Chapter 2
Survey of Biodata Analysis from a Data Mining Perspective

Peter Bajcsy, Jiawei Han, Lei Liu, and Jiong Yang

Summary

Recent progress in biology, medical science, bioinformatics, and biotechnology has led to the accumulation of tremendous amounts of biodata that demands in-depth analysis. On the other hand, recent progress in data mining research has led to the development of numerous efficient and scalable methods for mining interesting patterns in large databases. The question becomes how to bridge the two fields, *data mining* and *bioinformatics*, for successful mining of biological data. In this chapter, we present an overview of the data mining methods that help biodata analysis. Moreover, we outline some research problems that may motivate the further development of data mining tools for the analysis of various kinds of biological data.

2.1 Introduction

In the past two decades we have witnessed revolutionary changes in biomedical research and biotechnology and an explosive growth of biomedical data, ranging from those collected in pharmaceutical studies and cancer therapy investigations to those identified in genomics and proteomics research by discovering sequential patterns, gene functions, and protein-protein interactions. The rapid progress of biotechnology and biodata analysis methods has led to the emergence and fast growth of a promising new field: *bioinformatics*. On the other hand, recent progress in data mining research has led to the development of numerous efficient and scalable methods for mining interesting patterns and knowledge in large databases, ranging from efficient classification methods to clustering, outlier analysis, frequent, sequential, and structured pattern analysis methods, and visualization and spatial/temporal data analysis tools.

The question becomes how to bridge the two fields, *data mining* and *bioinformatics*, for successful data mining of biological data. In this chapter, we present a general overview of data mining methods that have been successfully applied to biodata analysis. Moreover, we analyze how data mining has helped efficient and effective biomedical data analysis and outline some research problems that may motivate the further development of powerful data mining tools in this field. Our overview is focused on three major themes: (1) data cleaning, data preprocessing, and semantic integration of heterogeneous, distributed biomedical databases, (2) exploration of existing data mining tools for biodata analysis, and (3) development of advanced, effective, and scalable data mining methods in biodata analysis.

- **Data cleaning, data preprocessing, and semantic integration of heterogeneous, distributed biomedical databases**

  Due to the highly distributed, uncontrolled generation and use of a wide variety of biomedical data, data cleaning, data preprocessing, and the semantic integration of heterogeneous and widely distributed biomedical databases, such as genome databases and proteome databases, have become important tasks for systematic and coordinated analysis of biomedical databases. This highly distributed, uncontrolled generation of data has promoted the research and development of integrated data warehouses and distributed federated databases to store and manage different forms of biomedical and genetic data. Data cleaning and data integration methods developed in data mining, such as those suggested in [92, 327], will help the integration of biomedical data and the construction of data warehouses for biomedical data analysis.

- **Exploration of existing data mining tools for biodata analysis**

  With years of research and development, there have been many data mining, machine learning, and statistics analysis systems and tools available for general data analysis. They can be used in biodata exploration and analysis. Comprehensive surveys and introduction of data mining methods have been compiled into many textbooks, such as [165, 171, 431]. Analysis principles are also introduced in many textbooks on bioinformatics, such as [28, 34, 110, 116, 248]. General data mining and data analysis systems that can be used for biodata analysis include SAS Enterprise Miner, SPSS, SPlus, IBM Intelligent Miner, Microsoft SQLServer 2000, SGI MineSet, and Inxight VizServer. There are also many biospecific data analysis software systems, such as GeneSpring, Spot Fire, and VectorNTI. These tools are rapidly evolving as well. A lot of routine data analysis work can be done using such tools. For biodata analysis, it is important to train researchers to master and explore the power of these well-tested and popular data mining tools and packages.

With sophisticated biodata analysis tasks, there is much room for research and development of advanced, effective, and scalable data mining methods in biodata analysis. Some interesting topics follow.

1. **Analysis of frequent patterns, sequential patterns and structured patterns: identification of cooccurring or correlated biosequences or biostructure patterns**

   Many studies have focused on the comparison of one gene with another. However, most diseases are not triggered by a single gene but by a combination of genes acting together. Association and correlation analysis methods can be used to help determine the kinds of genes or proteins that are likely to cooccur in target samples. Such analysis would facilitate the discovery of groups of genes or proteins and the study of interactions and relationships among them. Moreover, since biodata usually contains noise or nonperfect matches, it is important to develop effective sequential or structural pattern mining algorithms in the noisy environment [443].

2. **Effective classification and comparison of biodata**

   A critical problems in biodata analysis is to classify biosequences or structures based on their critical features and functions. For example, gene sequences isolated from diseased and healthy tissues can be compared to identify critical differences between the two classes of genes. Such features can be used for classifying biodata and predicting behaviors. A lot of methods have been developed for biodata classification [171]. For example, one can first retrieve the gene sequences from the two tissue classes and then find and compare the frequently occurring patterns of each class. Usually, sequences occurring more frequently in the diseased samples than in the healthy samples indicate the genetic factors of the disease; on the other hand, those occurring only more frequently in the healthy samples might indicate mechanisms that protect the body from the disease. Similar analysis can be performed on microarray data and protein data to identify similar and dissimilar patterns.

3. **Various kinds of cluster analysis methods**

   Most cluster analysis algorithms are based on either Euclidean distances or density [165]. However, biodata often consist of a lot of features that form a high-dimensional space. It is crucial to study differentials with scaling and shifting factors in multidimensional space, discover pairwise frequent patterns and cluster biodata based on such frequent patterns. One interesting study using microarray data as examples can be found in [421].

4. **Computational modeling of biological networks**

   While a group of genes/proteins may contribute to a disease process, different genes/proteins may become active at different stages of the disease. These genes/proteins interact in a complex network. Large amounts of data generated from microarray and proteomics studies provide rich resources for theoretic study of the complex biological system by computational modeling of biological networks. If the sequence of genetic activities across the different stages of disease development can be identified, it may be possible to develop pharmaceutical interventions that target the different stages separately, therefore achieving more effective treatment of the disease. Such path analysis is expected to play an important role in genetic studies.

5. **Data visualization and visual data mining**

   Complex structures and sequencing patterns of genes and proteins are most effectively presented in graphs, trees, cubes, and chains by various kinds of visualization tools. Visually appealing structures and patterns facilitate pattern understanding, knowledge discovery, and interactive data exploration. Visualization and visual data mining therefore play an important role in biomedical data mining.

2.2 Data Cleaning, Data Preprocessing, and Data Integration

Biomedical data are currently generated at a very high rate at multiple geographically remote locations with a variety of biomedical devices and by applying several data acquisition techniques. All bioexperiments are driven by a plethora of experimental design hypotheses to be proven or rejected based on data values stored in multiple distributed biomedical databases, for example, genome or proteome databases. To extract and analyze the data perhaps poses a much bigger challenge for researchers than to generate the data [181]. To extract and analyze information from distributed biomedical databases, distributed heterogeneous data must be gathered, characterized, and cleaned. These processing steps can be very time-consuming if they require multiple scans of large distributed databases to ensure the data quality defined by biomedical domain experts and computer scientists. From a semantic integration viewpoint, there are quite often challenges due to the heterogeneous and distributed nature of data since these preprocessing steps might require the data to be transformed (e.g., log ratio transformations), linked with distributed annotation or metadata files (e.g., microarray spots and gene descriptions), or more exactly specified using auxiliary programs running on a remote server (e.g., using one of the BLAST programs to identify a sequence match). Based on the aforementioned data quality and

integration issues, the need for using automated preprocessing techniques becomes eminent. We briefly outline the strategies for taming the data by describing data cleaning using exploratory data mining (EDM), data preprocessing, and semantic integration techniques [91, 165].

2.2.1 Data Cleaning

Data cleaning is defined as a preprocessing step that ensures data quality. In general, the meaning of data quality is best described by the data interpretability. In other words, if the data do not mean what one thinks, the data quality is questionable and should be evaluated by applying data quality metrics. However, defining data quality metrics requires understanding of data gathering, delivery, storage, integration, retrieval, mining, and analysis. Data quality problems can occur in any data operation step (also denoted as a lifecycle of the data) and their corresponding data quality continuum (end-to-end data quality). Although conventional definitions of data quality would include accuracy, completeness, uniqueness, timeliness, and consistency, it is very hard to quantify data quality by using quality metrics. For example, measuring accuracy and completeness is very difficult because each datum would have to be tested for its correctness against the "true" value and all data values would have to be assessed against all relevant data values. Furthermore, data quality metrics should measure data interpretability by evaluating meanings of variables, relationships between variables, miscellaneous metadata information and consistency of data.

In the biomedical domain, the data quality continuum involves answering a few basic questions.

1. How do the data enter the system? The answers can vary a lot because new biomedical technologies introduce varying measurement errors and there are no standards for data file formats. Thus, the standardization efforts are important for data quality, for instance, the Minimum Information About a Microarray Experiment (MIAME) [51] and MicroArray and Gene Expression (MAGE) [381] standardization efforts for microarray processing, as well as, preemptive (process management) and retrospective (cleaning and diagnostic) data quality checks.
2. How are the data delivered? In the world of electronic information and wireless data transfers, data quality issues include transmission losses, buffer overflows, and inappropriate preprocessing, such as default value conversions or data aggregations. These data quality issues have to be addressed by verifying checksums or relationships between data streams and by using reliable transmission protocols.
3. Where do the data go after being received? Although physical storage may not be an issue anymore due to its low cost, data storage can encounter problems with poor accompanying metadata, missing

time stamps, or hardware and software constraints, for instance, data dissemination in Excel spread sheets stored on an Excel-unsupported platform. The solution is frequently thorough planning followed by publishing data specifications.

4. Are the data combined with other data sets? The integration of new data sets with already archived data sets is a challenge from the data quality viewpoint since the data might be heterogeneous (no common keys) with different variable definitions of data structures (e.g., legacy data and federated data) and time asynchronous. In the data mining domain, a significant number of research papers have addressed the issue of dataset integrations, and the proposed solutions involve several matching and mapping approaches. In the biomedical domain, data integration becomes essential, although very complex, for understanding a whole system. Data are generated by multiple laboratories with various devices and data acquisition techniques while investigating a broad range of hypotheses at multiple levels of system ontology.
5. How are the data retrieved? The answers to this question should be constructed with respect to the computational resources and users' needs. Retrieved data quality will be constrained by the retrieved data size, access speed, network traffic, data and database software compatibility, and the type and correctness of queries. To ensure data quality, one has to plan ahead to minimize the constraints and select appropriate tools for data browsing and exploratory data mining (EDM) [92, 327].
6. How are the data analyzed? In the final processing phase, data quality issues arise due to insufficient biomedical domain expertise, inherent data variability, and lack of algorithmic scalability for large datasets [136]. As a solution, any data mining and analysis should be an interdisciplinary effort because the computer science models and biomedical models have to come together during exploratory types of analyses [323]. Furthermore, conducting continuous analyses and cross-validation experiments will lead to confidence bounds on obtained results and should be used in a feedback loop to monitor the inherent data variability and detect related data quality problems.

The steps of microarray processing from start to finish that clearly map to the data quality continuum are outlined in [181].

2.2.2 Data Preprocessing

What can be done to ensure biomedical data quality and eliminate sources of data quality corruption for both data warehousing and data mining? In general, multidisciplinary efforts are needed, including (1) process management, (2) documentation of biomedical domain expertise, and (3) statistical and database analyses [91]. Process management in the biomedical domain should support standardization of content and format [51, 381],

automation of preprocessing, e.g., microarray spot analysis [26, 28, 150], introduction of data quality incentives (correct data entries and quality feedback loops), and data publishing to obtain feedback (e.g., via MedLine and other Internet sites). Documenting biomedical domain knowledge is not a trivial task and requires establishing metadata standards (e.g., a document exchange format MAGE-ML), creating annotation files, and converting biomedical and engineering logs into metadata files that accompany every experiment and its output data set. It is also necessary to develop text-mining software to browse all documented and stored files [439]. In terms of statistical and database analyses for the biomedical domain, the focus should be on quantitative quality metrics based on analytical and statistical data descriptors and on relationships among variables.

Data preprocessing using statistical and database analyses usually includes data cleaning, integration, transformation, and reduction [165]. For example, an outcome of several spotted DNA microarray experiments might be ambiguous (e.g., a background intensity is larger than a foreground intensity) and the missing values have to be filled in or replaced by a common default value during data cleaning. The integration of multiple microarray gene experiments has to resolve inconsistent labels of genes to form a coherent data store. Mining microarray experimental data might require data normalization (transformation) with respect to the same control gene and a selection of a subset of treatments (data reduction), for instance, if the data dimensionality is prohibitive for further analyses. Every data preprocessing step should include static and dynamic constraints, such as foreign key constraints, variable bounds defined by dynamic ranges of measurement devices, or experimental data acquisition and processing workflow constraints. Due to the multifaceted nature of biomedical data measuring complex and context-dependent biomedical systems, there is no single recommended data quality metric. However, any metric should serve operational or diagnostic purpose and should change regularly with the improvement of data quality. For example, the data quality metrics for extracted spot information can be clearly defined in the case of raw DNA microarray data (images) and should depend on (a) spot to background separation and (b) spatial and topological variations of spots. Similarly, data quality metrics can be defined at other processing stages of biomedical data using outlier detection (geometric, distributional, and time series outliers), model fitting, statistical goodness of fit, database duplicate finding, and data type checks and data value constraints.

2.2.3 Semantic Integration of Heterogeneous Data

One of the many complex aspects in biomedical data mining is semantic integration. Semantic integration combines multiple sources into a coherent data store and involves finding semantically equivalent real-world entities from several biomedical sources to be matched up. The problem arises when,

for instance, the same entities do not have identical labels, such as, gene_id and g_id, or are time asynchronous, as in the case of the same gene being analyzed at multiple developmental stages. There is a theoretical foundation [165] for approaching this problem by using correlation analysis in a general case. Nonetheless, semantic integration of biomedical data is still an open problem due to the complexity of the studied matter (bioontology) and the heterogeneous distributed nature of the recorded high-dimensional data.

Currently, there are in general two approaches: (1) construction of *integrated* biodata warehouses or biodatabases and (2) construction of a *federation* of heterogeneous distributed biodatabases so that query processing or search can be performed in multiple heterogeneous biodatabases. The first approach performs data integration beforehand by data cleaning, data preprocessing, and data integration, which requires common ontology and terminology and sophisticated data mapping rules to resolve semantic ambiguity or inconsistency. The integrated data warehouses or databases are often multidimensional in nature, and indexing or other data structures can be built to assist a search in multiple lower-dimensional spaces. The second approach is to build up mapping rules or semantic ambiguity resolution rules across multiple databases. A query posed at one site can then be properly mapped to another site to retrieve the data needed. The retrieved results can be appropriately mapped back to the query site so that the answer can be understood with the terminology used at the query site. Although a substantial amount of work has been done in the field of database systems [137], there are not enough studies of systems in the domain of bioinformatics, partly due to the complexity and semantic heterogeneity of biodata. We believe this is an important direction of future research.

2.3 Exploration of Existing Data Mining Tools for Biodata Analysis

With years of research and development, there have been many data mining, machine learning, and statistical analysis systems and tools available for use in biodata exploration and analysis. Comprehensive surveys and the introduction of data mining methods have been compiled into many textbooks [165, 171, 258, 281, 431]. There are also many textbooks focusing exclusively on bioinformatics [28, 34, 110, 116, 248]. Based on the theoretical descriptions of data mining methods, many general data mining and data analysis systems have been built and widely used for necessary analyses of biodata, e.g., SAS Enterprise Miner, SPSS, SPlus, IBM Intelligent Miner, Microsoft SQLServer 2000, SGI MineSet, and Inxight VizServer. In this section, we briefly summarize the different types of existing software tools developed specifically for solving the fundamental bioinformatics problems. Tables 2.1 and 2.2 provide a list of a few software tools and their Web links.

Table 2.1. Partial list of bioinformatics tools and software links. These tools were chosen based on authors' familiarity. We recognize that there are many other popular tools.

Sequence analysis
NCBI/BLAST: http://www.ncbi.nih.gov/BLAST
ClustalW (multi-sequence alignment): http://www.ebi.ac.uk/clustalw/
HMMER: http://hmmer.wustl.edu/
PHYLIP: http://evolution.genetics.washington.edu/phylip.html
MEME (motif discovery and search): http://meme.sdsc.edu/meme/website/
TRANSFAC: http://www.cbrc.jp/research/db/TFSEARCH.html
MDScan: http://bioprospector.stanford.edu/MDscan/
VectorNTI: http://www.informax.com
Sequencher: http://www.genecodes.com/
MacVector: http://www.accelrys.com/products/macvector/

Structure prediction and visualization
RasMol: http://openrasmol.org/
Raster3D: http://www.bmsc.washington.edu/raster3d/raster3d.html
Swiss-Model: http://www.expasy.org/swissmod/
Scope: http://scop.mrc-lmb.cam.ac.uk/scop/
MolScript: http://www.avatar.se/molscript/
Cn3D: http://www.ncbi.nlm.nih.gov/Structure/CN3D/cn3d.shtml

2.3.1 DNA and Protein Sequence Analysis

Sequence comparison, similarity search, and pattern finding are considered the basic approaches to protein sequence analysis in bioinformatics. The mathematical theory and basic algorithms of sequence analysis can be dated to 1960s when the pioneers of bioinformatics developed methods to predict phylogenetic relationships of the related protein sequences during evolution [281]. Since then, many statistical models, algorithms, and computation techniques have been applied to protein and DNA sequence analysis.

Table 2.2. Partial list of bioinformatics tools and software links.

Genome analysis
PHRED/PHRAP: http://www.phrap.org/
CAP3: http://deepc2.zool.iastate.edu/aat/cap/cap.html
Paracel GenomeAssembler: http://www.paracel.com/products/paracel_genomeassembler.php
GenomeScan: http://genes.mit.edu/genomescan.html
GeneMark: http://opal.biology.gatech.edu/GeneMark/
GenScan: http://genes.mit.edu/GENSCAN.html
X-Grail: http://compbio.ornl.gov/Grail-1.3/
ORF Finder: http://www.ncbi.nlm.nih.gov/gorf/gorf.html
GeneBuilder: http://l25.itba.mi.cnr.it/ webgene/genebuilder.html

Pathway analysis and visualization
KEGG: http://www.genome.ad.jp/kegg/
EcoCyc/MetaCyc: http://metacyc.org/
GenMapp: http://www.genmapp.org/

Microarray analysis
ScanAlyze/Cluster/TreeView: http://rana.lbl.gov/EisenSoftware.htm
Scanalytics: MicroArray Suite: http://www.scanalytics.com/product/microarray/index.shtmlExpression
Profiler (Jaak Vilo, EBI): http://ep.ebi.ac.uk/EP/
Knowledge-based analysis of microarray gene expression data using SVM: http://www.cse.ucsc.edu/research/compbio/genex/genex.html
Silicon Genetics - gene expression software: http://www.sigenetics.com/cgi/SiG.cgi/index.smf

Most sequence alignment tools were based on a dynamic programming algorithm [373], including pairwise alignment tools such as the Basic Local Alignment Search Tool (BLAST) [12] and multiple sequence alignment tools such as ClustalW [176]. A series of tools was developed to construct phylogenetic trees based on various probability models and sequence alignment principles. Many of the phylogenetic tools have been packaged into software packages, such as PHYLIP and PAUP* [124]. Hidden Markov models

(HMM) is another widely used algorithm especially in (1) protein family studies, (2) identification of protein structural motifs, and (3) gene structure prediction (discussed later). HMMER, which is used to find conserved sequence domains in a set of related protein sequences and the spacer regions between them, is one of the popular HMM tools.

Other challenging search problems include promoter search and protein functional motif search. Several probability models and stochastic methods have been applied to these problems, including expectation maximization (EM) algorithms and Gibbs sampling methods [28].

2.3.2 Genome Analysis

Sequencing of a complete genome and subsequent annotation of the features in the genome pose different types of challenges. First, how is the whole genome put together from many small pieces of sequences? Second, where are the genes located on a chromosome? The first problem is related to genome mapping and sequence assembly. Researchers have developed software tools to assemble a large number of sequences using similar algorithms to the ones used in the basic sequence analysis. The widely used algorithms include PHRAP/Consed and CAP3 [188].

The other challenging problem is related to prediction of gene structures, especially in eukaryotic genomes. The simplest way to search for a DNA sequence that encodes a protein is to search for open reading frames (ORFs). Predicting genes is generally easier and more accurate in prokaryotic than eukaryotic organisms. The eukaryotic gene structure is much more complex due to the intron/exon structure. Several software tools, such as GeneMark [48] and Glimmer [343], can accurately predict genes in prokaryotic genomes using HMM and other Markov models. Similar methodologies were used to develop eukaryotic gene prediction tools such as GeneScan [58] and GRAIL [408].

2.3.3 Macromolecule Structure Analysis

Macromolecule structure analysis involves (1) prediction of secondary structure of RNA and proteins, (2) comparison of protein structures, (3) protein structure classification, and (4) visualization of protein structures. Some of the most popular software tools include DALI for structural alignment, Cn3d and Rasmol for viewing the 3D structures, and Mfold for RNA secondary structure prediction. Protein structure databases and associated tools also play an important role in structure analysis. Protein Data Bank (PDB), the classification by class, architecture, topology, and homology (CATH) database, the structural classification of proteins (SCOP) database, Molecular Modeling Database (MMDB), and Swiss-Model resource are among the best protein structure resources. Structure prediction is still

an unsolved, challenging problem. With the rapid development of proteomics and high throughput structural biology, new algorithms and tools are very much needed.

2.3.4 Pathway Analysis

Biological processes in a cell form complex networks among gene products. Pathway analysis tries to build, model, and visualize these networks. Pathway tools are usually associated with a database to store the information about biochemical reactions, the molecules involved, and the genes. Several tools and databases have been developed and are widely used, including KEGG database (the largest collection of metabolic pathway graphs), EcoCyc/MetaCyc [212] (a visualization and database tool for building and viewing metabolic pathways), and GenMAPP (a pathway building tool designed especially for working with microarray data). With the latest developments in functional genomics and proteomics, pathway tools will become more and more valuable for understanding the biological processes at the system level (section 2.7).

2.3.5 Microarray Analysis

Microarray technology allows biologists to monitor genome-wide patterns of gene expression in a high-throughput fashion. Applications of microarrays have resulted in generating large volumes of gene expression data with several levels of experimental data complexity. For example, a "simple" experiment involving a 10,000-gene microarray with samples collected at five time points for five treatments with three replicates can create a data set with 0.75 million data points! Historically, hierarchical clustering [114] was the first clustering method applied to the problem of finding similar gene expression patterns in microarray data. Since then many different clustering methods have been used [323], such as k-means, a self-organizing map, a support vector machine, association rules, and neural networks. Several commercial software packages, e.g., GeneSpring or Spotfire, offer the use of these algorithms for microarray analysis.

Today, microarray analysis is far beyond clustering. By incorporating a priori biological knowledge, microarray analysis can become a powerful method for modeling a biological system at the molecular level. For example, combining sequence analysis methods, one can identify common promoter motifs from the clusters of coexpressed genes in microarray data using various clustering methods. Furthermore, any correlation among gene expression profiles can be modeled by artificial neural networks and can hopefully reverse-engineer the underlying genetic network in a cell (section 2.7).

2.4 Discovery of Frequent Sequential and Structured Patterns

Frequent pattern analysis has been a focused theme of study in data mining, and a lot of algorithms and methods have been developed for mining frequent patterns, sequential patterns, and structured patterns [6, 165, 437, 438]. However, not all the frequent pattern analysis methods can be readily adopted for the analysis of complex biodata because many frequent pattern analysis methods are trying to discover "perfect" patterns, whereas most biodata patterns contain a substantial amount of noise or faults. For example, a DNA sequential pattern usually allows a nontrivial number of insertions, deletions, and mutations. Thus our discussion here is focused on sequential and structured pattern mining potential adaptable to noisy biodata instead of a general overview of frequent pattern mining methods.

In bioinformatics, the discovery of frequent sequential patterns (such as motifs) and structured patterns (such as certain biochemical structures) could be essential to the analysis and understanding of the biological data. If a pattern occurs frequently, it ought to be important or meaningful in some way. Much work has been done on discovery of frequent patterns in both sequential data (unfolded DNA, proteins, and so on) and structured data (3D model of DNA and proteins).

2.4.1 Sequential Pattern

Frequent sequential pattern discovery has been an active research area for years. Many algorithms have been developed and deployed for this purpose. One of the most popular pattern (motif) discovery methods is BLAST [12], which is essentially a pattern matching algorithm. In nature, amino acids (in protein sequences) and nucleotides (in DNA sequences) may mutate. Some mutations may occur frequently while others may not occur at all. The *mutation scoring matrix* [110] is used to measure the likelihood of the mutations.

Figure 2.1 is one of the scoring matrices. The entry associated with row A_i and column A_j is the score for an amino acid A_i mutating to A_j. For a given protein or DNA sequence S, BLAST will find all similar sequences S' in the database such that the aggregate mutation score from S to S' is above some user-specified threshold. Since an amino acid may mutate to several others, if all combinations need to be searched, the search time may grow exponentially. To reduce the search time, BLAST partitions the query sequence into small segments (3 amino acids for a protein sequence and 11 nucleotides for DNA sequences) and searches for the exact match on the small segments and stitches the segments back up after the search. This technique can reduce the search time significantly and yield satisfactory results (close to 90% accuracy).

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
A	5	-2	-1	-2	-1	-1	-1	0	-2	-1	-2	-1	-1	-3	-1	1	0	-3	-2	0
R	-2	7	-1	-2	-4	1	0	-3	0	-4	-3	3	-2	-3	-3	-1	-1	-3	-1	-3
N	-1	-1	7	2	-2	0	0	0	1	-3	-4	0	-2	-4	-2	1	0	-4	-2	-3
D	-2	-2	2	8	-4	0	2	-1	-1	-4	-4	-1	-4	-5	-1	0	-1	-5	-3	-4
C	-1	-4	-2	-4	13	-3	-3	-3	-3	-2	-2	-3	-2	-2	-4	-1	-1	-5	-3	-1
Q	-1	1	0	0	-3	7	2	-2	1	-3	-2	2	0	-4	-1	0	-1	-1	-1	-3
E	-1	0	0	2	-3	2	6	-3	0	-4	-3	1	-2	-3	-1	-1	-1	-3	-2	-3
G	0	-3	0	-1	-3	-2	-3	8	-2	-4	-4	-2	-3	-4	-2	0	-2	-3	-3	-4
H	-2	0	1	-1	-3	1	0	-2	10	-4	-3	0	-1	-1	-2	-1	-2	-3	2	-4
I	-1	-4	-3	-4	-2	-3	-4	-4	-4	5	2	-3	2	0	-3	-3	-1	-3	-1	4
L	-2	-3	-4	-4	-2	-2	-3	-4	-3	2	5	-3	3	1	-4	-3	-1	-2	-1	1
K	-1	3	0	-1	-3	2	1	-2	0	-3	-3	6	-2	-4	-1	0	-1	-3	-2	-3
M	-1	-2	-2	-4	-2	0	-2	-3	-1	2	3	-2	7	0	-3	-2	-1	-1	0	1
F	-3	-3	-4	-5	-2	-4	-3	-4	-1	0	1	-4	0	8	-4	-3	-2	1	4	-1
P	-1	-3	-2	-1	-4	-1	-1	-2	-2	-3	-4	-1	-3	-4	10	-1	-1	-4	-3	-3
S	1	-1	1	0	-1	0	-1	0	-1	-3	-3	0	-2	-3	-1	5	2	-4	-2	-2
T	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	2	5	-3	-2	0
W	-3	-3	-4	-5	-5	-1	-3	-3	-3	-3	-2	-3	-1	1	-4	-4	-3	15	2	-3
Y	-2	-1	-2	-3	-3	-1	-2	-3	2	-1	-1	-2	0	4	-3	-2	-2	2	8	-1
V	0	-3	-3	-4	-1	-3	-3	-4	-4	4	1	-3	1	-1	-3	-2	0	-3	-1	5

Fig. 2.1. BLOSUM 50 mutation scoring matrix.

Tandem repeat (TR) detection is one of the active research areas. A tandem repeat is a segment that occurs more than a certain number of times within a DNA sequence. If a pattern repeats itself a significant number of times, biologists believe that it may signal some importance. Due to the presence of noise, the actual occurrences of the pattern may be different. In some occurrences the pattern may be shortened—some nucleotide is missing—while in other occurrences the pattern may be lengthened—a noise nucleotide is added. In addition, the occurrence of a pattern may not follow a fixed period. Several methods have been developed for finding tandem repeats. In [442], the authors proposed a dynamic programming algorithm to find all possible asynchronous patterns, which allows a certain type of imperfection in the pattern occurrences. The complexity of this algorithm is $O(N^2)$ where N is the length of the sequence.

The number of amino acids in a protein sequence is around several hundred. It is useful to find some segments that appear in a number of proteins. As mentioned, the amino acid may mutate without changing its biological functions. Thus, the occurrences of a pattern may be different. In [443], the authors proposed a model that takes into account the mutations of amino acids. A mutation matrix is constructed to represent the likelihood of mutation. The entry at row i and column j is the probability for amino acid i to mutate to j. For instance, assume there is a segment $ACCD$ in a protein. The probability that it is mutated from $ABCD$ is $Prob(A|A) \times Prob(C|B) \times Prob(C|C) \times Prob(D|D)$. This probability can be viewed as

the expected chance of occurrences of the pattern $ABCD$ given that the protein segment $ACCD$ is observed. The mutation matrix serves as a bridge between the observations (protein sequences) and the true underlying models (frequent patterns). The overall occurrence of a pattern is the aggregated expected number of occurrences of the pattern in all sequences. A pattern is considered frequent if its aggregated expected occurrences are over a certain threshold. In addition, [443] also proposed a probabilistic algorithm that can find all frequent patterns efficiently.

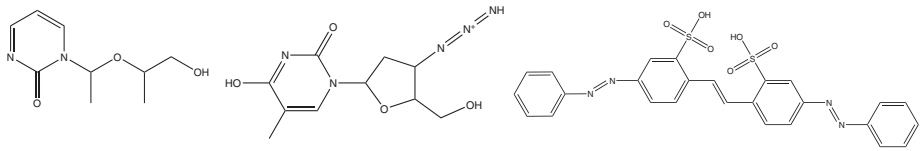
2.4.2 Mining Structured Patterns in Biodata

Besides finding sequential patterns, many biodata analysis tasks need to find frequent structured patterns, such as frequent protein or chemical compound structures from large biodata sets. This promotes research into efficient mining of frequent structured patterns. Two classes of efficient methods for mining structured patterns have been developed: one is based on the apriori-like candidate generation and test approach [6], such as FSG [234], and the other is based on a frequent pattern growth approach [166] by growing frequent substructure patterns and reducing the size of the projected patterns, such as gSpan [436]. A performance study in [436] shows that a gSpan-based method is much more efficient than an FSG-based method.

Mining substructure patterns may still encounter difficulty in both the huge number of patterns generated and mining efficiency. Since a frequent large structure implies that all its substructures must be frequent as well, mining frequent large, structured patterns may lead to an exponential growth of search space because it would first find all the substructure patterns. To overcome this difficulty, a recent study in [437] proposes to mine only closed subgraph patterns rather than all subgraph patterns, where a subgraph G is *closed* if there exists no supergraph G' such as $G \subset G'$ and $support(G) = support(G')$ (i.e., they have the same occurrence frequency). The set of closed subgraph patterns has the same expressive power of the set of all subgraph patterns but is often orders of magnitude more compact than the latter in dense graphs. An efficient mining method called *CloseGraph* has been developed in [437], which also demonstrates order-of-magnitude performance gain in comparison with gSpan.

Figure 2.2 shows the discovered closed subgraph patterns for class CA compounds from the AIDS antiviral screen compound dataset of the Developmental Therapeutics Program of NCI/NIH (March 2002 release). One can see that by lowering the minimum support threshold (i.e., occurrence frequency), larger chemical compounds can be found in the dataset.

Such structured pattern mining methods can be extended to other data mining tasks, such as discovering structure patterns with angles or geometric constraints, finding interesting substructure patterns in a noisy environment, or classifying data [99]. For example, one can use the discovered structure patterns to distinguishing AIDS tissues from healthy ones.



(a) min_supp = 20% (b) min_sup = 10% (c) min_supp = 5%

Fig. 2.2. Discovered substructures from an antiviral screen compound dataset.

2.5 Classification Methods

Each biological object may consist of multiple attributes. The relationship/ interaction among these attributes could be very complicated. In bioinformatics, classification is one of the popular tools for understanding the relationships among various conditions and the features of various objects. For instance, there may be a training dataset with two classes of cells, normal cells and cancer cells. It is very important to classify these cells so that when a new cell is obtained, it can be automatically determined whether it is cancerous. Classification has been an essential theme in statistics, data mining, and machine learning, with many methods proposed and studied [165, 171, 275, 431]. Typical methods include decision trees, Bayesian classification, neural networks, support vector machines (SVMs), the k-nearest neighbor (KNN) approach, associative classification, and so on. We briefly describe three methods: SVM, decision tree induction, and KNN.

The support vector machine (SVM) [59] has been one of the most popular classification tools in bioinformatics. The main idea behind SVM is the following. Each object can be mapped as a point in a high-dimensional space. It is possible that the points of the two classes cannot be separated by a hyperplane in the original space. Thus, a transformation may be needed. These points may be transformed to a higher dimensional space so that they can be separated by a hyperplane. The transformation may be complicated. In SVM, the kernel is introduced so that computing the separation hyperplane becomes very fast. There exist many kernels, among which three are the most popular: *linear kernel, polynomial kernel*, and *Gaussian kernel* [353]. SVM usually is considered the most accurate classification tool for many bioinformatics applications. However, there is one drawback: the complexity of training an SVM is $O(N^2)$ where N is the number of objects/points. There are recent studies, such as [444], on how to scale up SVMs for large datasets. When handling a large number of datasets, it is necessary to explore scalable SVM algorithms for effective classification.

Another popularly used classifier is the decision-tree classifier [171, 275]. When the number of dimensions is low, i.e., when there exist only a small number of attributes, the accuracy of the decision tree is comparable to that of SVM. A decision tree can be built in linear time with respect to the

number of objects. In a decision tree, each internal node is labeled with a list of ranges. A range is then associated with a path to a child. If the attribute value of an object falls in the range, then the search travels down the tree via the corresponding path. Each leaf is associated with a class label. This label will be assigned to the objects that fall in the leaf node. During the decision tree construction, it is desirable to choose the most distinctive features or attributes at the high levels so that the tree can separate the two classes as early as possible. Various methods have been tested for choosing an attribute. The decision tree may not perform well with high-dimensional data.

Another method for classification is called *k-nearest neighbor* (KNN) [171]. Unlike the two preceding methods, the KNN method does not build a classifier on the training data. Instead, when a test object arrives, it searches for the k neighboring points closest to the test object and uses their labels to label the new object. If there are conflicts among the neighboring labels, a majority voting algorithm is applied. Although this method does not incur any training time, the classification time may be expensive since finding KNN in a high-dimensional space is a nontrivial task.

2.6 Cluster Analysis Methods

Clustering is a process that groups a set of objects into *clusters* so that the similarity among the objects in the same cluster is high, while that among the objects in different clusters is low. Clustering has been popular in pattern recognition, marketing, social and scientific studies, as well as in biodata analysis. Effective and efficient cluster analysis methods have also been studied extensively in statistics, machine learning, and data mining, with many approaches proposed [165, 171], including k-means, k-medoids, SOM, hierarchical clustering (such as DIANA [216], AGNES [216], BIRCH [453], and Chameleon [215]), a density-based approach (such as Optics [17]), and a model-based approach. In this section, we introduce two recently proposed approaches for clustering biodata: (1) clustering microarray data by biclustering or p-clustering, and (2) clustering biosequence data.

2.6.1 Clustering Microarray Data

Microarray has been a popular method for representing biological data. In the microarray gene expression dataset, each column represents a condition, e.g., arobetic, acid, and so on. Each row represents a gene. An entry is the expression level of the gene under the corresponding condition. The expression level of some genes is low across all the conditions while others have high expression levels. The absolute expression level may be a good indicator not of the similarity among genes but of the fluctuation of the expression levels. If the genes in a set exhibit similar fluctuation under all

conditions, these genes may be coregulated. By discovering the coregulation, we may be able to refer to the gene regulative network, which may enable us to better understand how organisms develop and evolve. Row clustering [170] is proposed to cluster genes that exhibit similar behavior or fluctuation across all the conditions.

However, clustering based on the entire row is often too restricted. It may reveal the genes that are very closely coregulated. However, it cannot find the weakly regulated genes. To relax the model, the concept of *bicluster* was introduced in [74]. A *bicluster* is a subset of genes and conditions such that the subset of genes exhibits similar fluctuations under a given subset of conditions. The similarity among genes is measured as the squared mean residue error. If the similarity measure (squared mean residue error) of a matrix satisfies a certain threshold, it is a bicluster. Although this model is much more flexible than the row clusters, the computation could be costly due to the absence of pruning power in the bicluster model. It lacks the *downward closure property* typically associated with frequent patterns [165]. In other words, if a supermatrix is a bicluster, none of its submatrixes is necessarily a bicluster. As a result, one may have to consider all the combinations of columns and rows to identify all the biclusters. In [74], a nondeterministic algorithm is devised to discover one bicluster at a time. After a bicluster is discovered, its entries will be replaced by random value and a new bicluster will be searched for in the updated microarray dataset. In this scheme, it may be difficult to discover the overlapped cluster because some important value may be replaced by random value. In [441], the authors proposed a new algorithm that can discover the overlapped biclusters.

Bicluster uses squared mean residue error as the indicator of similarity among a set of genes. However, this leads to a problem: For a set of genes that are highly similar, the squared mean residue error could still be high. Even after including a new random gene in the cluster, the resulting cluster should also have high correlation; as a result, it may still qualify as a bicluster. To solve this problem, the authors of [421] proposed a new model, called *p-clusters*. In the p-cluster model, it is required that any 2-by-2 submatrix (two genes and two conditions) $[x_{11}, x_{12}, y_{11}, y_{12}]$ of a p cluster satisfies the formula $|(x_{11} - x_{12}) - (y_{11} - y_{12})| \leq \delta$ where δ is some specified threshold. This requirement is able to remove clusters that are formed by some strong coherent genes and some random genes. In addition, a novel two-way pruning algorithm is proposed, which enables the cluster discovery process be carried out in a more efficient manner on average [421].

2.6.2 Clustering Sequential Biodata

Biologists believe that the functionality of a gene depends largely on its layout or the sequential order of amino acids or nucleotides. If two genes or proteins have similar components, their functionality may be similar. Clustering the biological sequences according to their components may

reveal the biological functionality among the sequences. Therefore, clustering sequential data has received a significant amount of attention recently. The foundation of any clustering algorithm is the measure of similarity between two objects (sequences). Various measurements have been proposed. One possible approach is the use of *edit distance* [160] to measure the distance between each pair of sequences. This solution is not ideal because, in addition to its inefficiency in calculation, the edit distance captures only the optimal global alignment between a pair of sequences; it ignores many other local alignments that often represent important features shared by the pair of sequences. Consider the three sequences *aaaabbb*, *bbbaaaa*, and *abcdefg*. The edit distance between *aaaabbb* and *bbbaaaa* is 6 and the edit distance between *aaaabbb* and *abcdefg* is also 6, to a certain extent contradicting the intuition that *aaaabbb* is more similar to *bbbaaaa* than to *abcdefg*. These overlooked features may be very crucial in producing meaningful clusters. Even though allowing *block operations*[1] [258, 291] may alleviate this weakness to a certain degree, the computation of edit distance with block operations is NP-hard [291]. This limitation of edit distance, in part, has motivated researchers to explore alternative solutions.

Another approach that has been widely used in document clustering is the keyword-based method. Instead of being treated as a sequence, each text document is regarded as a set of keywords or phrases and is usually represented by a weighted word vector. The similarity between two documents is measured based on keywords and phrases they share and is often defined in some form of normalized dot-product. A direct extension of this method to generic symbol sequences is to use short segments of fixed length q (generated using a sliding window through each sequence) as the set of "words" in the similarity measure. This method is also referred to in the literature [154] as the q-gram based method. While the q-gram based approach enables significant segments (i.e., keywords/phrases/q grams) to be identified and used to measure the similarity between sequences regardless of their relative positions in different sequences, valuable information may be lost as a result of ignoring sequential relationship (e.g., ordering, correlation, dependency, and so on) among these segments, which impacts the quality of clustering.

Recently statistics properties of sequence construction were used to assess the similarity among sequences in a sequence clustering system, CLUSEQ [441]. Sequences belonging to one cluster may subsume to the same probability distribution of symbols (conditioning on the preceding segment of a certain length), while different clusters may follow different underlying probability distributions. This feature, typically referred to as *short memory*, which is common to many applications, indicates that, for a certain sequence, the empirical probability distribution of the next symbol given the preceding segment can be accurately approximated by observing

[1] A consecutive block can be inserted/deleted/shifted/reversed in a sequence with a constant cost with regard to the edit distance.

no more than the last L symbols in that segment. Significant features of such probability distribution can be very powerful in distinguishing different clusters. By extracting and maintaining significant patterns characterizing (potential) sequence clusters, one could easily determine if a sequence should belong to a cluster by calculating the likelihood of (re)producing the sequence under the probability distribution that characterizes the given cluster. To support efficient maintenance and retrieval of the probability entries,[2] a novel variation of the suffix tree [157], namely the *probabilistic suffix tree* (PST), is proposed in [441], and it is employed as a compact representation for organizing the derived (conditional) probability distribution for a cluster of sequences. A probability vector is associated with each node to store the probability distribution of the next symbol given the label of the node as the preceding segment. These innovations enable the similarity estimation to be performed very fast, which offers many advantages over alternative methods and plays a dominant role in the overall performance of the clustering algorithm.

2.7 Computational Modeling of Biological Networks

Computational modeling of biological networks has gained much of its momentum as a result of the development of new high-throughput technologies for studying gene expressions (e.g., microarray technology) and proteomics (e.g., mass spectrometry, 2D protein gel, and protein chips). Large amounts of data generated by gene microarray and proteomics technologies provide rich resources for theoretic study of the complex biological system. Recent advances in this field have been reviewed in several books [29, 49].

2.7.1 Biological Networks

The molecular interactions in a cell can be represented using graphs of network connections similar to the network of power lines. A set of connected molecular interactions can be considered as a pathway. The cellular system involves complex interactions between proteins, DNA, RNA, and smaller molecules and can be categorized in three broad subsystem: *metabolic network or pathway*, *protein network*, and *genetic or gene regulatory network*.

Metabolic network represents the enzymatic processes within a cell, which provide energy and building blocks for the cell. It is formed by the combination of a substrate with an enzyme in a biosynthesis or degradation reaction. Typically a mathematical representation of the network is a graph with vertices being all the compounds (substrates) and the edges linking two adjacent substrates. The catalytic activity of enzymes is regulated *in vivo* by

[2] Even though the hidden Markov model can be used for this purpose, its computational inefficiency prevents it from being applied to a large dataset.

multiple processes including allosteric interactions, extensive feedback loops, reversible covalent modifications, and reversible peptide-bond cleavage [29]. For well-studied organisms, especially microbes such as E. coli, considerable information about metabolic reactions has been accumulated through many years and organized into large online databases, such as EcoCyc [212].

Protein network is usually meant to describe communication and signaling networks where the basic reaction is between two proteins. These protein-protein interactions are involved in signal transduction cascade such as p53 signaling pathway. Proteins are functionally connected by post-translational, allosteric interactions, or other mechanisms into biochemical circuits [29].

Genetic network or regulatory network refers to the functional inference of direct causal gene interactions. According to the Central Dogma DNA $\rightarrow$ RNA $\rightarrow$ Protein $\rightarrow$ functions, gene expression is regulated at many molecular levels. Gene products interact at different levels. The analysis of large-scale gene expression can be conceptualized as a genetic feedback network. The ultimate goal of microarray analysis is the complete reverse engineering of the genetic network. The following discussion will focus on the genetic network modeling.

2.7.2 Modeling of Networks

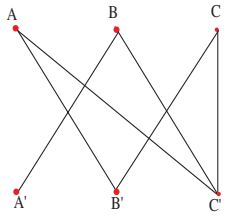
A systematic approach to modeling regulatory networks is essential to the understanding of their dynamics. Network modeling has been used extensively in social and economical fields for many years [377]. Recently several high-level models have been proposed for the regulatory network including Boolean networks, continuous systems of coupled differential equations, and probabilistic models. These models have been summarized by Baldi and Hartfield [29] as follows.

Boolean networks assume that a protein or a gene can be in one of two states: *active* or *inactive*, symbolically represented by 1 or 0. This binary state varies in time and depends on the state of the other genes and proteins in the network through a discrete equation:

$$X_i(t+1) = F_i[X_1(t), \ldots, X_N(t)], \tag{2.1}$$

where function F_i is a Boolean function for the update of the ith element as a function of the state of the network at time t [29]. Figure 2.3 gives a simple example. The challenge of finding a Boolean network description lies in inferring the information about network wiring and logical rules from the dynamic output (see Figure 2.3) [252].

Gene expression patterns contain much of the state information of the genetic network and can be measured experimentally. We are facing the challenge of inferring or reverse-engineering the internal structure of this genetic network from measurements of its output. Genes with similar temporal expression patterns may share common genetic control processes



(a) Network wiring

$$
\begin{aligned}
A' &= B \\
B' &= A\ OR\ C \\
C' &= (A\ AND\ B) \\
&OR\ (B\ AND\ C) \\
&OR\ (A\ AND\ C)
\end{aligned}
$$

(b) Logical rules

input			output		
A	B	C	A'	B'	C'
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	1	0	0
0	1	1	1	1	1
1	0	0	0	1	0
1	0	1	0	1	1

(c) Dynamic output

Fig. 2.3. Target Boolean network for reverse engineering: (a) The network wiring and (b) logical rules determine (c) the dynamic output.

and may therefore be related functionally. Clustering gene expression patterns according to a similarity or distance measure is the first step toward constructing a wiring diagram for a genetic network [378].

Continuous model/Differential equations can be an alternative model to the Boolean network. In this model, the state variables X are continuous and satisfy a system of differential equations of the form

$$\frac{dX_i}{dt} = F_i[X_1(t), \ldots, X_N(t), I(t)], \tag{2.2}$$

where the vector $I(t)$ represents some external input into the system. The variables X_i can be interpreted as representing concentrations of proteins or mRNAs. Such a model has been used to model biochemical reactions in the metabolic pathways and gene regulation. Most of the models do not consider spatial structure. Each element in the network is characterized by a single time-dependent concentration level. Many biological processes, however, rely heavily on spatial structure and compartmentalization. It is necessary to model the concentration in both space and time with a continuous formalism using partial differential equations [29].

Bayesian networks are provided by the theory of graphical models in statistics. The basic idea is to approximate a complex multidimensional probability distribution by a product of simpler local probability distributions. A Bayesian network model for a genetic network can be presented as a directed acyclic graph (DAG) with N nodes. The nodes may represent genes or proteins and the random variables X_i levels of activity. The parameters of the model are the local conditional distributions of each random variable given the random variables associated with the parent nodes,

$$P(X_1, \ldots, X_N) = \prod_i P(X_i | X_j : j \in N^{(i)}), \tag{2.3}$$

where $N^{(i)}$ denotes all the parents of vertex i. Given a data set D representing expression levels derived using DNA microarray experiments, it is possible to use learning techniques with heuristic approximation methods to infer

the network architecture and parameters. However, data from microarray experiments are still limited and insufficient to completely determine a single model, and hence people have developed heuristics for learning classes of models rather than single models, for instance, models for a set of coregulated genes [29].

2.8 Data Visualization and Visual Data Mining

The need for data visualization and visual data mining in the biomedical domain is motivated by several factors. First, it is motivated by the huge size, the great complexity and diversity of biological databases; for example, a complete genome of the yeast *Saccharomyces cerevisiae* is 12 million base pairs, of humans 3.2 billion base pairs. Second, the data-producing biotechnologies have been progressing rapidly and include spotted DNA microarrays, oligonucleotide microarrays, and serial analyses of gene expression (SAGE). Third, the demand for bioinformatics services has been dramatically increasing since the biggest scientific obstacles primarily lie in storage and analysis [181]. Finally, visualization tools are required by the necessary integration of multiple data resources and exploitation of biological knowledge to model complex biological systems. It is essential for users to visualize raw data (tables, images, point information, textual annotations, other metadata), preprocessed data (derived statistics, fused or overlaid sets), and heterogeneous, possibly distributed, resulting datasets (spatially and temporally varying data of many types).

According to [122], the types of visualization tools can be divided into (1) generic data visualization tools, (2) knowledge discovery in databases (KDD) and model visualization tools, and (3) interactive visualization environments for integrating data mining and visualization processes.

2.8.1 Data Visualization

In general, visualization utilizes the capabilities of the human visual system to aid data comprehension with the help of computer-generated representations. The number of generic visualization software products is quite large and includes AVS, IBM Visualization Data Explorer, SGI Explorer, Visage, Khoros, S-Plus, SPSS, MatLab, Mathematica, SciAn, NetMap, SAGE, SDM and MAPLE. Visualization tools are composed of (1) visualization techniques classified based on tasks, data structure, or display dimensions, (2) visual perception type, e.g., selection of graphical primitives, attributes, attribute resolution, the use of color in fusing primitives, and (3) display techniques, e.g., static or dynamic interactions; representing data as line, surface or volume geometries; showing symbolic data as pixels, icons, arrays or graphs [122]. The range of generic data visualization presentations spans line

graphs, scatter plots, 2D isosurfaces, 3D isosurfaces, rubber sheets, volume visualizations, parallel coordinates, dimensional stacking, ribbons with twists based on vorticity, streaklines using three time frames, combinations of slicing and isosurface, and scalar or vector or star glyphs. Most of these visualization forms are well suited for two-, three-, and four-dimensional data. However, special attention should be devoted to high-dimensional data visualization since biomedical information visualization quite often involves displaying heterogeneous multidimensional data. The list of high-dimensional table visualizations includes parallel coordinates, dimensional stacking (general logic diagrams or multiple nesting of dimensions using treemaps to display a 5D view of the DNA Exon/Intron data), multiple line graphs, scatter plot matrices (e.g., hyperslice and hyperbox), multiple bar charts, permutation matrices, survey "point-to-line" graphs, animations of scatter plots (the Grand Tour or the projection pursuit techniques), "point-to-curve" graphs (Andrew's curves), glyphs and icon-based visualization, the display of recursive correlation between dimensions (fractal foams), radial or grid or circular parallel coordinate visualizations (Radviz, Gridviz, overlapping star plots), and finally clustering visualization using dendrograms or Kohonen nets [122]. The most frequent high-dimensional biomedical data visualization is clustering visualization because of its direct use in studies searching for similarities and differences in biological materials. Nonetheless, one should also mention the applications of other sophisticated visualization systems, such as virtual reality environments for exploratory and training purposes, collaborative visualization systems for basic research (NCSA Biology Workbench), and telemedicine and telesurgery. In future, collaborative visualization systems would benefit from grid computing, scalable visualization capabilities, and integration with the tools providing qualitative views of a dataset [267].

2.8.2 KDD and Model Visualization

Visual data mining discovers implicit and useful knowledge from large data sets using data and/or knowledge visualization techniques [165]. It is the human visual and brain system that gives us the power of data model understanding and phenomenon comprehension based on visual data mining. While KDD and data mining experts focus on the KDD process and generate data models, researchers studying human computer interfaces, computer graphics, multimedia systems, pattern recognition, and high-performance computing work on effective visual data model visualizations. The benefits of data-mining model visualization are threefold [122]. First, anyone conducting the data analysis has to trust the developed model. In addition to good quantitative measures of "trust," visualization can reveal several model aspects to increase our trust. Second, good model visualization improves understanding of the model, especially semantic understanding.

Third, several data mining techniques lead to multiple data models, and it is natural to ask questions about model comparisons.

Comparing many data models requires establishing appropriate metrics, visualizing model differences, and interpreting the differences. Thus, appropriate model visualization is critical for interpreting data. In the biomedical domain, visual data mining delivers presentations of data mining models and helps interpret them in the biological domain. For example, visualization of decision trees, clusters, and generalized or association rules does not fulfill its purpose unless an expert biologist can connect the visual data model representation with the underlying biological phenomenon. Thus, many commercial software packages support model visualization tools, for instance, software by Spotfire, InforMax, or Affymetrics. Nevertheless, there is still a need to develop a metric to evaluate effectiveness of the variety of visualization tools and to permeate the KDD process with visualization to give useful insights about data. Figure 2.4 shows how microarray processing steps can be combined with visual data mining (inspection) of spot features and labels obtained by clustering.

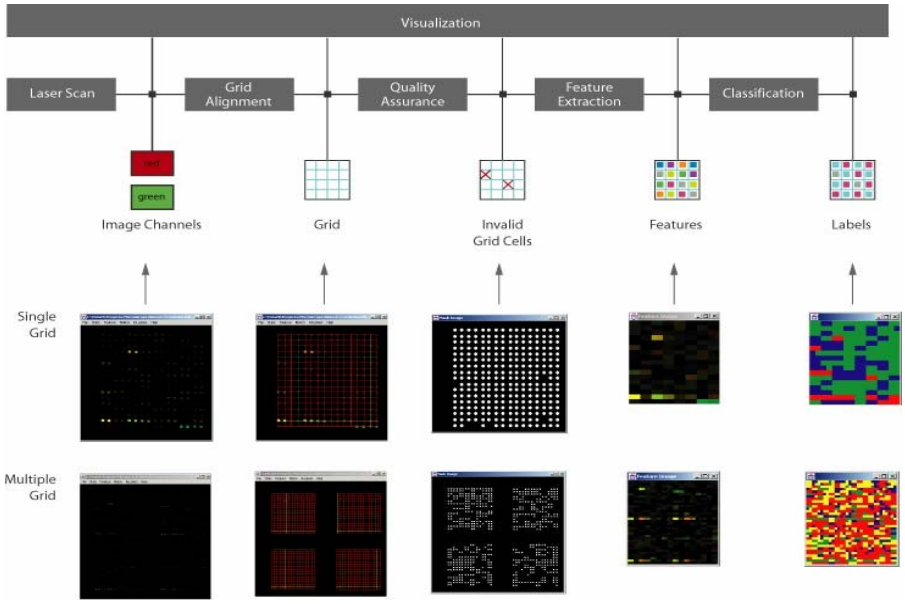


Fig. 2.4. Example of visualization combined with visual inspection of spotted DNA microarray data using I2K software developed at NCSA.

2.8.3 Integration of Data Mining and Visualization Processes

Having available all generic visualization tools and visualizations of data models, one would like to build an environment for visualization of the knowledge discovery in databases (KDD) process including exploratory data mining. In the KDD process, defined as the process of discovering useful knowledge within data [123], exploratory data mining is a class of methods used by the KDD process to find a pattern that is valid, novel, potentially useful, and ultimately understandable [122]. From a user viewpoint, the role of a user can be either passive, e.g., viewing data without any significant influence on the conducted data mining, or active, e.g., making decisions based on presented information visualization. In addition, the integration of data mining and visualization should be realized by various combinations of data mining and visualization components and characterized by seamless interface and repeatable execution at any workflow point. From a software design viewpoint, the integration environment has to be designed (a) with modular components performing individual workflow steps and (b) with common data objects so that the objects can be easily passed between processing and visualization components [122]. There are several software integration packages, e.g., D2K by NCSA and Iris Explorer by SGI, that meet these integration requirements by using a visual programming paradigm.

In the biomedical domain, integration challenges remain in developing software tools and environments that can be used for solving biological problems rather than general data mining problems. For example, there is a need for an integrated data workflow for (a) comparative studies visualizing comparisons of genes from different species, (b) multilevel studies visualizing phylogenetic trees at several levels of detail, (c) interactive studies visualizing polymer docking for drug design, and (d) mapping gene function in the embryo [267]. Building software environments of this kind requires not only bringing together data mining and visualization researchers but also unifying the domain-specific languages for common elements, e.g., defining terms for input and output data variables, intermediate data products, and user interfaces. This type of project has been demonstrated by Variagenics, Inc. and Small Design Firm in a nucleic acid sequence of the human genome consisting of 3.2 billion base pairs and displayed in a coherent three-dimensional space while preserving accurate spatial and size relationships [3]. The last but not the least important issue is related to visualization of the exponentially increasing volume of biological data that must utilize distributed computational resources and interoperability of all existing tools. This issue is being addressed by the development of (a) policies on data sharing and standards [51, 381], (b) computational grids, and (c) visualization techniques for large data sets [162].

2.9 Emerging Frontiers

There are many emerging technologies and research frontiers in bioinformatics. In this section, we present two emerging frontiers in bioinformatics research: text mining in bioinformatics and systems biology.

2.9.1 Text Mining in Bioinformatics

Bioinformatics and biodata analysis involve worldwide researchers and practitioners from many different fields: genetics, biochemistry, molecular biology, medical science, statistics, computer science, and so on. It becomes a challenging task to find all the related literature and publications studying the same genes and proteins from different aspects. This task is made even more demanding by the huge number of publications in electronic form that are accessible in medical literature databases on the Web.

The number of studies concerning automated mining of biochemical knowledge from digital repositories of scientific literature, such as MEDLINE and BIOSIS, has increased significantly. The techniques have progressed from simple recognition of terms to extraction of interaction relationships in complex sentences, and search objectives have broadened to a range of problems, such as improving homology search, identifying cellular location, and deriving genetic network technologies [179].

Natural language processing (NLP), also called computational linguistics or natural language understanding, attempts to process text and deduce its syntactic and semantic structure automatically. The two primary aspects of natural language are syntax and lexicon. Syntax defines structures such as the sentence made up of noun phrases and verb phrases. The smallest structural entities are words, and information about words is kept in a lexicon, which is a machine-readable dictionary that may contain a good deal of additional information about the properties of the words. Many techniques have been developed to construct lexicons and grammars automatically. For example, starting with a modest amount of manually parsed text, a parser can be "trained" by constructing rules that match the manually produced structures. This is a machine learning approach. Other kinds of analysis methods look for certain regularities in massive amounts of text. This is the statistical approach. NLP has become an important area over the last decade with the increasing availability of large, on-line corpora [23, 380].

The earliest work focused on tasks using only limited linguistic context and processing at the level of words, such as identifying protein names, or on tasks relying on word cooccurrence and pattern matching. The field now has progressed into the area of recognizing interactions between proteins and other molecules. There are two main methods in this area. The first approach is based on occurrence statistics of gene names from MEDLINE documents to predict the connections among genes [386]. The second approach uses

specific linguistic structures to extract protein interaction information from MEDLINE documents [105].

Besides the recognition of protein interactions from scientific text, NLP has been applied to a broad range of information extraction problems in biology. Combining with the Unified Medical Language System (UMLS), NLP has been used for learning ontology relations in medical databases and identifying the structure of noun phrases in MEDLINE texts. Incorporating literature similarity in each iteration of PSI-BLAST search has shown that supplementing sequence similarity with information from biomedical literature search could increase the accuracy of homology search results. Methods have also been developed (a) to cluster MEDLINE abstracts into "themes" based on a statistical treatment of terms and unsupervised machine learning, and (b) to classify terms derived from standard term-weighting techniques to predict the cellular location of proteins from description abstracts [179].

2.9.2 Systems Biology

System-level understanding, the approach advocated in systems biology, requires a shift in focus from understanding genes and proteins to understanding a system's structure and dynamics [191]. A system-level understanding of a biological system can be derived from an insight into four key properties, according to the prominent systems biologist Kitano [225]:

1. *System structures.* These include the network of gene interactions and biochemical pathways, as well as the mechanisms by which such interactions modulate the physical properties of intracellular and multicellular structures.
2. *System dynamics.* The principles about how a system behaves over time under various conditions can be understood through metabolic analysis, sensitivity analysis, dynamic analysis methods such as phase portrait and bifurcation analysis, and by identifying essential mechanisms underlying specific behaviors.
3. *The control method.* The mechanisms that systematically control the state of the cell can be modulated to minimize malfunctions and provide potential therapeutic targets for treatment of disease.
4. *The design method.* Strategies to modify and construct biological systems having desired properties can be devised based on definite design principles and simulations.

Computational biology has two distinct branches: (1) knowledge discovery, or data mining, which extracts the hidden patterns from huge quantities of experimental data, forming hypotheses as a result, and (2)

simulation-based analysis, which tests hypotheses with in silico experiments, providing predictions to be tested by in vitro and in vivo studies [224].

Although traditional bioinformatics has been used widely for genome analysis, simulation-based approaches have received little mainstream attention. Current experimental molecular biology is now producing the high-throughput quantitative data that is needed to support simulation-based research. At the same time, substantial advances in software and computational power have enabled the creation and analysis of reasonably realistic yet intricate biological models [224].

It is crucial for individual research groups to be able to exchange their models and create commonly accepted repositories and software environments that are available to all. Systems Biology Markup Language (SBML) [189], CellML (http://www.cellml.org/), and the Systems Biology Workbench are examples of efforts that aim to form a de facto standard and open software platform for modeling and analysis. These efforts significantly increase the value of the new generation of databases concerned with biological pathways, such as the Kyoto Encyclopedia of Genes and Genomes (KEGG), Alliance for Cellular Signaling (AfCS), and Signal Transduction Knowledge Environment (STKE), by enabling them to develop machine-executable models rather than merely human-readable forms [224].

Building a full-scale organism model or even a whole-cell or organ model is a challenging enterprise. Several groups, such as Virtual Cell [348] and E-Cell [405], have started the process. Multiple aspects of biological processes have to be integrated and the model predictions must be verified by biological and clinical data, which are at best sparse for this purpose. Integrating heterogeneous simulation models is a nontrivial research topic by itself, requiring integration of data of multiple scales, resolutions, and modalities.

2.9.3 Open Research Problems

The future of bioinformatics and data mining faces many open research problems in order to meet the requirements of high-throughput biodata analysis. One of the open problems is data quality maintenance related to (a) experimental noise, e.g., the hybridization process and microarray spot irregularities, and (b) the statistical significance of experiments, e.g., the number of experimental replicas and their variations. Other open problems include unknown model complexity and visualization difficulties with high-dimensional data related to our limited understanding of underlying phenomena. Although dimensionality reduction approaches reduce the number of data dimensions, they also introduce the problems of feature selection and feature construction. It has also become very clear over the last few years that the growing size of bioinformatics data poses new challenges on file standards, data storage, access, data mining, and information retrieval. These open research problems can be solved in future

by forming interdisciplinary teams, consolidating technical terms, introducing standards, and promoting interdisciplinary education.

How to integrate biological knowledge into the designing and developing of data mining models and algorithms is an important future research direction. There exists an extensive amount of information or knowledge about the biological data. For instance, the functionality of the majority of yeast genes is captured in the gene ontology (GO). The GO is a directed acyclic graph (DAG) that illustrates the relationship (similarity) among the genes. If we can combine this information into the data mining process, e.g., in clustering algorithms, then we can produce more biologically meaningful clusters with higher efficiency. Currently, integration of biological knowledge in the data mining procedure is still a challenging problem. It is desirable to find a way to represent prior biological knowledge as a model that can be integrated into the data mining process.

Recently, many researchers have realized that although a good number of genes have been discovered and have been playing an important role in the analysis of genetic and proteomic behaviors of biological bodies, the discovered genes are only about 1% to 2% of human (or animal) genome; most of the genome belongs to so-called "dark" matter, such as introns and "junk." However, recent studies have shown that a lot of biological functions are strongly influenced or correlated with the dark part of the genome, and it is a big open problem to find the rules or regularities that may disclose the mystery of the "dark matter" of a genome. This should be an interesting research problem that data mining may contribute to as well.

2.10 Conclusions

Both data mining and bioinformatics are fast-expanding and closely related research frontiers. It is important to examine the important research issues in bioinformatics and develop new data mining methods for scalable and effective biodata analysis.

In this chapter, we have provided a short overview of biodata analysis from a data mining perspective. Although a comprehensive survey of all kinds of data mining methods and their potential or effectiveness in biodata analysis is well beyond the task of this short survey, the selective examples presented here may give readers an impression that a lot of interesting work has been done in this joint venture. We believe that active interactions and collaborations between these two fields have just started. It is a highly demanding and promising direction, and a lot of exciting results will appear in the near future.

Acknowledgments

The work was supported in part by National Science Foundation under Grants IIS-02-09199 and IIS-03-08215, National Institutes of Health under Grants No. 2 P30 AR41940-10 and PHS 2 R01 EY10457-09, the University of Illinois at Urbana-Champaign, the National Center for Supercomputing Applications (NCSA), and an IBM Faculty Award. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding agencies.