Statistical Analysis and Factor Analysis of Gene Expression

David Carlson
Advisor: Professor Lawrence Carin

4/9/2010

Contents

1 Abstract

Recently, gene expression samples, where relative amounts of mRNA in a sample are measured, are becoming increasing popular to use due to decreasing costs and an increasing knowledge base. However, the form of the data still poses significant hurdles to data analysis due to the large number of factors compared to a low number of samples. In order to derive accurate and usable conclusions about the data, it is necessary to use complex statistical and signal processing methods. One way to analyze the data is to use machine learning methods that assume a sparse solution; this assumption makes the algorithms arrive at a solution that is based upon only a handful of genes, giving significantly more viable and useful results. This conclusion can be verified through testing the results on a reserved, unused part of the data.

Additionally, there are methods that can be used to reduce the dimensionality of the data so that more traditional statistical methods could be used to regress on the data. Methods such as factor analysis, including sparse PCA, use correlations between factors in the data to project a high-dimensionality system into a lower-dimension system. By using these methods, the goal is to reduce noise by following broad strokes of genes instead of fast-varying individual genes.

Using these methods, we attempt to look at gene expression data sets for multiple problems and determine indicators of outcome. One such data set we look at was on heart disease, where terminal heart patients at Duke Hospital provided blood for a microarray sample as well as clinical factors (simple blood tests) and other information. With this data, we attempt to predict whether they will be a case (MI or death) or a control (alive after 2 years). Additionally, we looked a data set of people that were given flu shots and attempted to predict the earliest we can tell whether they will become ill.

2 Introduction and Theory

There is a long process in order to get the data in the right form and there is a significant theory behind the methods and why they work. I expound on the methods and theory here.

2.1 Microarrays

The original data from microarrays are provided by a commercial company from blood samples. It is possible to use different body samples besides blood, but blood is among the most common. However, when the data returns it is not in an ideal format; instead, the data is absolute (logarithmic) measurements instead of relative measurements. Because of variations in sample size and random fluctuations in addition to difficulty with measurements, the total measurements are highly variable. In order to use the data accurately, it must be standardized to a specific curve (which is found through the examination of microarray samples over time). There are multiple methods to do the standardization, which all yield reasonable close solutions. In addition to commercially available standalone applications, it is quite common to use the Bioconductor package for the R programming environment (`http://www.bioconductor.org/`). I used this package as verification for standardization done by a standalone package, and the results matched up very well.

2.2 Separation of Testing and Training Sets

There are many different ways to test a solution to a dataset, and I need to explain the methods that we did and the assumptions behind them.

For the heart disease dataset, the setup is very straightforward. We used roughly half of the dataset for training and the rest for testing. In actually, the set for training is slightly more than half, because we need to balance the number of cases and controls in the training data to get the algorithm to work correctly. Additionally, in order to determine the validity of our results, we randomized the training and testing set for each iteration (with a seed, so that we can reproduce the results if necessary), so that we are sure we get randomized data sets and the results are not a fluke. Additionally, this allows us to get a histogram on our prediction accuracy over multiple iterations.

However, the setup for the flu time analysis regression is more controversial. With so few data points, separating into a training and large testing set is difficult to perform without shrinking our already tiny training set. Instead, we take out a couple data points at a time and determine how the solution works for those data points. There are two ways to do this: take out one person at a time and test whether you can determine whether that person will get sick or not or take out an entire time point and all samples in that time point. In my analysis, I took out an entire time point and using this regressed over the data and iterated through all the time points. Using these iterations you can plot at each point how accurately you are able to predict who will get sick or not.

However, there is a significant problem with this method: there is the possibility that the algorithms will learn which people are healthy or sick from their other data points. In theory, the genes not related to sickness will be highly variable and unable to regress upon, while the genes related to sickness will follow a expected time-evolving path that will be able to be picked up on.

Overall, this can be a problem if you allow the algorithm to use too much information but is an unlikely case for highly sparse solutions. Furthermore, after doing this method the results matched up with the results for the case where one person at a time was taken out.

2.3 The High p, Low n Problem

In traditional statistics, we often are capable of acquirable a great deal of samples (n) for a relatively low number of variables (p), as seen in [3], creating an overdetermined system, which leads to accurate results for least squares regression. However, in microarray analysis, the number of variables is in the thousands (for our data sets, roughly 10,000-50,000) while there are severe constraints on the number of samples due to issues of cost, willing and available patients, and time. Because of this, often sample points are very limited. For the flu regressions we basically had samples in the teens for all the time points, and for the heart data we had 447 samples. Both of these data sets are considered fairly large, but still even for the heart disease data set, 447 is significantly less that 50,000, creating a situation where the there are 2 more orders of magnitude of factors than samples.

If we attempted to use basic linear regression, we run into the problem that the problem is significantly underdetermined, and with the inverse matrix the factor weights become tragically variable with no reliable relation to applicable data. Indeed, when one attempts to do this you can get perfect regression in the training data, but the testing data is com-

pletely random due to the highly variable inverse solution to the highly underdetermined least squares problem, giving us absolutely no useful information about the system.

Instead, we need to use methods that can deal with the "high p, low n" situation. To do this, we can attempt to cluster genes or assume sparsity, which is possible through a variety of methods. The work in this area has been expanding in recent years, and since the 1990 an explosion of methods to do bioinformatics have been developed.

2.4 Bayesian Elastic Net and Relevance Vector Machine

Traditionally, in the medical field and many other applications, the Lasso methods has been used. As first described in by Tibshirani [6] in the 1990s, this methods places a limit on the L1 norm of the parameter weights in order to generate sparsity in the system. Specifically, there is a condition $\Sigma_{i=1}^{p}|w_i| < M$, where M is defined by the parameters that the system is regressed upon and w_i is the i^{th} parameter weight. This means that there is a limit on how many factors can be involved in the solution, because not the regression will pick only the statistically most important factors.

2.4.1 Bayesian Elastic Net

However, while the lasso gives sparse, viable solutions it tends to create situations where the solution is too sparse. Every biological system is complicated, so it doesn't tend to isolate to a single gene, but instead a broad combination of genes. However, sparsity cannot be abandoned with these methods if we are to get a realistic, viable solution. For that reason, the Elastic Net was introduced [9]. It combines the limit from the Lasso on the L1 norm and and a penalty from the L2 norm used in ridge regression. (The L2 norm is defined as $\Sigma_{i=1}^{p}w_i^2$). Using this, one can adjust parameters to "adjust the net" to choose the level of sparsity used in the model.

The biggest achievement of this model is that it is able to include multiple genes that vary together to improve consistency. For example, in the Lasso there is that problem that when two genes are highly correlated with the solution but the model is too sparse to use both of them. Choosing both would improve consistency, so the elastic net could be "expanded" to include such overlapping factors.

For this project, we used a Bayesian framework to implement the Elastic Net, so that the correct properties were obtained by setting priors and implementation was sped up via the use of variational bayesian techniques. The greater details can be seen in this paper by Minhua Chen [4]. Additionally, we adapted the method so that it can process multitask data, necessarily requiring in systems with multiple time points that solutions between time points are slowly varying to increase viability and consistency.

2.4.2 Relevance Vector Machine(RVM)

Additionally, while most models focus on linear regression, there are other ways of viewing the problem. One such method is the RVM created by Mike Tipping [7], which focuses on using kernels to relate similarity between unknown solutions and known solutions. That is, that instead of simple linear regression you instead fit a model $\Sigma_{i=1}^{p}w_iK(x, x_i)$. In this way, we measure similarity between points and enforcing sparsity can get a solution that is based on very few vectors. Additionally, while the kernel is typically defined as a

Gaussian kernel (using the pattern that $K(x, x_i) = exp(\frac{-(x-x_i)^2}{\sigma^2})$, it is possible to set up the system using the linear weights of the factors as the kernel. Then using this system we are able to approach linear regression with a new system that provides good results in reasonable amounts of time. Additionally, medically it is significantly easier to explain biologically that certain genes are going wrong instead of claiming that people have similar gene expression to healthy people, so using this kernel provides a more understandable solution.

Using this method can give better solutions than the Lasso, but it tends to suffer from the same problem of become too sparse in complex situations. However, it does provide a good alternative to try to get solutions to compare with the elastic net.

2.4.3 Necessary Simplifications

Because we unfortunately don't have unlimited resources in terms of computing power and time, it is impossible for us to regress usefully on the entire data set. In order to make the data processing actually tractable, we limit the number of genes we regress upon. There are many ways to do this, but the method we choose to use was to use a subset of genes that had the highest Fisher scores, which is defined as $|\frac{\mu_{case}-\mu_{control}}{\sigma_{case}+\sigma control}|$. From this it can be seen that genes that have the most individually separable will be included in the regression.

However, it needs to be stated that this information by itself is not good enough to solve regression. If one only used the top scoring genes, we wouldn't be able to do as good as using one of the previously described methods. In fact when when we implement the methods above, the strongest coefficients are more likely to have high scores, but the genes that are chosen are far spreading throughout the list of scores.

2.5 Factor Analysis

While enforcing sparsity is one method of dealing with these types of data sets, alternatively there are methods to attempt to reduce dimensionality without reducing information. At the simplest, one can simply take the eigenvectors of the system and use only the highest values; however, while this works it has the shortcoming of forcing orthogonality on the system. There are other ways of reducing dimensionality, however, and one such one method is factor analysis. The most basic expression for this breakdown is $x - \mu = LF + \epsilon$, where L is defined as the loading matrix and F is the factor matrix. μ is the averages for each original factor and ϵ is noise. At the conclusion of the factor analysis algorithm, you have a new data matrix F, which is of dimension $k \times n$, where n is the original number of samples and k is the new number of dimensions (factors), where $k << p$.

The actually implementation of factor analysis is performed by a variety of models and algorithms, which all fit the given model, but have significantly different ways of processing and occasionally results. One of the most common methods for performing factor analysis is the PCA (Principle Component Analysis), which the basics are detailed in a book by Jolliffe [2]. However, for our work we really want to remove information that appears to be superfluous and get broader patterns of gene motion. To do this, we used a method called Sparse Principle Component Analysis [8], which uses a limiting condition on the L1 norm to limit the factors from expounding patterns that are very weak. While this could remove information that is necessary for the prediction of heart disease, there is no real loss

because with such a weak pattern the projection onto the new factor would be highly noisy and randomized, giving little, if any, information, and would be lost in the data processing.

2.6 Biclustering

While factor analysis does a good job of reducing dimensionality, it unfortunately cannot account for similarity between samples. That is, a large number of samples we expect to be highly similar, and clustering these samples together may give us a better look at the groups of people and how groups as a hole are able to be predicted. This algorithm and method was described by Church and Cheng [1]. The basics of the algorithm are iterations occur trading off between cluster samples and factors to create a new data array where information is highly usable.

Despite the great results this algorithm has given in other cases, in our case, due to the highly similar profiles of all patients, we were unable to obtain useful results using this algorithm. Additionally, there are other methods that might be able to create useful types of clusters for people, such as neural net clustering [5] and other methods using manifolds.

3 Data Analysis

3.1 Heart Disease

For heart disease, we were provided a dataset from Duke University Hospital that was comprised of 447 samples of patients with critical forms of heart disease, and the data consisted of basic medical data (bmi, weight, heart rate, RDW, etc.) as well as full mRNA expression values. For this data, we were attempting to determine whether a patient was a case (MI or death) or a control (still living after 2 years). Traditionally, in this type of patient doctors are unable to predict the outcome accurately, having a roughly 50-50 chance of getting it right. Therefore, by providing us the data, the doctors hoped that we were able to determine with reasonable accuracy the future condition of the patient, but their hopes were not high. Previous attempts had not been very successful, but this attempt was significantly more data available.

To analyze the heart disease dataset, we went through a variety of algorithms looking at the expression data as well as the clinical factors. For the clinical factors, we found that the RDW (Red Blood Cell Distribution Width) was the most important factor to determining the outcome. Expanding on this, we found the correlating genes to the RDW factor in the expression data; then, using this data we regressed only on this correlated genes using the elastic net. Surprisingly, we were able to achieve accuracy basically equivalent to the rate when we regressed on a much larger subset of genes (not exclusive to RDW expressions). The error rates (how often we were wrong in our guess), is shown in Figure 1. The data is shown in a histogram, where many iterations were run with randomized training and testing subsets of the data, and each results have a different error rate. They are grouped in the figure.

The histogram shows that the error rate in our multiple runs on the data were almost approximately the same as those using the top fisher scores. The implications of this are that the RDW genes themselves give can explain for a large part the heart disease outcome.

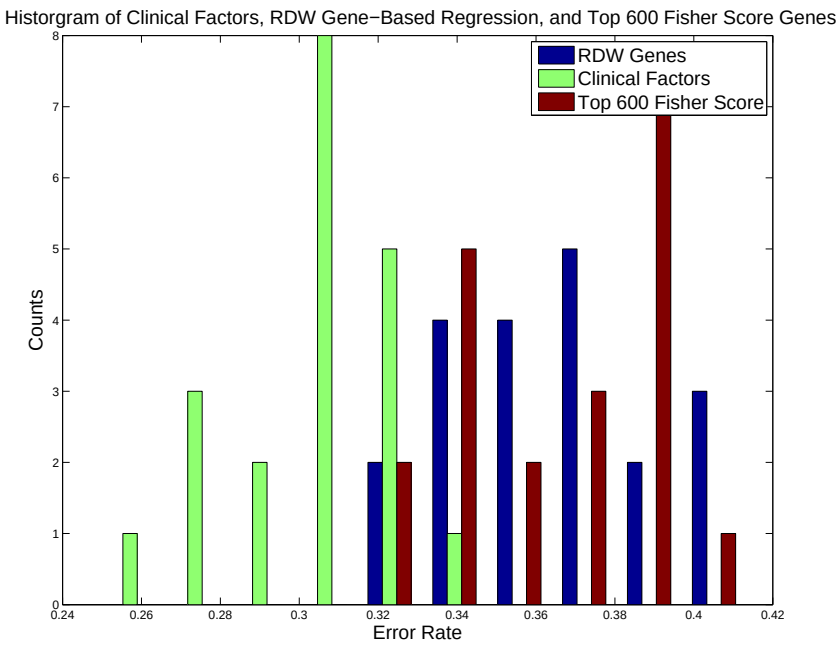

Figure 1: Histogram of the Error Rates for an Elastic Net Regression on only the Clinical Factors, the Genes with the Top 600 Fisher Scores, and the Genes Associated with the RDW factor

At the same time though, it should be noted that the analysis from the gene expression couldn't equal our performance looking only at the clinical factors (simple measurements and blood tests done routinely at a doctor's office). But the fact that the gene analysis can support the conclusion from the clinical factors is highly significant.

Additionally, the pf-pd curve paints a larger image of the regressions. As a note, the pf-pd curve was averaged over multiple random iterations to give a smoother, more representative curve. The curve for only the clinical factors can be seen in Figure 2. The curve for the genes associated with the RDW clinical factor is in Figure 3. As can been seen in the figures, the curves show that information is gained from this analysis, but that the accuracy is better in that based on the clinical factors, but that both give significant advantages over simple guessing.

3.2 Flu Vaccinations

As part of a DARPA project, Duke has been given funding to analyze data based on flu inoculations. As part of this research, blood was taking and expression data was acquired for multiple time points after people were given a variety of inoculations, including a typical flu virus and the rhinovirus. The people were tracked by doctors who determined whether the patient became ill and at what point they were able to measure virus forms in the nasal drip. The goal is to be able to predict before a patient becomes contagious so that the effect of the virus can be reduced.

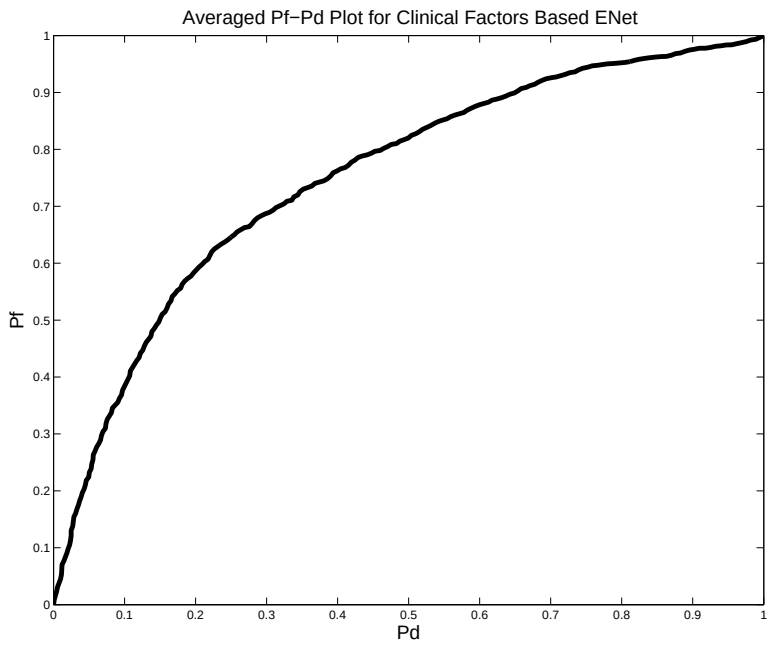

Figure 2: The pf-pd Curve based on the Elastic Net Regression on only the Clinical Factors

3.2.1 Flu

The generic flu dataset was by far our most complete data, having about 14 people at each data point. Additionally, there are several genes that showed definite progressions over time. For example, Figure 4 shows the genes of ENOSF1 and LILRB2 which evolve and separate over time. This genes are very good at showing the reaction in the body and immune system as the patient get set, but we would like to be able to show at the earliest point possible. There are genes that show separation through all time points, such as HLA-DOB shown in Figure 5. This gene does not show any time evolution and is therefore a good predictor of early time. However, since the patients do not change from time point to time point, we expect that there is some correlation between samples from the same people at different time points. Because of our limited sampling basis, it is difficult to validate this gene because it is possible that the separation could just be random, much as all genes that are separable at all time. Because of this, these types of genes are only used sparingly in the model and the predictor. Using a multi-task RVM method, we were able to obtain results where we could reasonably predict solutions. We fitted a curve of the form $\alpha exp(-\beta t)$ to the regression, and the results are shown in Figure 6. Additionally, the pf-pd curves give additional information about the individual time points. This data is shown in Figure 7.

3.2.2 Rhinovirus and RSV

In addition to the results for the flu virus, we also examined patterns after inoculation of the rhinovirus and RSV. At the point when we were performing the regression, we were

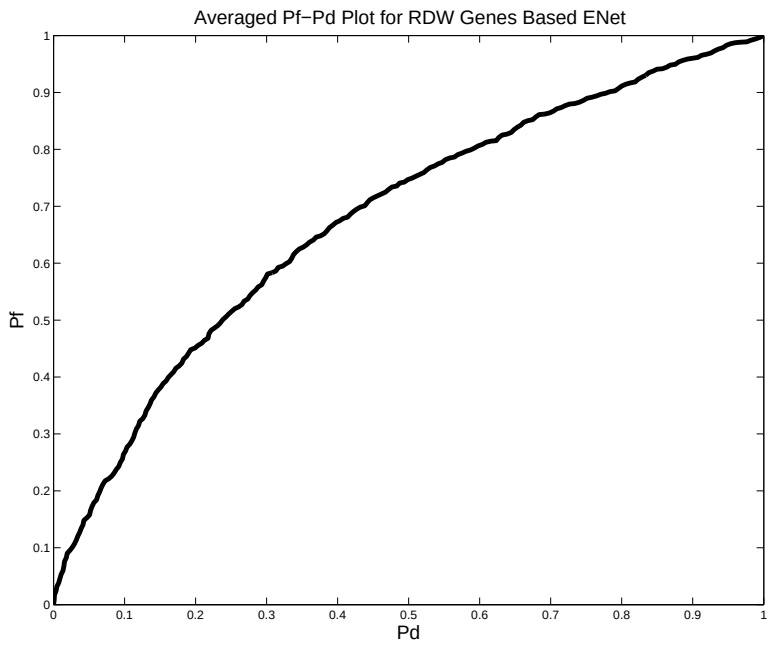

Figure 3: The pf-pd Curve based on the Elastic Net Regression on the Genes Associated with RDW

still waiting on parts of data at many time points, so the results are very preliminary. Nevertheless, we were encouraged by the preliminary results, where using a multi-task RVM method we obtained reasonably good results. The error rate for the rhinovirus data is shown in Figure 8 and the error rate for the RSV data is shown in Figure 9.

4 Conclusions

4.1 Heart Disease

The goal of this project was to develop a test that doctors could perform to identify the best course of action for each patient as determined by expression information. Overall, our methods reveal a prediction that yielded much greater accuracy in outcome that simple guessing, and the ability to have a more accurate glance into the future will help doctors to determine the best course of action for patients. However, what may be more important in the end is the link between RDW and heart disease outcome, as well as the genes related to RDW. At the very least, this investigation revealed that there is significantly more information to be revealed about RDW, but also the fact that there are genes associated with RDW and confirmed by the expression analysis could lead to treatments to help alleviate conditions of heart disease.

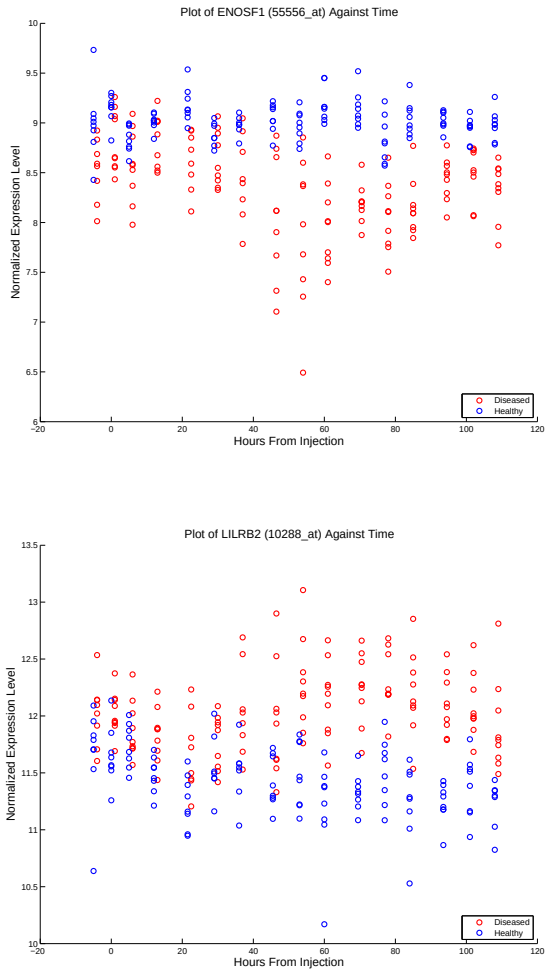

Figure 4: Example of Genes that Show Time-Evolution

4.2 Flu Inoculations

The initial goal of this project was to explore the possibility of determining how early it is possible to predict who will get sick with diseases, with the immediate application to situations such as submarines, where one sick person can infect many others. Because of the possible rehabilitating effect of viruses in such close quarters, the DOD funded a DARPA project, part of which was interested in these flu inoculations.

As a result of our investigation, we found that we are able to predict who would develop the illness roughly 24 hours after the shot, significantly before a patient would become contagious and show visible signs.

Additionally, it seems highly plausible that we would be able to predict before vaccinations were administered who would develop the disease based on their current health. In our small data set, it seemed highly likely that this approach could give very good results, but because our sample size is unfortunately so small it is difficult to determine whether we are simply "learning" which samples are which or whether we are hitting actual health indicators.

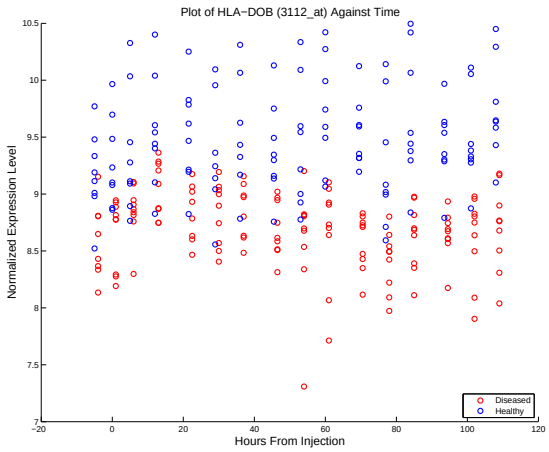

Figure 5: Example of Genes that Show Constant Separation

Overall, this data gives promising results, but to really certify our conclusions it would be desirable to use a significantly larger data set that would allow for proper testing techniques.

5 Thanks

I would like to thank Professor Lawrence Carin as well as grad student Minhua Chen for mentoring me through the summer and fall.

References

[1] Yizong Cheng and George M. Church. Biclustering of expression data. In *Poceedings of the 8th International Conference on Intelligent Systems for Molecular Biology*, pages 93–103, 2000.

[2] I. T. Jolliffe. *Principle Component Analysis*. Springer, 2nd edition, 2002.

[3] Lawrence K Mcknight, Adam Wilcox, and George Hripcsak. The effect of sample size and outcome prevalence on supervised machine learning of narrative data. *Proc AMIA Symp.*, pages 519–522, 2002.

[4] et. all Minhua Chen, David Carlson. The bayesian elastic net: Classifying multi-task gene-expression data. *Submitted to Trans. on Signal Processing*, 2009.

[5] Kadim Tasdemir and Erzsébet Merényi. Exploiting data topology in visualization and clustering of self-organizing maps. *IEEE Trans. on Neural Networks*, 20(4):549–562, April 2009.

[6] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288, 1996.

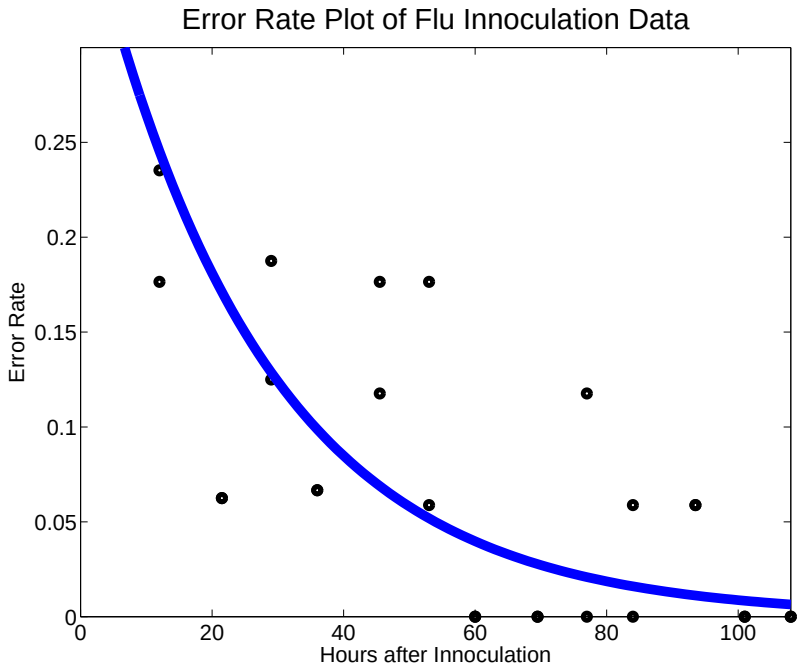

Figure 6: Error Rate of Prediction of Flu Data as a Function of Time

[7] Michael E Tipping. The relevance vector machine. *Advances in Neural Information Processing Systems*, 12(652-658), 2000.

[8] Hui Zou, Trever Hastie, and Robert Tibshirani. Sparce principle component analysis. *JCGS*, 15(2):262–286, 2006.

[9] Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *J. R. Statist. Soc. B*, 67(2):301–320, 2005.

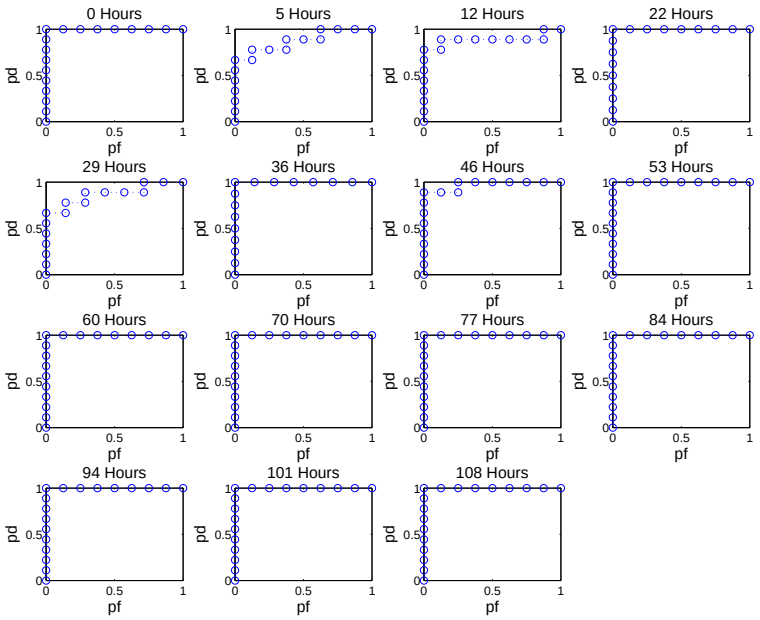

Figure 7: Pf-Pd Curves at individual Time Points

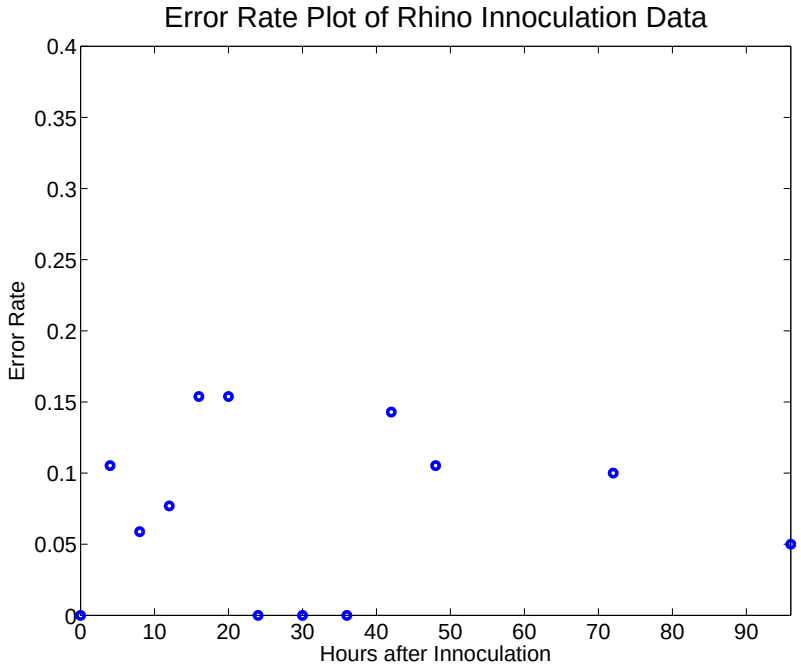

Figure 8: Error Rate of Prediction of Rhinovirus Data as a Function of Time

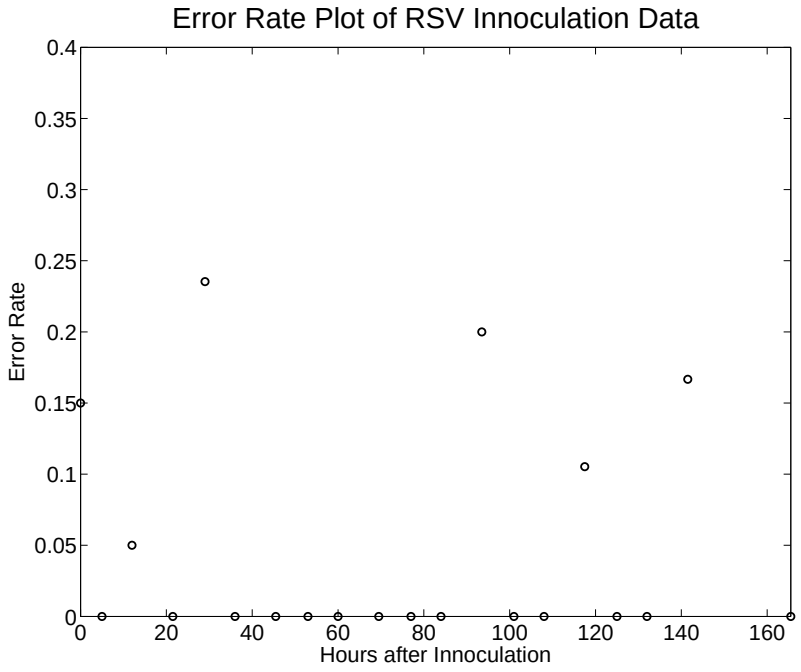

Figure 9: Error Rate of Prediction of RSV Data as a Function of Time