Data Analysis and Mining in the Life Sciences

Nam Huyn
SurroMed, Inc.
2375 Garcia Ave, Mountain View, CA 94043, USA
phuyn@surromed.com

Abstract

Biotech companies routinely generate vast amounts of biological measurement data that must be analyzed rapidly and mined for diagnostic, prognostic, or drug evaluation purposes. While these data analysis tasks are critical to their success, they have not benefited from recent advances that emerged from database and KDD research. In this paper, we focus on two such tasks: on-line analysis of clinical study data, and mining broad datasets for biomarkers. We examine the new requirements that are not met by current data analysis technologies and we identify new database and KDD research to address these needs. We describe our experience implementing a *Scientific OLAP* system and a data mining platform for the support of biomarker discovery at SurroMed, and we outline some key technical challenges that must be overcome before data analysis and data mining technologies can be widely adopted in the biotech industry.

1 Introduction

A central mission among a growing number of biotech companies is to discover biological markers. A *biological marker,* or biomarker, is a "characteristic that is measured and evaluated as an indication of normal biological processes, pathogenic processes or pharmacologic responses to therapeutic intervention" [10]. For example, high levels of cholesterol in human blood have commonly been used as a biomarker for heart diseases. New biomarkers are being sought that enable diseases to be diagnosed more accurately or earlier than is currently possible. Thanks to breakthroughs in *high-throughput measurement* technologies in the last five years [14, 13], tools such as gene chips, protein chips, and mass spectrometry are now widely available that are capable of detecting hundreds of thousands of gene products, proteins, and small organic molecules. These tools enable biotech companies to routinely generate, from tiny volumes of biological materials, very high volumes of measurement data that must be summarized, compared, and viewed efficiently. This approach to biomarker discovery is illustrated in Figure 1.

These data analysis tasks are critical to the success of biotech companies in biomarker discovery, yet support from technologies such as OLAP (see [3, 16] for recent surveys of On-Line Analytical Processing) and data mining has been inadequate. While these technologies have been widely adopted in financial and e-commerce arenas, such is not the case in the biotech industry. To understand why, let us take a closer look at the nature of data generated in *clinical studies,* i.e., controlled scientific experiments designed to answer specific clinical research or engineering questions such as drug efficacy, biomarker identification, and measurement method validation. Typically, the protocol for a clinical study specifies the following "ingredients": subject population, i.e., a well-characterized collection of subjects to be included in the study; biological samples, i.e., what kinds of samples (e.g., tissues, body fluids), how many and when they are drawn from the subjects; measurement methods, i.e., biological/chemical assays and instruments used to analyze the samples. Figure 2 shows a view of what the data schema might look like in a clinical study aimed at evaluating drug efficacy.

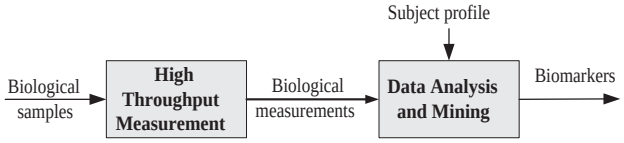


Figure 1. "Shotgun" approach to biomarker discovery.

subject	*draw*	*clinicalCls*	*drugCls*	m_1	m_2	...
John	1	Asthma	A	3.1	5.4	...
John	2	Asthma	A	4.6	5.3	...
Jane	1	Healthy	B	1.2	5.5	...
Jane	2	Healthy	B	1.7	5.6	...

Figure 2. Multidimensional view of clinical study data.

In this view, each row corresponds to an observation, i.e., a biological sample with all its characteristics and measurements performed. The *draw* column represents the time point when the sample is taken, the *clinicalCls* (resp. *drugCls*) column represents the disease (resp. drug) group which the subject belongs, and the m_i's represent biological measurements. This example illustrates the fact that clinical study data have a natural

multidimensional view, where observations are the facts of interest, *draw*, *clinicalCls* and *drugCls* are the dimensions, and the biological measurements are the target measures. While this view of clinical study data suggests that OLAP and data mining tools may be used for their analysis, a closer look reveals some fundamental differences between clinical studies and traditional applications:

- The subject population is carefully selected to minimize sampling biases, especially when the number of these participants is limited (typically in the 100's). Also, biological samples are drawn at carefully planned time points.

- Observations are linked to subjects, while in traditional data analysis applications, subjects are usually not tracked across transactions.

- The number of measurements made on each biological sample is several orders of magnitude larger than the number of samples, while in traditional applications, the number of facts usually far exceeds the number of target measures.

- An important goal of data analysis in clinical studies is to generate and validate hypotheses, following established scientific methods. For instance, the purpose may be to validate drug efficacy in clinical trials, validate bioanalytical methods in assay development, evaluate therapeutic effects in drug discovery, identify disease biomarkers for diagnosis or prognostics purposes, study protein interactions, or calibrate and optimize instruments.

- Traditionally, measures that are the target of analysis are chosen carefully and often have clear meaning. In clinical studies by contrast, we are typically less selective about them, and domain knowledge about the measurements made is often limited. In fact, many clinical studies are designed precisely to help discover this knowledge.

These differences translate into requirements that have not been met by mainstream data analysis technologies. In the next section, we propose the concept of Scientific OLAP to accommodate the requirements unique to online analysis of clinical study data and we describe our experience implementing such a system at SurroMed. In Section 3, we explain the challenges mining broad datasets for biomarkers. We review some of the relevant data mining approaches from the literature and explain why they are not adequate. We then describe our experience implementing a data mining platform that supports biomarker discovery at SurroMed. Section 4 summarizes key future challenges and the paper concludes in Section 5.

2 Scientific OLAP for Clinical Studies

In this section, we show new on-line data analysis requirements that are not found in traditional OLAP but that turn out to be very important for the domain of clinical studies. Generally, these requirements include more rigorous and richer types of data analysis using established statistical methods, more stringent notions of comparisons, the need to qualify results to minimize chances of making the wrong inference based on a limited number of observations, and the ability to handle large numbers of target measures. We propose the concept of *Scientific OLAP* as an extension of traditional OLAP that accommodates these unique requirements, which we describe below.

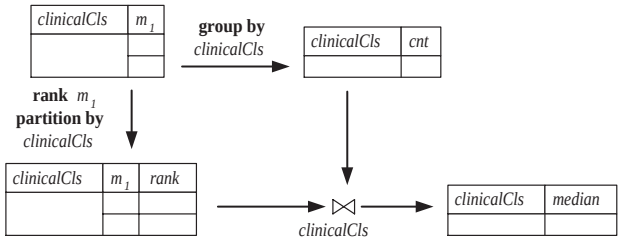
2.1 Rank-based aggregation

In traditional OLAP and SQL systems, standard aggregate operators are typically limited to `COUNT`, `SUM`, `AVG`, `STDDEV`, `MIN`, and `MAX`. Notably missing from these systems are the `MEDIAN` operator and the more general `PERCENTILE` operator. However, in many experimental sciences and in biology in particular, summarizing data using medians and percentiles is the **norm**, for good reasons. First, measurable biological entities, such as the concentration of many proteins expressed in human serum, often are not normally distributed. For these biological entities, `MEDIAN` gives a more accurate summary than `AVG`. Furthermore, measurements often are noisy and error-prone, which make `MEDIAN` a more robust operator against outliers. Also, `PERCENTILE` gives a more detailed summary of the data distribution and is commonly used to define and identify outliers.

While rank-based aggregate operators such as `MEDIAN` and `PERCENTILE` are absent from traditional SQL and OLAP systems, a partial solution has recently appeared in some commercial systems where SQL is extended with a family of functions, called *analytic* functions, that provides better support for analytical processing. An example is the `RANK` analytic function which computes the ranking for each row in a rowset, relative to a row-dependent group of rows. To illustrate how this function works, consider the view from Figure 2 and let us call this view *observations*. The following query:

```
SELECT subject, clinicalCls, m1,
  RANK() OVER
  (PARTITION BY clinicalCls ORDER BY m1)
FROM observations WHERE draw =1
```

computes the ranking in m_1 of all observations at time 1 within each clinical group. This ranking will be useful for computing the median in m_1 for each clinical group, as sketched in the following figure:

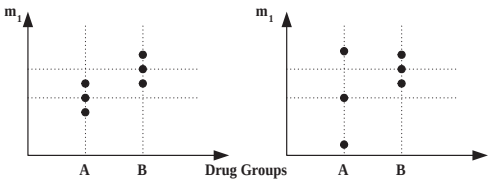


The `RANK` analytic function may be used to implement medians and percentiles, but the lack of true rank-based aggregations makes the implementation of many statistics commonly used in clinical studies both cumbersome and inefficient.

2.2 Multiple-group comparison

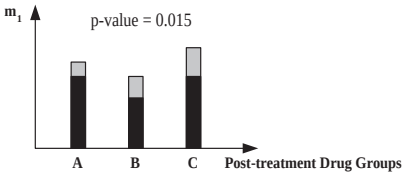
A common question in clinical studies is whether or not several groups of observations differ with respect to some measures in any *significant* way and not by chance. For instance, to study the effect of several drugs on human subjects, a separate group of subjects is often recruited for each drug, and in order to ensure that no bias has been introduced in the drug group assignment, it is important to verify that the drug groups exhibit no significant differences before any drug is administered. Another common example of group comparison arises in studies for diagnostic markers where a battery of measurements is performed on subjects that belong to different disease groups and where measurements that show significant differences between the groups are to be identified.

Support for multiple-group comparisons in traditional OLAP systems is typically limited to using first-order statistics such as the mean. However, as the following figure illustrates, these statistics are no longer sufficient to detect subtle but important differences.



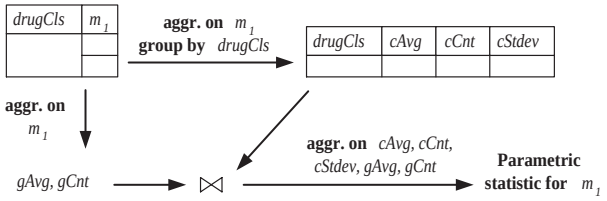
In this figure, the differences in means of measurement m_1 between drug groups A and B are identical in both graphs. However, since the values are more scattered in the right graph than in the left graph, intuitively the difference on the right should be less significant than the difference on the left.

Thus, in order to support group comparisons which are clearly more stringent in clinical studies, summaries in OLAP must include not only the group averages but also some second-order statistics such as the variance within each group and some measure of how significant the differences are. OLAP front-end tools that support richer visualization are also needed. For instance, the effects of drugs A, B, and C on measurement m_1 might be summarized in the following plot:

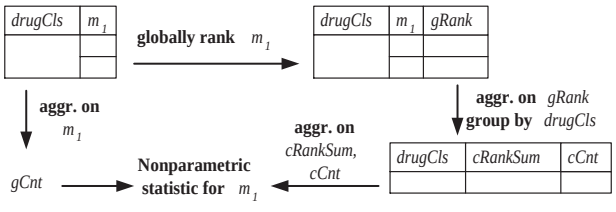


using something called the *p-value* to measure the probability that the drug effects are identical by chance (i.e., the smaller the p-value is, the more significant the difference becomes).

A statistical test commonly used to measure significance is the standard *ANOVA F-statistic* (see Analysis of Variance in [12]), which can be implemented easily using an aggregate query nested within another: the inner query summarizes the statistics within each group, the outer query combines these statistics across all groups, and both queries use only standard SQL aggregate operators, as sketched in the following figure:



This statistic assumes that the data is normally distributed (in statistics, tests that use the normality assumption are called *parametric*). Alternatively, we can use non-parametric statistics which are more robust against data distribution variability, such as the Kruskal-Wallis statistic [12]. To test the difference between groups of values, the global ranking of all values is used: the groups are dissimilar if the sum of the ranks within a group is disproportionate to its size. Using the `RANK` analytic function, the Kruskal-Wallis statistic can be implemented as sketched in the following figure:

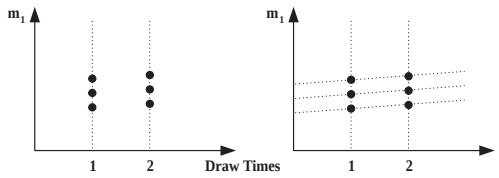


2.3 Repeated observations

So far, in comparing groups of observations, we have ignored that observations from different groups may be related to each other. For example, some measurements are made on the *same* subject albeit at *different* time points, e.g., before and after treatment. These related

observations, or *repeated observations*, exemplify what is known in classical statistical testing as *repeated measures*. Note that in repeated observations, the "common" parameter is not restricted to a subject and the "varying" parameter can be any experimental condition. For instance, in an experiment designed to evaluate the effects of varying an instrument's settings on the measurements, the common parameter could be a calibration sample and the varying parameter could be the speed at which the instrument is run.

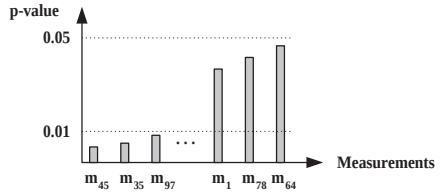
If we ignore these relationships between observations, we may fail to detect small but significant group differences, as the following figure illustrates:



On the left, the difference between groups is not statistically significant, since the difference in mean is small compared with the variance within each group. But on the right, with the additional knowledge that the observations are paired, intuition tells us that the difference should be significant, since the observed values consistently increase as we move from one group to the other, albeit in very small amounts. Note that repeated observations are distinct from time series for which trend analysis is supported in many OLAP systems, since the varying parameter does not have to have a natural progression. Also, traditional multidimensional models have no provisions for capturing the concept of repeated observations. To support their analysis, these models must be extended with annotations that can be used to define which dimensions, if any, play the role of common parameter. Once these relationships are captured, significance testing is not difficult: statistics commonly used for comparing groups of repeated observations include the *Paired T-Test* and the *Wilcoxon signed-rank statistic* [12], which can be implemented using standard SQL aggregations and the `RANK` analytic function.

2.4 Scaling with the number of measures

In Section 1, we used a multidimensional view of the clinical study data where each measurement is treated as a separate target measure. Since the number of measurements in typical clinical studies is extremely large (say in the 10,000's), this view is not practical: in order to visualize the summary statistics for all the measures using traditional OLAP front-end tools, one would have to sequence through a large number of screens! A better alternative is to represent the measurement type as a dimension. Thus, *slicing* on a particular measurement would show a summary of the comparative statistics for that measurement. This representation also allows us to compare all the measurements side by side in the same plot, to use common OLAP operations such as *dicing* to view only those measurements whose difference satisfies a user-specified significance threshold, and to rank the measurements according to their level of significance. An extended OLAP front-end tool might visualize the significance of the measurements in one chart as shown in the following plot:



which would help quickly reveal the important measurements.

2.5 Implementation of a Scientific OLAP system

Figure 3 depicts an on-line data analysis system we implemented directly on top of a relational database, which we use routinely to analyze clinical study data:

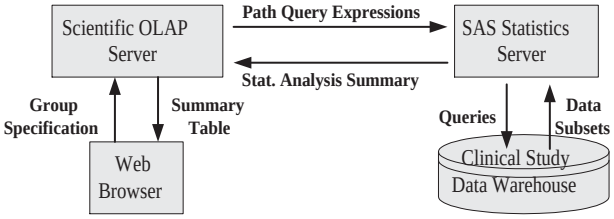


Figure 3. A scientific OLAP system for clinical studies.

In this implementation, comparative statistics, included in most analysis result summaries, are evaluated in a statistics server, separate from the database server. While some statistics could have been implemented using straight SQL, the use of an established statistical computation engine to compute them is purely for acceptance reasons, that is, at least until a database extension certified for statistical analysis is available. This decoupling results in processing inefficiencies mainly due to high volumes of network traffic and to the inability to take advantage of query optimization: for instance, instead of relying on the relational engine to optimize the execution of an aggregate query, the data is aggregated on a group-at-a-time basis. Also, because aggregate view materialization is not used, every new view request is evaluated against the base data, which results in further delay in processing the request.

To specify how data is to be aggregated and compared, we do not use the cube manipulation metaphor embodied in traditional OLAP front-end tools. Instead, the interface allows the user to recursively partition a

given group of observations along any dimensions into subgroups, and to select arbitrary subgroups to analyze or compare. To illustrate this approach we call *dynamic group specification*, Figure 4 shows a hierarchy of groups of observations that the user obtained by first expanding the top node (representing the initial set of observations) along the *clinicalCls* dimension, and then expanding the remaining nodes along the *drugCls* and *draw* dimensions:

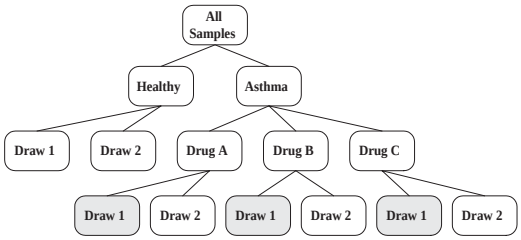


Figure 4. Dynamic group specification.

From this hierarchy, if the user wants to compare the different drug groups of asthma patients, he would select the nodes as highlighted in the figure. The system then maps this group specification to SQL path query expressions which the statistics server submits to the database server for execution. The advantage of our approach to group specification is two-fold: first, in order to view a particular aggregation summary, the user is not required to "navigate" through summaries for the intermediate aggregations, which may involve unnecessary computations; furthermore, since our group hierarchy does not require two nodes from the same level to be expanded along the same dimension, our method of group specification provides more flexibility than traditional OLAP systems. However, the lack of navigational capability is also a disadvantage, because it does not allow the user to follow a train of thought. Also, our approach does not scale well with dimensions that have a large number of distinct values. Finally, because the number of measures can be large, the summary of (comparative) analysis is shown in a table format, one row per measure, instead of a bar chart format used in a typical OLAP system.

3 Mining Broad Datasets for Biomarkers

In this section, we assume that the measurements collected on biological samples are used as the measurable characteristics for biomarkers. Thus, the biomarkers we are looking for are combinations of measurements (or simply measures) that can be used to predict a clinical endpoint, say a given disease. A *broad dataset* is simply a collection of observations where the number of measurements is much larger (typically several orders of magnitude larger) than the number of observations [15]. Figure 5 illustrates a broad dataset with only 200 observations but 100 times as many measurements.

observation	*clinicalCls*	m_1	m_2	...	m_{20000}
1	Asthma			...	
⋮	⋮	⋮	⋮	...	⋮
200	Healthy			...	

Figure 5. A broad dataset for biomarker discovery.

To find a biomarker, we would like to use these observations as a training set for building a classifier that can accurately predict the clinical class from all or a subset of the measurements. This description almost fits the classical definition of supervised learning [17], except that the input to the problem is a broad dataset. In traditional supervised learning applications, a large number of observations are typically available for training, and the data dimensionality is usually much smaller than the number of observations. Moreover, domain knowledge is often available to help pre-select dimensions that are relevant to the application. In our application, none of these assumptions hold for the following reasons:

- Currently, the biological processes that underlie many diseases are still poorly understood. To study these diseases, since we have little *a priori* knowledge of what measurements are important, we measure as many biological entities as possible, many of which we know very little about. For example, the number of different proteins that can be measured in human blood is estimated to be in the 100,000's.

- While traditional biomarkers use single biological entity measurements (e.g. $CD4^+$ T-cell concentration), modern bioanalytical instruments can perform a variety of measurements at a much lower granularity (e.g., subspecies of $CD4^+$ T-cells), many of which do not directly correspond to known biological entities (e.g., mass spectrometry data [11]). Our main premise is that a combination of several of these lower granularity measurements may be a much better disease indicator than many of the biomarkers currently in use.

- Because of the potential interactions between biological entities, many of which are currently unknown, "derived" measures are commonly considered besides the "base" measures. For example, the ratio between T-cell and total white blood cell counts is known to be a better indicator for asthma than both counts used separately. Thus, if we systematically combine the base measures to derive new measures using products and ratios for example, the number of final measures to be considered can be astronomical.

Most traditional classification techniques require that the number of dimensions be small compared with the number of training samples and thus cannot be applied to analyze our broad datasets directly. For those techniques that do not impose such a requirement, finding a model with good prediction accuracy is highly unlikely because of the large number of candidate models that can fit the training set perfectly.

Thus, the high dimensionality of our broad datasets must be reduced drastically before accurate classifiers can be built. Figure 6 illustrates the relationship between *dimensionality reduction* and predictive modeling in biomarker identification.

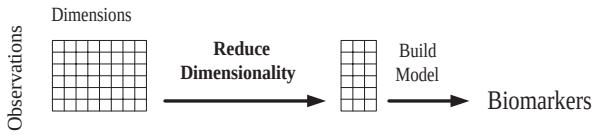


Figure 6. Critical role of dimensionality reduction in biomarker discovery.

It is important to distinguish between dimensionality reduction and *data reduction*, a term often mentioned in the KDD literature. In many data mining problems that involves analyzing a large number of observations, building a model can be time consuming. The challenge there is how to reduce these observations to a much smaller subset so that a model can be built more efficiently, without degrading the quality of the model too much. For example, various statistical sampling techniques have been devised to solve this data reduction problem, which clearly does not address our problem. The distinction between dimensionality reduction and data reduction is illustrated in Figure 7.

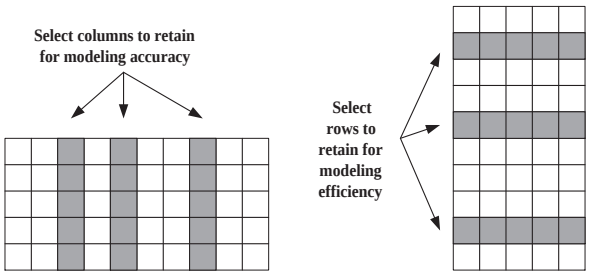


Figure 7. Horizontal vs. vertical reduction.

Dimensionality reduction is not only critical for biomarker identification but also important in its own right because it may provide valuable insights into the precise role various biological entities play in many disease processes. Surprisingly, there has been relatively little KDD research in this area, as most research has focused on scaling up with the number of observations rather than the dimensionality. In the following, we discuss three main approaches to dimensionality reduction: feature elimination, feature synthesis, and feature subset selection. Generally, these approaches can be used independently or in combination.

3.1 Feature Elimination

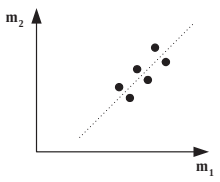
This approach assumes that among the initially large pool of dimensions, many will not be useful in discriminating between different clinical classes and thus can be eliminated from consideration. It is important to eliminate as many dimensions as possible early on because of the sheer number of dimension combinations we must consider eventually. There are mainly two ways a dimension can be eliminated:

- It can be irrelevant, i.e., by itself, it cannot discriminate between the classes.
- It can be redundant, i.e., it has strong similarity with another dimension.

In the first case, each dimension is evaluated separately, using any technique that scores how well it can discriminate between classes, such as the measure of statistical significant difference described in Section 2. A dimension is eliminated if the score is lower than a user-specified threshold. In the second case, each pair of dimensions is evaluated for similarity using, for instance, some measure of correlation. If the similarity score is higher than a user-specified threshold, one dimension in the pair can be eliminated.

As a practical way to implement this approach, we first eliminate all irrelevant dimensions, and among the remaining dimensions, eliminate the ones that are redundant. In a pair of highly similar dimensions, we choose to eliminate the most irrelevant one. This approach can be efficiently implemented and scales with the square of dimensionality.

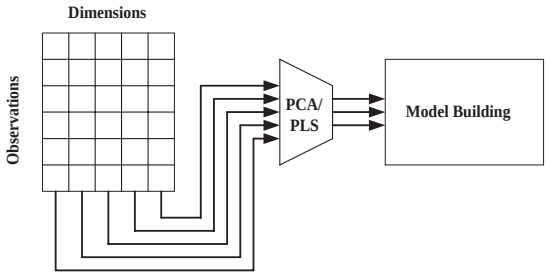
The main challenge using this approach is how to choose the thresholds appropriately. There may be dimensions that are bad discriminators individually but excellent discriminators when used in combination, as the following figure illustrates.



Thus, setting a threshold too aggressively may result in discarding the wrong dimensions, but setting it too conservatively may not reduce the combinatorics sufficiently.

3.2 Feature Synthesis

The main idea of this approach to data reduction is to combine the original dimensions (assumed to be numeric) into new and fewer dimensions that retain much of the information encoded in the original dimensions. *Principal Component Analysis* (PCA) and *Partial Least Square Regression* (PLS) are commonly used techniques for computing new dimensions that are linear combinations of the original ones and are statistically uncorrelated with each other. The reader is referred to [5, 18] for a more detailed description of these techniques. Essentially, they are numerical techniques that find orthogonal directions in the original multidimensional space that maximize the variance in the observations. For the purpose of supervised classification, PLS performs better than PCA since it also maximizes the correlation of the observations with the class label. To mine a broad data set, the observations are first projected onto the new dimensions (computed by PCA/PLS). These projections can then be used as the training data for a model builder, as illustrated in the following figure:



PCA and PLS are usually used to reduce the dimensionality of a dataset when the dimensionality is less than the number of observations. When the dimensionality is larger, these techniques must be extended, but the number of new dimensions can never exceed the number of observations. As a consequence, the reduction ratio needed for our broad datasets would be very significant: it would be very unlikely that the new dimensions are better than the original dimensions at discriminating between classes. Furthermore, if the new dimensions produced by PCA/PLS are used to build classifiers, classification of new data still uses all the original dimensions, since each new dimension is a function of all the original dimensions. From a practical standpoint, biomarkers that require measuring a large number of biological entities are not desirable. Finally, the numeric coefficients in the combinations do not tell us a lot about the original dimensions (e.g., the fact that two dimensions are highly correlated) and may not help us focus our attention on a few promising dimensions on which to do further analyses.

3.3 Feature Subset Selection

The problem of identifying dimension subsets that can be used to build accurate classifiers is not new and is known in the KDD community as *feature subset selection* (see [9, 2] for recent surveys). What makes this problem interesting is not only the high combinatorics involved but also the absence of obvious pruning heuristics: for example, prediction accuracy is not a monotonic function of the dimension sets with respect to set inclusion.

Unfortunately, most work in this area has been motivated differently than ours. Most of this work implicitly assumes a good predictive model that uses the full set of dimensions can be built. Since computational complexity of model building increases rapidly with dimensionality, their main goal is to reduce the number of dimensions in order to improve model building efficiency without degrading the prediction accuracy too much. Typically in this work, we observe that experimental results are given for relatively small dimensionalities (e.g., a few hundreds at the most) and the reduction ratio is not very significant.

In contrast, in mining broad data sets for biomarkers, the problem characteristics are vastly different. Since our data dimensionality is much higher than the number of observations used for training, a drastic dimensionality reduction is not only desirable but imperative. Thus, dimensionality reduction is no longer an optimization issue but rather a necessity. Consequently, efficient solutions to the feature subset selection problem are critical and must scale well with dimensionality.

To the best of our knowledge, most work from the literature does not address the issue of scaling with dimensionality. While many techniques (e.g., those that exhaustively evaluate all dimension subsets) are clearly not scalable, we identified a few that seem promising. In the remainder of this section, we discuss two such techniques, both based on greedy methods, that look interesting.

3.3.1 Stepwise Discriminant Analysis

This technique is worth mentioning because not only it is well known in the statistics community [4] but also it is implemented in many commercial statistical packages. *Stepwise discriminant analysis* (SDA) does not analyze all dimension subsets exhaustively, but rather tries to iteratively modify a candidate dimension subset (starting from the empty set) until no improvement is possible. As sketched in Figure 8, this procedure takes as input a set M of dimensions, a threshold F_{enter} for including a dimension in the candidate solution, a threshold F_{remove} for removal, a desirable size of the solution, and produces a solution dimension subset B. In this figure, $F(S, v)$ denotes a statistic (based on *Wilk's Lambda* [4]) that measures the contribution of dimension v to group

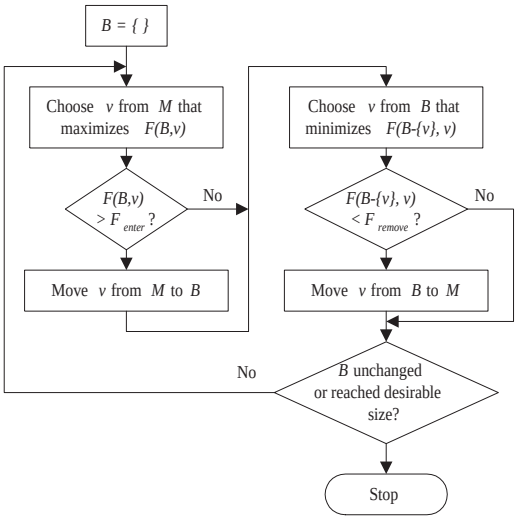


Figure 8. Stepwise discriminant analysis algorithm.

discrimination in addition to the contribution of a given dimension subset S.

Note that in its most general form, SDA performs both inclusion and removal steps at each iteration. A simpler form of SDA, called *forward* SDA, does not have the removal step. This form of SDA is particularly interesting because of its computational efficiency: for a $n \times d$ data set (with n observations and d dimensions) where $n << d$, if we restrict ourselves to subsets of at most n dimensions, the running time complexity of the algorithm is $n^3 d$. The forward SDA method is shown in Figure 9 as the forward arrow.

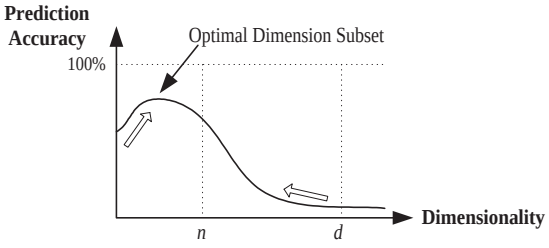


Figure 9. Forward vs. backward search for biomarkers.

However, since the algorithm steps forward greedily, it may be stuck on a path to a suboptimal solution. Keeping the removal step in the general form of SDA may help us undo bad decisions made earlier on, but the main challenge there is how to select appropriate values for the two thresholds so as to keep the number of iterations reasonably bounded. How these thresholds affect the algorithm's behavior and model accuracy is not well understood.

3.3.2 Cross-entropy-based feature elimination

Like other greedy methods, this technique, due to Koller and Sahami [7], starts with the full dimension set and iteratively removes dimensions from the set until no removal is possible. As sketched in Figure 10, this procedure takes as input a set M of dimensions, a parameter K for fine tuning, a desirable size of the solution, and produces a solution dimension subset B. In this figure, S_v represents an "information cover" for v, i.e., a dimension subset to which v adds little additional information, and $E(v, S_v)$ denotes a *cross-entropy* measure [6] that quantifies the amount of information dimension v gives us beyond what S_v already captures. Thus, a dimension is removed from consideration if the information loss caused by the removal is small.

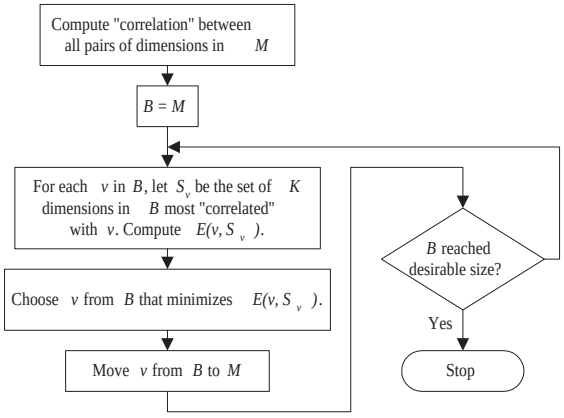


Figure 10. Feature elimination algorithm based on cross-entropy.

What sets this technique apart from the others is the information-theoretic nature of the criterion used for feature selection. The main advantage of using this criterion is that when the number of dimensions is much larger than the number of observations, the concept remains meaningful while the commonly used concept of prediction accuracy runs into difficulties. They also claim that the backward elimination strategy used in their technique is less likely to lead to a suboptimal solution than the commonly used forward inclusion strategy, because intuitively you are trying to preserve information in the full dimension set. Moreover, because the criterion for selecting dimensions does not incorporate any specific learning biases, the dimension subset solution is suitable for building predictive models using a wide range of classification techniques. To implement the algorithm efficiently, Koller and Sahami [7] used an approximation of the cross-entropy criterion: for a $n \times d$ broad dataset, the running time complexity to find a subset of at most n dimensions is $d^2(n + logd)$. The Koller and Sahami method is shown in Figure 9 as the backward arrow.

Before this technique can be useful, several issues re-

main to be addressed. First, computing cross-entropy requires accurate estimates of various probability distributions. Unfortunately, this accuracy can be severely limited by the number of observations available for training, especially when dealing with continuous dimensions which must be discretized. Moreover, while the particular approximation to cross-entropy used in this technique allows an efficient algorithm to be implemented, it can lead to solutions that are suboptimal. Thus, better approximations are desirable, but the challenge is to how to keep the algorithm reasonably efficient. Finally, the number of dimensions we would like to retain should be less than the number of observations which is much less than the original dimensionality. Consequently, since the optimal solution is very "far" from the full dimension set, it is not at all clear whether or not a backward elimination strategy would still be superior to a forward inclusion strategy.

3.4 Implementation of a Data Mining Platform for Biomarker Discovery

Figure 11 illustrates the approach we used to implement a platform for biomarker discovery.

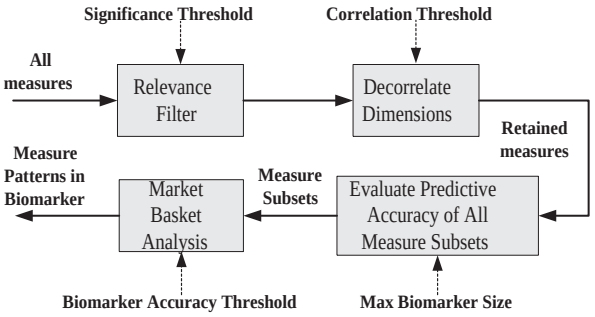


Figure 11. SurroMed's data mining platform for biomarker discovery.

In our approach, we first try to eliminate measures that are redundant or irrelevant to distinguishing samples from different clinical classes, using techniques described in Section 3.1. We then analyze all measure combinations of a given size by building a classifier for each combination and counting the errors made by the classifier. At this point, the user can either select the top few combinations to use as biomarkers, or perform a market basket analysis [1] on all those highly scored combinations to identify any useful measure patterns in biomarkers. In practice, our tool limits the user to the analysis of only small biomarkers, i.e., biomarkers with at most two or three measures. Analyzing larger biomarkers would take a prohibitively long time, unless the initial thresholds are set high enough to reduce the combinatorics. Unfortunately then, as we pointed out in Section 3.1, we may lose many good biomarkers.

4 Future Challenges

We briefly summarizes some of the key technical challenges that remain to be addressed in order to extend current data analysis technologies to the life sciences and perhaps also to other disciplines where controlled scientific experiments are conducted.

Precomputing rank-based aggregations A common approach used in many OLAP systems to speed up aggregate queries is to use materialized subqueries to answer the original queries. This approach assumes that the original queries can be answered using the subqueries. For instance, an `AVG` query using a set S of group-by attributes can be computed as a weighted average over any `AVG` query that uses a superset of S as group-by attributes. However, most rank-based aggregate operators (e.g. `MEDIAN`) are not associative, and the use of materialized queries to optimize queries involving these operators is not obvious.

User-defined percentiles Medians and percentiles do not have a standard definition, especially for even-sized sets of values and bags. Short of providing a generic user-defined aggregation facility, it is not clear how to support all their variant definitions efficiently.

Custom comparative statistics Among the commonly used comparative statistics techniques, many are difficult to express as a composition of SQL aggregate queries. Implementing these techniques requires using sophisticated aggregation mechanisms that can be difficult to provide. For example, traditional aggregation can only reduce a set of values to a single value. To implement user-definable independent group comparative statistics may require using the *powerset aggregation* which would reduce a set of sets of values to a single value. Another aggregation, required for implementing user-definable paired groups comparative statistics, is the *multi-attribute* aggregation where the aggregate operator can take an arbitrary number of arguments.

Large scale visualization Traditional OLAP front-end tools provide a very limited form of visualization: bar charting. Comparing a large number of measures (say in the 10,000's) requires using visualization techniques beyond bar charts that should be both intuitive and compact (see [8] for a survey of visualization techniques used in pharmaceutical research), and that can be implemented efficiently. The challenge is to identify such a powerful and general technique.

Mining broad datasets One of the main unmet challenges is to make feature subset selection algorithms scalable with respect to the number of dimensions. Also, most existing greedy algorithms identify only one solution, which is clearly not adequate since many optimal biomarkers are expected to be found and we would like to identify as many of them as possible. Finally, the

purpose of decoupling feature elimination from feature subset selection is mainly to reduce the combinatorics. Unfortunately, we may never be able to achieve this goal without losing many useful dimensions.

5 Conclusion

The volume of experimental biological data generated in the life sciences is growing at an alarming rate. Yet, well integrated software tools and scalable algorithms for analyzing this data quickly are still underdeveloped. We described two data analysis tasks whose solution is critical to the success of many biotech companies but raises challenges that have yet to be addressed in database and KDD research. Bioinformatics must industrialize, but until we overcome the challenges posed, it will remain a major bottleneck in the quest for new or better disease treatments.

Acknowledgements

We would like to thank Curtis Hastings and Jonathan Heller for helpful comments on an earlier draft of this paper.

References

[1] R. Agrawal, T. Imilienski and A. Swami. Mining association rules between sets of items in large datasets. In *Proc. ACM SIGMOD,* pp. 207–216, Washington, D.C., 1993.

[2] A. Blum and P. Langley. Selection of relevant features and examples in machine learning. In *Artificial Intelligence,* pp. 245–271, 1997.

[3] S. Chaudhuri and U. Dayal. An Overview of Data Warehousing and OLAP Technology. In *SIGMOD Record,* 26(1), pp. 65–74, 1997.

[4] R. I. Jennrich. Stepwise Discriminant Analysis. In *Statistical Methods for Digital Computers,* Vol. 3, K. Enslein, A. Ralston, and H. S. Wilf (Eds.), Wiley, New York, pp. 76–96, 1977.

[5] I. T. Joliffe. *Principal Component Analysis.* Springer-Verlag, 1986.

[6] S. Kullbach and R. A. Leibler. On Information Theory and Sufficiency. In *Annals of Mathematical Statistics,* Vol. 22, 1951.

[7] D. Koller and M. Sahami. Toward Optimal Feature Selection. In *Proc. 13th Int. Conf. on Machine Learning,* Bari, Italy, July 1996, pp. 284–292.

[8] B. Ladd and S. Kenner. Information Visualization and Analytical Data Mining in Pharmaceutical R&D. In *Current Opinion in Drug Discovery & Development*, 3(3), pp. 280–291, 2000.

[9] H. Liu and H. Motada. *Feature Extraction, Construction and Selection: a Data Mining Perspective.* Kluwer, 1998.

[10] Biomarkers Definitions Working Group. *Biomarkers and Endpoints in Clinical Trials: Preferred Definitions and Conceptual Framework.* National Institutes of Health.

[11] S. Norton, P. Huyn, C. Hastings, and J. Heller. Data Mining of Spectroscopic Data for Biomarker Discovery. In *Current Opinion in Drug Discovery & Development*, 4(3), pp. 325–331, 2001.

[12] R. L. Ott. *An Introduction to Statistical Methods and Data Analysis.* Duxbury Press, 1993.

[13] S. D. Patterson. Protein Identification and Characterization by Mass Spectrometry. In *Current Protocols in Molecular Biology*, Wiley, 1998.

[14] J. Ren. High-Throughput Screening of Genetic Mutations/Polymorphisms by Capillary Electrophoresis. In *Combinatorial Chemistry & High Throughput Screening*, 3(1), pp. 11–25, 2000.

[15] S. Tsur. Data Mining in the Bioinformatics Domain. In *Proc. 26th Int. Conf. on Very Large Data Bases,* pp. 711–714, Cairo, Egypt, 2000.

[16] P. Vassiliadis and T. Sellis. A Survey of Logical Models for OLAP Databases. In *SIGMOD Record,* 28(4), pp. 64–69, 1999.

[17] S. Watanabe. *Pattern Recognition: Human and Mechanical.* Wiley, New York, 1985.

[18] H. Wold. Partial Least Squares. In *Encyclopedia of Statistical Sciences,* Samuel Kotz and Norman L. Johnson, eds., Vol. 6, New York: Wiley, 1985, pp. 581–591.