genes with about 80% accuracy (20% error rate) on the restricted MFEP specific subset. By the proposed methods, the similarity of pairs MFEP gene expression maps can be used to estimate the pairs of genes have similar gene functions or not. Therefore, this method could be used to predict an unknown gene have similar functions to a given known gene or not.

The selected weak classifiers were able to identify the features that are more important for the prediction. By checking the most selected original features and wavelet features we were able to locate the significant voxels and area in the mouse brain. The most selected voxels generally corresponded to the salient neuroanatomical features of the analyzed brain slice. For example, in Figures 8 and 12, the most selected voxels correspond to cortex and striatum. The top selected wavelet features in Figures 9 and 13 also feature cortex and striatum. These observations are consistent with the major molecular and anatomical features of the brain slice.

In the current study, the samples were divided into two classes in accordance to our binary classification formulation. In the future, we plan to use finer split of the samples (e.g., four or more classes) to improve the precision and finally model the problem as a regression problem. There are many linear and non-linear regression algorithms (especially the online versions) that can potentially handle large amounts of training data. In the future, we will try different regularizers besides of boosting, such that there is no need to make arbitrary thresholds of labels.

Furthermore, since the Euclidean distance between wavelet representations may be insufficient to capture non-linearity in the complicated gene map-to-gene function relationship, we would like to investigate other information that is not captured by the wavelet representation. We also plan to incorporate other features besides the wavelet features into the analysis to further improve the results.

5. ACKNOWLEDGMENTS

This work was supported in part by the National Science Foundation under grants IIS-0705215, IIS-0237921 and by the National Institutes of Health under grant R01 MH68066-05 funded by NIMH, NINDS and NIA, and NINDS grant 1 R01 NS050148. This project is also funded in part under a grant to Z.O. with the Pennsylvania Department of Health. The funding agencies specifically disclaim responsibility for any analyses, interpretations and conclusions.

6. REFERENCES

[1] The Gene Ontology Consortium, *Creating the gene ontology resource: Design and implementation*, Genome Research, vol. 11, pp. 1425-1433, 2001.

[2] Francisco M. Couto, Mário J. Silva, Pedro M. Coutinho, *Measuring semantic similarity between Gene Ontology terms*, Data & Knowledge Engineering, v.61 n.1, p.137-152, April, 2007

[3] Spiridon C. Denaxas, Christos Tjortjis, A *go-driven semantic similarity measure for quantifying the biological relatedness of gene products*, Intelligent Decision Technologies, v.3 n.4, p.239-248, December 2009

[4] Haiying Wang, Francisco Azuaje, Olivier Bodenreider, Joaquín Dopazo, *Gene Expression Correlation and Gene Ontology-Based Similarity: An Assessment of Quantitative Relationships*, Computational Intelligence in Bioinformatics and Computational Biology, 2004, p.25-31, 7-8 Oct, 2004

[5] Duygu Ucar, Fatih Altiparmak, Hakan Ferhatosmanoglu, Srinivasan Parthasarathy, *Mutual Information Based Extrinsic Similarity for Microarray Analysis*, Proceedings of the 1st International Conference on Bioinformatics and Computational Biology, April 08-10, 2009, New Orleans, LA

[6] Torsten Schön , Alexey Tsymbal , Martin Huber, *Gene-pair representation and incorporation of GO-based semantic similarity into classification of gene expression data*, Proceedings of the 7th international conference on Rough sets and current trends in computing, June 28-30, 2010, Warsaw, Poland

[7] Jose L. Sevilla, Victor Segura, Adam Podhorski, Elizabeth Guruceaga, Jose M. Mato, Luis A. Martinez-Cruz, Fernando J. Corrales, *Angel Rubio, Correlation between Gene Expression and GO Semantic Similarity*, IEEE/ACM Transactions on Computational Biology and Bioinformatics, Volume 2 Issue 4, October 2005

[8] Lord PW, Stevens RD, Brass A, Goble CA, *Investigating semantic similarity measures across the Gene Ontology: the relationship between sequence and annotation*, Bioinformatics, 2003 Jul 1, 19(10):1275-83

[9] Ram P. Singh, Vanessa M. Brown, Abhijit Chaudhari, Arshad H. Khan, Alex Ossadtchi, Daniel M. Sforza, A. Ken Meadors, Simon R. Cherry, Richard M. Leahy, Desmond J. Smith, *High-resolution voxelation mapping of human and rodent brain gene expression*. J Neurosci Methods, 2003. 125(1-2): p. 93-101.

[10] Dahai Liu and Desmond J. Smith, *Voxelation and gene expression tomography for the acquisition of 3-D gene expression maps in the brain*, Methods, Volume 31, Issue 4, 2003, p. 317-325.

[11] Mark H. Chin, Alex B. Geng, Arshad H. Khan, Wei-Jun Qian, Vladislav A. Petyuk, Jyl Boline, Shawn Levy, Arthur W. Toga, Richard. Smith, Richard M. Leahy, and Desmond J. Smith. *A genome-scale map of expression for a mouse brain section obtained using Voxelation*, Physiol, Genomics 30: p. 313-321. 2007.

[12] Li An, Hongbo Xie, Mark H. Chin, Zoran Obradovic, Desmond J. Smith, and Vasileios Megalooikonomou. *Analysis of multiplex gene expression maps obtained by voxelation*, BMC Bioinformatics. 2009; 10(Suppl 4): S10.

[13] Li An, Hongbo Xie, Zoran Obradovic, Desmond J. Smith, Vasileios Megalooikonomou, *Identifying Gene Functions using Functional Expression Profiles obtained by Voxelation*, Proceedings of the First ACM International Conference on Bioinformatics and Computational Biology 2010, pp. 188-197.

[14] Maximilian Diehn, Gavin Sherlock, Gail Binkley, Heng Jin, John C. Matese, Tina Hernandez-Boussard, Christian A. Rees, J. Michael Cherry, David Botstein, Patrick O. Brown and Ash A. Alizadeh, *SOURCE: a unified genomic resource of functional annotations, ontologies, and gene expression data*, Nucleic Acids Res. 2003 Jan 1;31(1):219-23.

[15] Stanford Genomic Resources, http://genome-www.stanford.edu/.

[16] Dekang Lin, *An Information-Theoretic Definition of Similarity. Proceedings of the Fifteenth International Conference on Machine Learning*, Madison, Wisconsin, p. 296-304, 1998

[17] Robert E. Schapire, *The Boosting Approach to Machine Learning An Overview*, Nonlinear Estimation and Classification, Springer, 2003.

[18] Jerome Friedman, Trevor Hastie, and Robert Tibshiran, *Additive logistic regression: a statistical view of boosting*, Ann. Statist, Volume 28, Number 2 (2000), 337-407.