Incorporating Known Pathways into Gene Clustering Algorithms for Genetic Expression Data

Ryan Atallah, John Ryan, David Aeschlimann

December 14, 2013

Abstract

In this project, we study the problem of classifying genes into biologically meaningful clusters by applying machine learning techniques to gene expression data. Our data consists of gene expression data from two types of mouse immunological cells (T-cells and Granulocytes) as well as data that summarizes known biological pathways and genetic associations in mice. We utilize this data about previously studied and understood genetic associations to improve the clustering classifications that are made based on the gene expression data alone.

1 Introduction

The applications of gene expression data are numerous, as information about the expression of individual genes can inform both academic understanding of genetic diseases as well as clinical treatments for patients. One interesting use of gene expression data is to "cluster" genes together in a biologically meaningful way. That is, given gene expression data, we wish to group genes into clusters that correspond to biological pathways in which the grouped genes code for proteins that work together at a functional level.

This is a problem that has been studied, as in [1]. One may apply a k-means algorithm to a matrix containing gene expression data to effectively cluster the genes. When attempting to cluster genes, however, often times a lot of information is already known about genetic associations in biological pathways. The ability to leverage this data to improve clustering predictions that would be made on the gene expression data alone is the challenge that we address in this paper.

2 Methods

We studied gene expression data that was extracted from two different immunological cell types from 39 different mice. mRNA levels for 25,134 genes were measured using microarrays. This information was stored in a $25,134 \times 39$ matrix $D = (d_{ij})$ where $d_{ij} \in [0, 15]$ is a measure of expression level (a higher value means higher expression level). Further, we accumulated data that summarized 13,476 known biological pathways. This data was stored in a $25,134 \times 13,476$ matrix $P = (p_{ij})$ defined thus:

$$p_{ij} = \begin{cases} n & n \in \mathbb{N} \text{ and } n > 0, \text{ if gene } i \text{ is associated with pathway } j \text{ with strength } n \\ 0 & \text{else} \end{cases}$$

We ran a k-means clustering algorithms on the matrix D for various values of k and used the elbow method to decide on an optimal k.

2.1 Finding the optimal k

We theorized that there exists a "correct" clustering of the genes considered, and therefore our chosen k should be the same for both the original data and the data augmented with the data about priorly determined pathways. Our approach was then to pick a value of k that worked well for both data sets and compare the clusters produced by running k-means on each set and get an idea of the utility of augmenting gene expression data with pathway priors.

We decided to pick an optimal k using the elbow method. The elbow method, which can be traced to speculation by Thorndike (1953) in [2], describes a good choice of k for k-means. The method can be described as follows: for fixed $k \in \mathbb{Z}^+$ let G denote the set of all training examples and C_k the set of cardinality k of centroids determined by the k-means algorithm. Now for $x \in G$ define $\mu_x = \arg\min_{\mu \in S_k} \|x - \mu\|^2$ where $\|\cdot\|$ denotes the Euclidean norm on $\mathbb{R}^n$. Then define the error of the k-means algorithm to be

$$\varepsilon_k = \sum_{x \in G} \|x - \mu_x\|^2$$

Now make a plot of ε_k as a function of k and note where there are "elbows" in the plot, i.e., points that mark a sharp change in ε_k. An elbow point is a good choice of k. We also performed k-means on the $25,134 \times 13,515$ matrix M that is just D and P concatenated. Performing the elbow method on both our results from running k-means on M and D, we picked an optimal k for clustering.

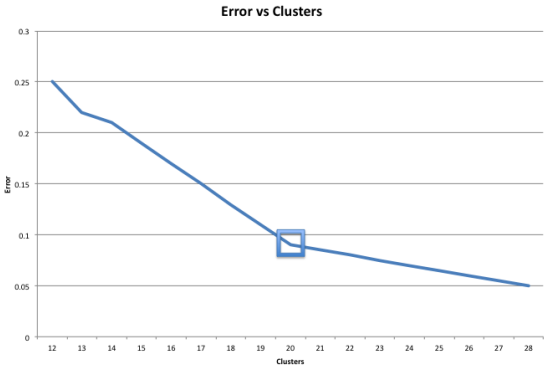


Figure 1: Example of elbow method

Then, we compared the results of k-means on M versus the results of k-means on D by comparing the similarity between the clusters determined by both methods, using the following method:

2.2 Choosing the Optimal Weight

Our strategy for incorporating the prior data is to prepend the gene expressions data matrix D to the genetic pathways matrix P multiplied by a scalar λ, which corresponds to how much we want to "weight" the information about known pathways. We consider each genetic pathway as an addition feature to a gene data point. The result is a matrix in $\mathbb{R}^{m \times n}$, where $m = 25{,}134$, $n = a + b$, $a = 13{,}476$ and $b = 39$. We constructed a finite set $S \subseteq \mathbb{R}$ of potential candidates for λ, ran k-means on the concatenated matrices with different choices of $\lambda \in S$ and compared the results to find the optimal choice for λ.

To pick our initial value λ' in S, we scaled P such that it has approximately equal weight relative to D. This can be computed by normalizing D to be between 0 and 1, and scaling P as follows:

$$\lambda' = \frac{b}{a(argmax_p P)}$$

Then, we selected our set S of λs to test by picking an even number of values greater than and smaller than λ', such that each λ increments evenly and begins at zero.

Thus, our equation for λ_i for the set $S = \{\lambda_1, \lambda_2 ... \lambda_r\}$ is

$$\lambda_i = \frac{b(ri)}{2a(argmax_p P)}$$

Then, to test each selection of λ, we used the two-fold cross validation method. In a typical supervised learning problem, cross-validation is used to test a trained model on classifying a subset of the original data. We can make a normative judgement about the quality of our unsupervised learning model by using distance as our metric of comparison, as opposed to label classification. The choice of λ that produces the clusters that are most tightly clustered with randomly selected test data may be the optimal choice of λ. We can conduct this analysis as follows:

1. Divide each matrix D and P into train and test (validate) matrices by randomly splitting up the genes into two groups D_t, D_v, P_t, and P_v such that D_t and P_t have 70% the rows of D and P, and D_v and P_v have the remaining 30%.
2. For each value $\lambda_i \in S$, compute the horizontal concatenation matrices $X_i t = [D_t \ \lambda(P_t)]$ and $X_i v = [D_v \ \lambda(P_v)]$.
3. Run the k-means clustering algorithm on $X_i t$ with the optimal selection of k from Section 2.1 to produce the result vector C_i of centroid locations.
4. Then, evaluate the clustering model by testing on the remaining 30% validation data. For each row x_j in $X_i v$, assign a cluster by finding the closest cluster c_j in C.

   $$c_j = argmin_c \sum_{d=1}^{k} ||x_j - c_d||$$

   The error value ε_i is computed by summing the distances between each point x_j and their corresponding cluster centroid c_j.

   $$\varepsilon_i = \sum_{j=1}^{m_v} \frac{1}{d_j} x_j - c_j$$

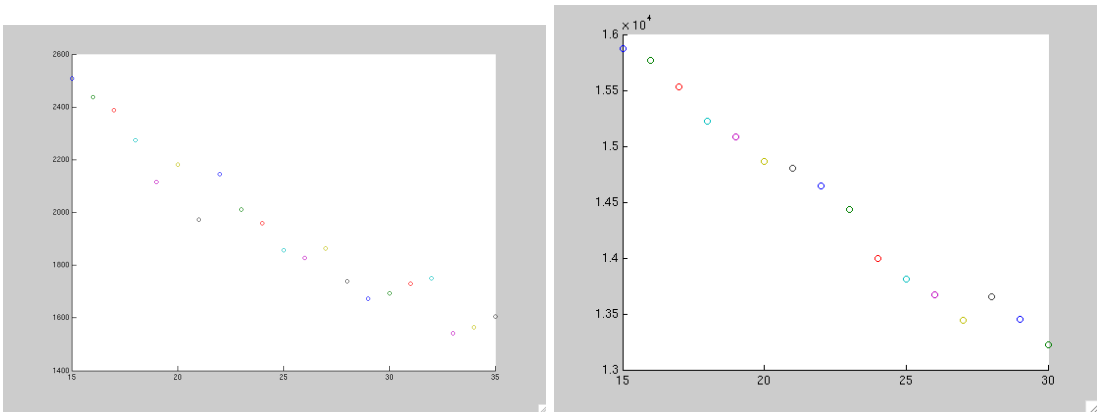
   where d_j is the number of points assigned to cluster c_j.
5. Pick the λ_i with the lowest classification error ε_i.

To smooth out inconsistencies and ensure that our conclusion was more generalizable, we decided to run this algorithm multiple times and compare the averages of each ε_i for our analysis. This helps reduce the effects of the random initial assignment of cluster positions when running the k-means algorithm, and local extrema in the training data set X_v.

3 Data

Because it is incredibly computationally expensive to run the k-means algorithm on all of our data set with thousands of features and training sets, we chose to run k-means only on the first 1,000 genes as a proof of concept of our method.

We ran the k-means algorithm first on the gene expression data D on varying values of k and plotted the results of the error function J to determine the location of the elbow points that mark the optimal choice of k. We made the assumption that the optimal choice of k would be something close to the value $k^* = \sqrt{\frac{n}{2}}$, as suggested by Mardia et al. in [3]. Then, we chose values of k that deviated from k^* by ± 10. Thus, our choices of k ranged from 10 to 30. This produced the graph shown in Figure 2a. We repeated the same procedure with the gene expression data agumented with pathway priors M. The plot of our k-means error values for each value of k on M can be seen in Figure 2b.



(a) Error of k-means run with varying k on expression data D.

(b) Error of k-means run with varying k on the augmented data M.

Figure 2: The elbow method for determining an optimal k.

From this analysis, we determined that the optimal choice of k was aproximately 24 due to the fact that in both Figures 2a and 2b, there was a sharp change in the slope of the plot of k-means error values around the point where $k = 24$.

We then compared the results of k-means on M versus the results of k-means on D by comparing the similarity between the clusters determined by both methods where for both $k = 24$ using the algorithm described in Section 2.1. We determined that selecting $k = 24$ gave us a reasonable error.

We then ran our 2-fold cross validation algorithm to find the optimal choice of λ in the concatenation of D and P. We first trained a cluster model on 70% of our original data set, and tested the resulting clusters on the remaining 30% to produce an error value for each λ_i as described in Section 2.2.

After running our validation algorithm ten times, taking the average of each trial's ε_i values, we were able to identify the optimal $\varepsilon* = \varepsilon_5$, as shown in Figure 3. Thus, we could select the optimal scalar

$$\lambda* = \lambda_5 = \frac{5(br)}{2a(argmax_p P)} = 0.0012.$$

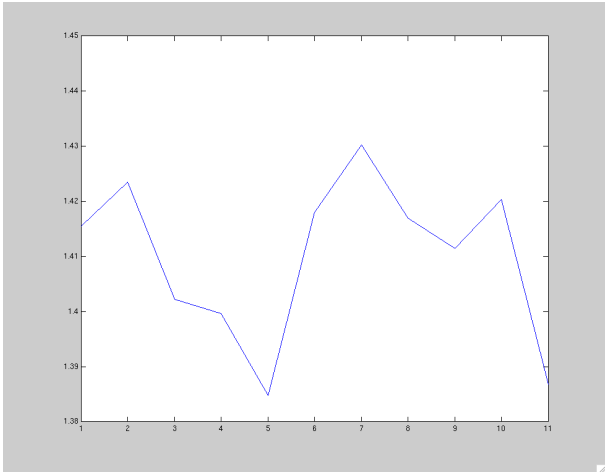


Figure 3: Validation error of each λ_i, averaged over 10 trials

4 Further Research

The analysis we have done thus far has been limited due to the computational load of running our algorithms on large data sets. Our next steps are to run our analysis on all 25,134 genes over more varied choices of k to get a more accurate clustering and more optimal choice of k.

Once we have this choice of k, we would evaluate it with the following method:

1. Enumerate all possible bijective mappings between the clusters trained on the data and the clusters trained on the data augmented with priors.
2. Sort these mappings by the number of points shared by each.
3. Go through and accept these mappings in sorted order. If a given mapping has a cluster already accepted, reject that mapping and move on.
4. Find the total number of points shared by each mapping pair c found via this method. Subtract this value from the total number of points k and divide by the total number of points: $\varepsilon = \frac{k-c}{k}$

Furthermore, we would like to extend our methodology to explore more strategies for incorporating the prior data into our gene expression data for clustering analysis. Thus far, we have only tried concatenating the pathways data to our gene expression data unweighted. We would like to explore weighting the pathways data other strategies.

5 References

[1] D. B. Fogel, "An introduction to simulated evolutionary optimization," IEEE Trans. Neural Networks, vol. 5, no. 1, pp. 3–14, 1994.

[2] Robert L. Thorndike (December 1953). "Who Belong in the Family?". Psychometrika 18 (4): 267–276.

[3] Kanti Mardia et al. (1979). Multivariate Analysis. Academic Press.