**Paper notes on class discovery**

**ArrayCGH-based classification of neuroblastoma into genomic subgroups (Michels et al. 2007)**

**Chromosomal and MicroRNA Expression Patterns Reveal Biologically Distinct Subgroups of 11q− Neuroblastoma (Buckley et al. 2010)**

**Model-based clustering of array CGH data (Sha et al. Bioinformatics 2009)**

**A 6-gene signature identifies four molecular subgroups of neuroblastoma (Abel et al. 2011)**

NB is a childhood tumor. This is the most common infants cancer. NB prognosis is dependent on: (1) progression, (2) age at diagnosis, (3) tumor histopathology, and (4) genetic factors: MYCN amplification (MNA) and DNA index. Generally, children diagnosed before 18 months with a localized tumor have favorable outcome. Older children with metastatic cancer show poor prognosis and have low survival rate. MNA is known to be associated with fast progression rate and poor progression despite age and stage. Thus, MNA is a main factor for risk stratification. However, most metastatic tumors do not show MNA, and other chromosomal aberrations being evaluated.

Previous studies partitioned sporadic NB tumors to three main classes based on the genetic profile of the tumor:

1. Type 1: low risk tumors with triploid DNA, numeric alterations and high expression of nerve growth factor TrkA.
2. Type 2A: intermediate risk with 11q deletion, 17q gain and no MNA.
3. Type 2B: high risk with MNA, 17q gain and 1p deletion.

There are many intermediate tumors that cannot be assigned to one of these clusters. For these tumors it is harder to predict the outcome.

Gene expression experiments on NB and NB subtypes:

1. De Preter et al (ref 14) built a 132 gene classifier for discriminating the subtypes.
2. Other studies reported predictive gene signatures (refs 14-20), while others focused on subclass classification and differential genes between these subtypes (refs 21-29).
3. Genes that are related to NB pathogenesis: MYCN (ref 30), ALK (ref 31), and PHOX2B(ref 32). MNA occurs in aggressive metastatic tumors, activating mutations or amplifying ALK (~7% of the tumors, refs 34-37). PHOX2B is mutated mainly in familial cases and in small percent of sporadic patients (refs 32, 38).

The clustering method was based on PCA analysis. The lowest variation genes were removed until a good grouping was defined by the PCA (how exactly?). In the PCA space, the clustering was done by stretching an edge from each node to its nearest neighbor and using the connected components of this graph. We shall call this method PCA-NN.

Datasets and preprocessing:

1. De peter dataset: 23 NB tumors, U133A arrays (Affymetrix).
2. McArdle and Wilzen: 30 NB tumors, 133A arrays.
3. Wang data: 101 NB tumors and one brain sample, HGU95Av2 (Affymetrix).

Preprocessing: gcRMA, low detection probes removal (max base-log < 6), and merging probes by gene ids. It leaved 7439 genes in dataset 1, 8106 in dataset 2 and 7542 in dataset3.

Results

The variation filter left 414 genes in dataset 1 and 716 in dataset 2. The intersection size between the datasets is 226(!). The low variation probes were removed until a clear separation between the groups was observed (overfitting?). The grouping was validated using LOOCV (not explained how exactly, and if it was on the selected gene groups than this is clearly overfitting of the data). The clustering was validated using LOOCV. Except few exceptions the clustering was stable (exceptions: 3 samples in dataset 1 and 2 in dataset 2). The final clusters are called p1-p4.

Enrichment analysis based on the prognosis features show: (1) MNA and del1p in p3, (2) late stage, 17q gain, and poor outcome in p2-p4 as compared to p1, (3) 11q deletion in p2 and p4 (14 out of 9 patients in the two datasets).

Six NB genes were selected based on biological relevance: predisposition (susceptibility, sickness promotion) genes ALK and PHOX2B, NB amplified genes MYCN and CCND1 (in ~20-30%, and ~5% of sporadic cases), differential NB genes NTRK1 and BIRC5. All six genes are differential when running ANOVA on p1-p4. Based on fold-change between each group and all other groups we get:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Gene | P1 | P2 | P3 | P4 |
| ALK | Down | Up | Up | Down |
| BIRC5 | Down | Up | Up | Down |
| MYCN |  |  | Up | Down |
| NTRK1 | Up |  | Down | Down |
| PHOX2B |  |  |  | Down |
| CCND1 |  |  |  | Down |

KM plots of the groups based on clinical data from dataset 1 and dataset 2:



Explanation: OS overall survival, EFS event free survival. Censored sample: a patient that still leaves or the monitoring was stopped. The p-value is calculated using the Mantel-Cox test. This is a nonparametric test for multiple distributions. The intuition: when we have only two lines, the test is a generalization on the Wilcoxon test to include censored patients.

The p1-p4 partition was refined to better fit the differential expression of the six NB genes. Then they used the new partition, r1-r4, as input to SAM. Combining the results of both datasets 1 and 2 yielded a signature of 98 genes of which 74 genes appeared also in the third data set (Wang data). These 74 genes were used for PCA-NN and hierarchical clustering analysis. Based on the hierarchical clustering, 4 groups were detected manually (un-clustered samples were assigned to clusters using the NN rule of the PCA method). The results (Figure 3 shows the heatmap):

|  |  |  |  |
| --- | --- | --- | --- |
| Dataset 3 cluster | Genetic marks | Prognosis | Datasets 1 and 2 fitting clusters |
| H1 |  | Good | P1 |
| H2 | Del11q, no MNA, 17q gain | Bad, Stage 3-4 | P2 |
| H3 | MNA, del1p, 17q gain | Worst, stage 4 | P3 |
| H4 | Del11q, no MNA, 17q gain | Bad, Stage 3-4 | P4 |

Table 4 from the paper summarized the p- vs. h- vs. known subtypes comparison:



In the supplementary: using PCA of global transcript of one dataset on the others. The results seem marginal, no quantitative score.

Figure 4 shows that the 6-gene signature detects the subtypes of all datasets.

Methods

Literature analysis: they went over 15 studies of NB gene expression studies. These together reported 1012 genes. 212 genes appear in at least two studies, 157 of them appear in all three datasets analyzed in this study. 30 genes also appear in MeSH as associated with NB. Six genes were then selected based on "biological relevance".

Statistical tests:

1. The PCA-NN method detected four groups. The patients of these groups were analyzed by their prognostic markers: stage, del1p, gain 17q, and MNA. The enrichment score was Fisher's exact test. Results are shown in Table 1.
2. The six genes were analyzed using ANOVA to test changes between the four subgroups. It seems they did more tests and combined the results (probably to get good results…).
3. SAM detected 98 genes differential between the detected subgroups (of the PCA-NN). The procedure: first refine the p-groups (detected by the PCA-NN) to fit the r-groups (detected by looking at the expression levels of the six NB genes). Then, they ran SAM on datasets 1 and 2 to get the top 4000 genes in each dataset (why?). They then filtered out genes with more than one probe (why?), and selected the top 98 genes according to fold change (how? This is a multiclass dataset). 74 genes appeared also in the third dataset. These genes were used for hierarchical clustering of the third dataset.

In table 4, we can find a comparison between the known subgroups (1, 2A, 2B) and the four groups detected in this study. Of note, the subgroups in these studies were not defined by the original authors of the studies. They were defined by the authors of this study based on prognosis features.

Summary: the analysis of datasets 1 and 2 is supervised and might overfit, but the validation on dataset 3 is reasonable. The main concern is that most validation seems to be highly twisted by manual fitting and no real quantitative assessment. Nevertheless, the detection of clinically similar 4 groups in all datasets is an appealing result. The detection of a new subgroup is innovative for this field.

**Class cover catch digraphs for latent class discovery in gene expression monitoring by DNA microarrays (Priebe et al. 2003)**

The input is two sets of observations over , and . We define class cover catch digraph (cccd) for against . Let . For each define . That is, is the maximal radius around such that the ball around contains only members from the same class of . We denote the radius of as . We add an edge to if .

We now calculate the (approximate) minimum dominating set of , denoted as . This set of observations is then clustered according to the values. This is done using hierarchical clustering (we have one feature for each sample here). For each k, we can horizontally cut the dendogram to get k clusters. To decide which k to use we can plot the empirical risk as a function of k. Assume that we analyze the class X. The empirical risk of a clustering solution is defined as the sum of samples that are not covered by the clustering plus the number of y samples erroneously covered by one of the clusters. By definition, a cluster covers the union of areas defined by its samples and their radius. The formal definition of this risk is:



To select k we define a dimensionally-penalized risk: . Thus we select the minimal k such that the penalized risk is minimized. By construction we get for , and for . Intuitively, we are looking at the the risk as a function of k and select k that is at the "elbow: of the curve. The parameter defines sharpness required for the "elbow". Once k is selected, we can assign each class sample to the clusters that cover it (see definition above).

Formally: . Another intuition is that we cluster the class samples based on their distances to samples from other classes (see Figure 1).

**Class discovery in gene expression data (Amir Ben Dor, Nir Friedman, Zohar Yakhini 2001)**

This paper deals with unsupervised inference of subclass in gene expression data. Shortcomings of clustering include: (1) only small subset of the genes differentiate between subtypes, and it require preprocessing of the genes; (2) we want fuzzy clustering and not a partition of the samples.

The input to the problem is a set of expression profiles . Each is a biological sample and N is the number of genes. A labeling vector assigns 1 (positive), -1 (negative) or 0 (control) for each sample. There are many statistics to measure the relevance of a gene to the labeling. Here the authors use TNoM (ref 3).

TNoM score

Given samples from two classes A and B, we want to measure if a gene g discriminates between the classes. We rank the expression values of g to get a ranking. We now define a vector . The TNoM score of the gene corresponds to the minimal number of misclassifications (check this) when a classifier is built based on the ranking .

The p-value of the TNoM score can be evaluated precisely. Let be the score of the gene g, under the real labeling. The p-value is defined as the probability to get a score at most s under random labeling.

Surprise scores

The idea here is that the overabundance of TNoM scores (as seen in figure 1, there are more significant genes, according to the TNoM score, than expected) can be used as a means to measure a partition of the samples.

Formally, let be the number of in the labeling vector (i.e., is the number of samples labeled -1). Let . For a score we define a Bernulli random variable that receives 1 the score of a gene is at most in a labeling . We now have random variables on . Given that genes had a score of at most in a labeling , we can ask how rare is to get this when .

Max-surprise score

We assume that the indicator random variables are independent. Using this assumption we get that the number follows a binomial distribution . So we can use the p-value of this distribution to measure a labeling . The final score is called Max-Surprise and it is defined as:

That is, the s that yields the best p-value.

Sanov score

The Max-Surprise score ignores the distribution of the scores (it takes the best ). We now use a multivariate distribution for this problem. Let be the set of scores (for a labeling). This set follows a multinomial distribution (under the null hypothesis and the independence assumption) in which. The frequency of the score in the data is. Under the null hypothesis, the probability of drawing a sample of scores is:

Here, is the probability of a score under the null hypothesis. This defines a vector of probabilities. The denotes the number of times the score was drawn. The probability can be approximated using the following theorem:

.

.

Thus, the value can be used as approximation for the desired probability. Also, this probability can be used to derive a p-value (*Sanov's Theorem*, ref 7):

.

 Note that we ask what is the probability to get types more skewed than howis.

This leads us to the Sanov-Surprise score: .

 Where is the distribution of genes scores under the labeling.

Classification scores

This is an alternative approach to measure the deviation of the scores of a labeling. The intuition here is to seek classes that are predictable based on the gene expression data. We shall use a simple feature selection and the naïve Bayes classifier. The selected features are those with good TNoM scores and the threshold is the one used in the Max-Surprise score.

Heuristics

We define the search graph. In this graph the nodes are the labeling options. Two nodes are neighbors if they agree on all but one labels. Thus, each node has neighbors. To search the graph we use simulated annealing. Reminder: in SA we have a temperature parameter t that is updated using an exponentially decreasing function (i.e., where and d is an integer). When we analyze a neighbor with a score of the current node that has a score , we move to it with probability .

Since we want to finds several, possibly overlapping, classes each time classes are found the genes that explain them are remove from the data. The process is then repeated to find other classes. Once no new classes are found, the labeling vectors found in the runs are evaluated on the entire data.

Table 1 shows the results on three real gene expression datasets. In two cases new labeling that was found had better LOOCV accuracy than the real labeling of the data. In some cases the p-value of the Max-Surprise score is not significant. Some of the p-values of the real labeling probably do not pass correction.

**Optimal Subclass Discovery for Discriminant Analysis (Zhu, 2004, IEEE)**

Introduction

LDA is an algorithm for classification and dimensionality reduction. This method attempts to minimize the Bayesian error by selecting meaningful features. The selected feature set is selected such that is maximized, where measures the variance between the class means and measures the variance within the classes (variance of the samples, see summary from the book in the next section). When the assumptions of the DA algorithm are not guaranteed, even unsupervised methods such as PCA can outperform DA (refs 14, 3, 18, 9). A simple example is when a class cannot be modeled by one Gaussian (Figure 1).

This paper shows that these problems of DA algorithms can be solved by dividing the classes to an adequate number of subclasses. Once these subclasses are defined, dimension reduction is done using the basis vectors of the covariance matrix of the subclass mean vectors. The new between-classes scatter matrix is defined as:

Where H is the total number of subclasses, is the prior and is the mean of a subclass . The discriminant vectors are then found using the generalized eigenvalues problem:

. (1)

Where is the covariance of the data, and is the diagonal matrix containing the eigenvalues of the solution.

Subclass divisions

The intuition for this problem is: we want to partition the classes such that the eigenvectors of the DA problem () will be similar to the vectors of PCA. We first show the relation between PCA and equation 1.

**Lemma 1:** If and have identical eigenvectors but then the eigenvectors of PCA are related to those of equation 1 : where . Where is the eigenvalue matrix of and is the eigenvalues matrix of .

**Proof:** assume that is the common eigenvector matrix of and , we get . Using equation 1 we get:

 (2)

**Lemma 2:** Let , and in the above equation, then:

1. If then
2. If then

**Proof:** Under the eigenvectors equivalency assumption we get:

Thus, if (i) holds then the order by which the eigenvalues are ordered in is increasing and . If (ii) holds then the order is decreasing, thus to get an increasing order in the eigenvalues of we need that turns the order backwards, thus:

.

Using the M matrices under each assumption yields for (i) and for (ii)

Summary of the lemmas: from lemma 1 we get that under the eigenvector equivalency assumption the eigenvectors of PCA, , are a base transformation of a matrix of the eigenvectors of the LDA solution. Then, we get that the order of the eigenvalues in diagonal matrices of and determines how M looks like. The lemmas lead to proposition 1: under our assumptions, the eigenvector of the generalized LDA problem (equation 1), , is identical to the eigenvector of PCA, , when is the value of . Thus, in our case the result of equation 1 is dependent on the ratio between the eigenvalues of and . Figure 2 shows the problem with what we have so far:



Explanation: In (a) the first eigenvector of the matrices is . We reduce our dimension to one, but we want to select a new axis that will have large variance according to but a small variance according to . According to we need to use , but has a larger variance in this direction. Here will be selected even though it is a bad option as a discriminant for classification (classes 1 and 3 are merged in ). In (b) again and are in conflict such that is selected instead of but here this yields a good classification discriminant.

As a result of Figure 1 we get **Theorem 1:** When the eigenvector of corresponds to one of the first eigenvectors of then the basis selected by equation 1 might not minimize the Bayes error (of the classification, on the training set). These cases are conflicts between PCA and LDA. When this number of conflicts is large, the probability that standard DA methods would fail increases. Explanation: clearly, this theorem shows the major problem with standard DA. In this paper the suggested approach tries to reduce the number of conflicts by dividing the classes to subclasses.

Let be the number of eigenvectors of that are along the same direction as eigenvectors of . A simple algorithm to calculate:

For i=1 to #(eigenvectors of )

 For j=1 to i

 If the

We can also compute. Where is the number of eigenvectors of . Note that here we consider only the top eigenvectors of . This is because in general, the first eigenvectors of are in accordance of the last eigenvectors of when the LDA criteria works well. It means that when the eigenvector of corresponds to small variation of we will not call this case a conflict.

We now move on to a general optimization criteria. We incorporate the eigenvalues and get:

.

Where is the number of eigenvectors after dividing the data to subclasses, and is the eigenvalue. Our goal is to maximize this function (?). Stopped reading here. See next paper (same authors).

**Subclass Discriminant Analysis (Zhu and Martinez, 2006 IEEE)**

Remark: this is a follow paper of the previous one. Hence, we shall use the same notations.

Extensions of LDA include:

1. NDA (ref 9): a nonparametric approach in which the normal distribution assumption is relaxed. However, we do assume that each class forms a homogeneous cluster.
2. aPAC (ref 16): approximate pairwise accuracy criterion.
3. PDA (ref 11, penalized DA): samples are weighted to reduce the role of unstable samples. Again we assume that each class forms a homogeneous cluster.
4. FDA (Flexible DA, ref 10): using nonlinear regression in the DA model.
5. GDA (Generalized DA, ref 2): kernel methods that try to map the data to a new kernel space and apply LDA in that space.

Here we try to model each class as a mixture of Gaussians. This model can approximate complex heterogeneous data (ref 21). Given the new subclass divisions we can now solve LDA. The difficulty here is to partition the classes such that the classification will be maximized. This paper presents two methods to address this problem. The first is based on LOOCV and the second is similar to the problem in the previous paper. The goal of this paper is mainly to improve classification and not to discover the underlying classes. As pointed out in the introduction, for the latter problem EM is used.

Further explanation of LDA extensions

The general Fisher-Rao criteria can be formulated as: were and define metrics (hence, they are positive definite and symmetric). Most DA methods reformulate or . For example, NDA estimates ( of the standard formulation) as:

Where is the mean of the nearest neighbors of the sample belonging to the class . Hence, this matrix corresponds to the between-class scatter. Also is a scale factor for dealing with outliers.

The aPAC method defines:

Where corresponds to the Mahalanobis distance between the classes, is a weight function that makes the contribution of each class pair equal to classification accuracy between the two classes.

The PDA method reformulate . Where penalizes eigenvectors that are noisy.

HLDA (heteroscedastic DA, ref 17) considers differences in class means and covariance. The authors use the Chernoff distance to estimate class similarity:

Where is the covariance of class , and is the average of and . For simplicity we assumed equal priors.

The SDA method suggested here accommodates for the situations that each of the above methods address. Another improvement is that SDA can produce more than features (as most DA methods produce). Given a partition of the classes to subclasses we define the between-class scatter as:

**Result 1:**  can be decomposed as with

Where is the prior of the (original) class , is the number of samples in subclass of the class . This result indicates that when we plug-in to LDA we intuitively look for separation according to the original and separation within each class (i.e., separation of the new subclasses). Practically, this approach may lead to too many subclasses (overfitting). An alternative approach is to define to favor separablity of subclasses of different classes:

Where is the prior of the subclass of class . Indirectly, this new metric also accounts for the subclass separation of the same original class. A virtue of this definition is that when we will try to use for optimization, we will favor the cases in which the subclass assignment improve classification. An example of using this new definition:



Here, the number of features is 3 (Fig a shows the raw data), and . The number of classes is and simple LDA solution as shown in (b). LDA will project the data along one axis, the X axis of (b). The horizontal axis is uniform distribution. SDA with is shown in (c). For each , the maximal rank of is 3 (as the original ).

Optimal subclass division

In the previous part we discussed how to score subclass division. In this section we will assume that we have a clustering algorithm that can dissect the classes and we discuss how to find the number of subclasses .

The first approach is based on LOOCV. The score we want to optimize is the classification accuracy: , where is an indicator function that receives 1 the sample is classified correctly. Given the new class partition we shall use LDA with the above definitions of to reduce dimensionality and to get the final model.

The second method is based on a stability score. From the previous paper we know that the standard Fisher criterion will not work when the angle between the eigenvector of and the first eigenvectors of is close to zero. This is formally computed as

Where . summarizes the disagreement between and . When is large we cannot maximize and minimize simultaneously. We want to divide the classes such that is minimized. We now have a problem. When increases the numbered of summed products increases also. It means that generally increases with . We therefore normalize:

The normalization constant stems from the fact that for a given eigenvector of we get .

Dividing classes to subclasses

We assume that each subclass is Gaussian. Of note, Kmeans and EM for mixture of Gaussians work better when the number of samples is large. The authors propose a new algorithm and compare it with Kmeans, EM and a non parametric clustering method (ref 15).

The proposed method is based on nearest neighbor. We start by sorting the sample vectors of class : . First, and are the two farthest vectors. Then is the closest vector to and is the closest to . The ordered set is then partitioned to clusters. In this process assume that each class is equally represented.

Each subclass is divided to the same number of classes. While this approach is oversimplifying the authors note that even if the data is from one Gaussian it can be represented as a mixture while the second direction is not true.

Experiments

The SDA method was compared with:

* LDA
* Direct LDA (DLDA)
* HLDA
* Kernel LDA with:
	+ Polynomial kernel, second degree (KP-LDA).
	+ Gaussian kernel with evaluated using CV (KG-LDA).
* NDA
* Complete Kernel Fisher DA (CKFD), again with polynomial or Gaussian kernel.

For NDA and HLDA, they tried different internal parameters and they report the mean and standard deviation. All these methods are used to first transform the data and then nearest-neighbor-method is used in the new space.

Datasets and results:

1. UCI datasets:
	1. WDBC – images of cells from benign and malignant cells from breast cancers. There are 569 samples. They use one repeat of 2-CV (even less than that, they use only one fold).
	2. Classify satellites using 36 features. Training set contains 4435 samples and the test set 2000 samples.
	3. MDD datasets: multi-features digital datasets. These datasets contain images of hand-written digits (1o classes). This dataset is represented in 5 different feature spaces.

Table 1 shows the classification accuracies. Table 2 shows the number of classes obtained in each dataset. Of note, SDA looks for the subclass division that shows the best classification and not the subclass division that best model the data.

1. Object recognition: they used the ETH-80 dataset. Images are of apples, pears, cars, cows, horses, dogs, tomatoes, and cups. There are 10 samples in each class. This dataset is used two times (LOOCV for each):
	1. Feature based approach: the features (300) are the distances from the centroids of the contour of the figure.
	2. Appearance based: use the pixels as features.

**Part-based statistical models for object classification and detection (Bernstein 2005, IEEE Computer Vision and Pattern Recognition).**

Two main problems in computer vision:

1. Classification of objects in pre-segmented images.
2. Detection of pre-specified objects in large images.

Important issues that such systems should address:

1. Scalability: the algorithms should be scalable when the number of classes explodes. Efficient multiclass classification algorithms should be sub-linear when classifying a new sample.
2. Modularity: we need a good way to combine different classifiers when we deal with many classes. This is because we have many sub-problems with possibly different classifiers. Probabilistic models are preferred to deal with this problem.
3. The classes are usually assigned arbitrarily and a vision system should be able to find new subclasses. For example, in Figure 3 we images of handwritten "7". Subclasses are American and European sevens. We will use mixture models to find such subclasses.

**Simultaneous Class Discovery and Classification of Microarray Data Using Spectral Analysis (QIU, JCB 2009)**

This paper proposes a new method called SPACC (Spectral Clustering for Class discovery and Classification) for simultaneous analysis of gene expression data. The method can help in classification analysis and in class discovery. The method is based on spectral clustering. The input is a weighted undirected graph in which nodes correspond to samples and edge weights correspond to similarity.

The graph is iteratively partitioned to two using spectral clustering. If all remaining samples are from the same class then we stop and define it as a leaf in the final hierarchy. The classification is based in nearest-neighbor approach. When classifying a new sample, its similarity with each leaf is calculated and the assignment is done to the most similar leaf.

Remark: this paper is very weak, given that it was published in 2009. There is no comparison to extant methods and the datasets used as case studies are old and known to be easy.

**Identification of pediatric septic shock subclasses based on genome-wide expression profiling (Wong et al. 2009 BMC Medicine)**

Introduction

Septic shock is an infection based disease. It is not a single homogeneous disease. Rather, it is heterogeneous and believed to be a syndrome comprised of several subclasses. Several attempts to find subclasses include physiological based methods and biomarker based methods (most from blood samples). Several studies (refs 5,6) tried to use serum IL6 levels to stratify patients. The goal was to find a population that could benefit from immune modulation therapy (refs 5,6). Other study identified subclass of children with increased survival rate due to admission of IL8 (ref 7). Refs 9-13 are studies of gene expression of septic shock patients.

In this data they had 98 septic shock patients and 32 controls. Initial differential genes detection: genes that have a 2-fold change in 25-50% of the patients as compared to the controls. This yielded 6099 genes. Then, they used hierarchical clustering and discovered three subclasses (Figure 1). Then they used this class partition as input to ANOVA and discovered 6934 probes (Bonferroni p < 0.05). Then they repeated the hierarchical clustering and rediscovered the same subclasses (Figure 2, PCA based on these genes in Figure 3).

 The subclasses where then analyzed based on the clinical phenotypes (Table 1). Subclass A had a significant higher illness severity, greater degree of organ failure, and higher mortality rate. Subclass A had higher Gram-Positive bacteria as compared to C. Subclass A patients are younger as compared to subclass B. Subclass B patients received higher proportion of hydrocortisone for cardiovascular shock as compared to C. Other features in Table 1 were not discriminative.

The 6934 genes were partitioned to 10 clusters using Kmeans. Ingenuity pathway-analysis is shown in Table 2. Then they took the genes of the discovered pathways (307 genes) and ran LOOCV on the subclasses using SVM and Fisher based feature selection. They report 91% accuracy (OVERFITTING?!). Of the top 100 selected genes, 44 corresponded to the top 5 pathways in Table 2. The expression matrix of these genes is shown in Figure 5.