The Knowledge Engineering Review, Vol. 14:3, 1999, 257–277
Printed in the United Kingdom

Knowledge discovery and data mining in biological databases

VLADIMIR BRUSIC[1] and JOHN ZELEZNIKOW[2]

[1]*Kent Ridge Digital Labs, 21 Heng Mui Keng Terrace, Singapore 119613. Email: vladimir@krdl.org.sg*
[2]*School of Computer Science and Computer Engineering, La Trobe University, Bundoora, Victoria, Australia. Email: johnz@latcs1.cs.latrobe.edu.au*

Abstract

The new technologies for Knowledge Discovery from Databases (KDD) and data mining promise to bring new insights into a voluminous growing amount of biological data. KDD technology is complementary to laboratory experimentation and helps speed up biological research. This article contains an introduction to KDD, a review of data mining tools, and their biological applications. We discuss the domain concepts related to biological data and databases, as well as current KDD and data mining developments in biology.

1 Introduction

Biological databases continue to grow rapidly. This growth is reflected by increases in both the size and complexity of individual databases as well as in the proliferation of new databases. A huge body of data is thus available for the extraction of high-level information, including the development of new concepts, concept interrelationships and interesting patterns hidden in the databases.

KDD is an emerging field combining techniques from databases, statistics and artificial intelligence, which is concerned with the theoretical and practical issues of extracting high level information (or knowledge) from a large volume of low-level data. Examples of high-level information derived from low-level data include forms that are more compact (e.g., short reports), more abstract (e.g., descriptive models of the process that generated data), or more useful (e.g., predictive models for estimating values of the future cases) than the low-level data. According to Fayyad et al. (1996), KDD refers to the overall process of discovering useful knowledge from databases, and data mining refers to a particular step in this process. They defined knowledge discovery in databases as *the non-trivial process of identifying valid, novel, potentially useful, and understandable patterns in data*. Data is a set of facts (stored in a file or a database) and a pattern is an expression in some language describing a subset of the data or a model applicable to the subset. Extracting a pattern also involves: (a) fitting a model to data, (b) finding structure from data, or (c) making any high-level description of a set of data. The KDD process is interactive and iterative (Brachman and Anand, 1996). KDD comprises multiple steps, which involve: (a) data preparation, (b) pattern searching, (c) knowledge evaluation, and (d) refinement. These steps can be repeated in multiple iterations. The core step of KDD is data mining – the application of specific tools for pattern discovery and extraction. The KDD process uses search or inference methods, rather than simple calculations.

The usage of standard algorithms such as BLAST (Altschul and Gish, 1996) or FASTA (Pearson, 1998) in comparing a given biological sequence with database entries does not equate to performing knowledge discovery, although these algorithms may be used in particular steps of the KDD process. The practical aspects of data mining include dealing with issues such as data storage and access, scalability of massive data sets, presentation of results and human-machine interaction.

The discovered patterns should be valid, in that the user should have a high degree of certainty of a correct result when the derived knowledge is extracted from new data. Various measures of validity are available, including prediction accuracy on new data and the utility or gain (for example in dollar value or speed-up). The estimation of novelty, usefulness, or understandability of the discovered knowledge is much more subjective and depends on the purpose of the KDD. Interestingness (Silberschatz and Tuzhilin, 1997) represents an overall measure of a pattern value, which combines validity, novelty, usefulness and simplicity. Biological data is inherently noisy, containing errors and biases. Filtering errors and de-biasing data help improve the results of KDD. This filtering can be performed at every step of the KDD process and is often based on human decision. Alternatively, the filtering can be internal to the data mining algorithm. The validation of the discovered knowledge is a critical issue for the data mining as well as the overall KDD process. Verification tasks themselves are a form of validation and involve estimation of the quality of data fitting and hence require the use of statistical tests. Discovery, in particular prediction tasks, requires careful validation.

Both the general requirements of the KDD process and the specific requirements of the application domain need to be considered in the design of a KDD process in biology. Data mining tasks have been defined in study of biological sequences. Examples including the finding of genes in DNA sequences (e.g., Krogh et al., 1994), regulatory elements in genomes (e.g., Brazma et al., 1997) and knowledge discovery on both transmembrane domain and signal peptide sequences (Shoudai et al., 1995). Numerous tools suitable for data mining in biology are available, yet the selection of an appropriate tool is non-trivial. The KDD process provides for the selection of the appropriate data mining methods by taking into account both domain characteristics and general KDD process requirements.

The KDD methodology is complementary to laboratory experiments and can accelerate the process of discovery in biology. This is achieved by both minimisation of the number of necessary experiments and by an improved capacity to interpret biological data. Besides prediction of biological function, the examples of successful applications include KDD for experiment planning (Honeyman et al., 1998a), and for a description of new biological concepts (Brusic et al., 1998b).

2 Introduction to KDD

2.1 Definitions

2.1.1 What is knowledge?

Data is raw material which needs to be processed – by a human, or a computer, or indeed by any other means. Information is data that has been organised (by a human or a computer) so that it is meaningful and useful. Conventional databases represent simple data types, such as numbers, strings and Boolean values. Knowledge is a form of information, besides raw data, interpreted data and expertise. Current applications require more complex information such as processes, procedures, actions, causality, time, motivations, goals, and common sense reasoning (Firebaugh, 1989: Ch. 9). Biological applications also require information on structure and organisation. The term *knowledge* describes this broader category of information.

2.1.2 What is knowledge discovery from databases?

At an abstract level, the Knowledge Discovery from Databases (KDD) field is concerned with the development of methods and techniques for making sense from data. KDD is useful where low-level data is difficult to understand or interpret because it is either too voluminous or too complex. If data are derived from a particularly complex domain the KDD process is typically performed on small data sets, relative to the complexity of the process that generated the data. At the core of the KDD process is the application of specific data mining methods for pattern discovery and extraction.

2.1.3 Why do we need KDD?

Fayyad et al. (1996) state that the traditional method of turning data into knowledge relies on manual analysis and interpretation. This approach is known in deductive databases, where the rules would be learned manually from interviewing experts (Zeleznikow and Hunter, 1994: Ch. 8). The classical approach to data analysis relies fundamentally on one or more analysts becoming intimately familiar with the data and serving as an interface between the data and the users and products. Manual probing of a data set is slow, expensive and highly subjective. With data volumes growing dramatically, manual data analysis is becoming impractical. Databases are increasing in size in two ways: the number N of records or objects in the database, and the number d of fields or attributes to an object. In the domain of Astronomy, databases containing the order of $N = 10^9$ objects are becoming common. In medical diagnostic applications there are databases containing even $d = 10^3$ fields. Biological databases are even more complicated, since related data are dispersed across heterogeneous and geographically scattered databases. In a database containing millions of records, with tens or hundreds of fields, some form of automated analysis is essential.

2.1.4 KDD process

The KDD process involves ten steps, the first nine were defined by Fayyad et al. (1996), and the last step by Zeleznikov and Stranieri (personal communication).

1. Learning the application domain – this includes developing relevant prior knowledge and identifying the goal and the initial purpose of the KDD process from the user's viewpoint.
2. Creating a target data set – including selecting a data set or focusing on a set of variables or data samples on which the discovery is to be performed.
3. Data cleaning and pre-processing – includes operations such as removing noise or outliers if appropriate, collecting the necessary information to model or account for noise and deciding on strategies for handling missing data fields.
4. Data reduction and projection – includes finding useful features to represent the data. With dimensionality reduction or transformation methods, the effective number of variables under consideration can be reduced, or invariant representations for the data can be found.
5. Choosing the function of data mining – includes deciding the purpose of the model derived by the data mining algorithm: summarisation, classification, regression and clustering.
6. Choosing the data mining algorithms – includes selecting methods to be used for searching for patterns in the data and matching a particular data mining method with the overall criteria of the KDD process. This process includes deciding which models and parameters might be appropriate. It also involves matching a particular data mining method with the overall criteria of the KDD.
7. Data mining – includes searching for patterns of interest in a particular representational form or a set of such representations including classification rules or trees, regression, clustering and dependency modeling.
8. Interpretation – involves possible further iterations of any of steps (1–7). This step can also involve visualisation of the extracted patterns and models or visualisation of the data given the extracted models;
9. Using discovered knowledge – this step involves acting directly on the discovered knowledge, incorporating the knowledge into another system for further action, or documenting and reporting the knowledge. It also includes checking and resolving potential conflicts with previously believed (or extracted) knowledge.
10. Evaluation of KDD purpose – newly discovered knowledge is often used to formulate new hypotheses; also new questions may be raised using the enlarged knowledge base. In this step the KDD process is evaluated for possible further use in both refinement and expansion of the purpose of the KDD process relative to the previous KDD cycle. The diagrammatic representation of the KDD process is given in Figure 1.

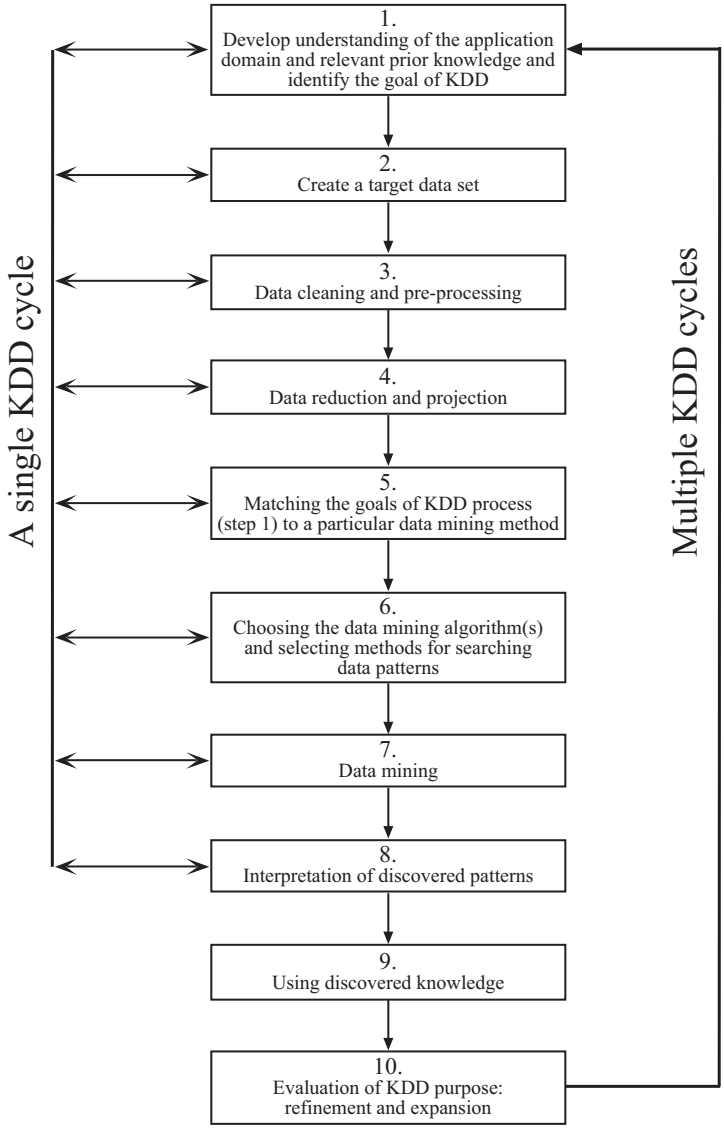

Figure 1 Steps of the KDD process

2.1.5 Data mining

Data mining is a problem-solving methodology that finds a formal description, eventually of a complex nature, of patterns and regularities in a set of data. Decker and Focardi (1995) consider various domains that are suitable for data mining, including medicine and business. They state that in practical applications, data mining is based on two assumptions. First, the functions that one wants to generalise can be approximated through some relatively simple computational model with a certain level of precision. Second, the sample data set contains sufficient information required for performing the generalisation. Fayyad et al. (1996) see data mining as the application of specific algorithms for extracting patterns from data. The additional steps in the KDD process are essential to ensure that useful knowledge is derived from the data. Blind application of data mining – known as *data dredging* – can easily lead to the discovery of meaningless or misleading patterns.

2.2 KDD – an interdisciplinary topic

KDD brings together distinct research fields including machine learning, pattern recognition, databases, statistics, artificial intelligence, knowledge acquisition, data visualisation and high-

performance computing. The common goal is extracting new high-level knowledge from data. The data mining component of KDD uses tools from statistics, machine learning, and pattern recognition to extract patterns from data. KDD focuses on the overall process of knowledge discovery from data, including (a) data issues (storage and access), (b) data set scaling and reduction, (c) visualisation of results, (d) man-machine interactions, (e) pattern recognition, (f) modelling algorithms, and (g) interpretation of results.

KDD data is a statistical endeavor. Statistics provides a language and framework for quantifying the uncertainty that results from inference of general patterns from a sample data. A specific concern that requires a careful consideration is that patterns, which appear to be statistically significant but in fact are not, can be found in any data set (even in randomly generated data). Data mining is legitimate if it is performed with the appropriate consideration of the statistical aspects of the studied problem. KDD provides tools to combine and automate, as much as possible, the process of data analysis and the art of hypothesis selection.

Data warehousing is the process of collecting, cleaning and reducing of transactional data for on-line analysis and decision support. Data warehousing facilitates two aspects of KDD process: data cleaning and data access. Data cleaning addresses data naming convention, uniform data representation, handling missing data, filtering noise and errors, and data de-biasing. Data access issues include defining uniform methods for accessing the data (including the data stored off-line).

After data is stored and accessible, KDD can be performed. A popular approach for analysis of data warehouses is called On-Line Analytical Processing (OLAP) (Codd, 1993). OLAP tools focus on providing multidimensional data analysis, which is superior to SQL in computing summaries and breakdowns along many dimensions. OLAP tools are targeted toward simplifying and supporting interactive data analysis, whilst the goal of KDD tools is to automate as much of the process as is possible. Thus, KDD is a step beyond what is currently supported by most standard database systems.

2.3 The data mining step of the KDD process

The data mining component of the KDD process often involves an iterative application of particular data mining methods. Data mining involves fitting models to observed data or producing various forms of data descriptions. The fitted models may represent the inferred knowledge. Human judgement is often required in deciding if models reflect useful or interesting knowledge. Two mathematical formalisms are used in model fitting: statistics and logic. A non-deterministic underlying model is assumed in statistical approach, whereas the logical model is purely deterministic. The statistical approach to data mining is most widely used for practical data mining applications because real-world data is commonly associated with uncertainty. Most data mining methods are based on well-developed techniques from machine learning, pattern recognition, and statistics (such as classification, clustering or regression).

In this section we describe two practical goals of data mining: prediction and description. These goals can be achieved by using various general data mining methods, which are described below. A more detailed explanation can be found in Fayyad et al. (1996)

Description focuses on finding interpretable patterns, which either quantify the existing data or capture the essential qualities within the data. Predictive data mining refers to assigning a value to a variable of interest in the context of a new or future case. Although the boundaries between prediction and description are not sharp, the distinction is useful for understanding the overall knowledge discovery goal. The relative importance of prediction and description on particular data mining applications can vary considerably. The goals of prediction and description can be achieved using a variety of particular data mining methods. These methods include classification, regression, clustering, summarisation, dependency modeling, and change and deviation detection (Fayyad et al., 1996).

Classification is a learning technique that finds a function, which maps a data item into one of several predefined classes (e.g., the prediction of peptides that bind MHC molecules – Brusic et al.,

1998a). Regression maps a data item to a real-valued variable (e.g., quantitative structure-activity relationship analysis – Kubinyi et al., 1998). Clustering is used to identify distinct or overlapping subsets within data, which provide better description (e.g., finding clusters of protein families – Tatusov et al., 1997). Summarisation comprises methods for finding a compact description for data. Dependency modelling refers to finding a model or a description that explains significant dependencies between variables (e.g., modelling the human genome). Change and deviation detection focuses on discovering the most significant changes in the data from previously measured or normative values (e.g., study of mutagenicity of active compounds – King et al., 1996).

Any data mining algorithm comprises three primary components (Fayyad et al., 1996), namely: (a) model representation, (b) model evaluation, and (c) search. Model representation is the language for describing the data patterns. The derivation of a representative model requires that the model representation must provide sufficient complexity, and that a sufficient amount of data is available. Model evaluation comprises activities for assessing the adequacy of the model, including measures of accuracy and interestingness. Search methods include model search and parameter search. The goal of a model search is finding the most adequate model representation for a given problem. Parameter search is an optimisation process for finding model parameters that produce the best data fitting.

2.4 Data mining tools: an overview

We shall briefly describe several popular techniques, namely: (a) decision trees and rules, (b) non-linear regression and classification methods, (c) example-based methods, (d) probabilistic models, and (e) relational learning models. In this paper, we provide an overview of data mining methods intended to help the reader understand the data mining methods and facilitate the selection of the "most appropriate method" for a given problem.

2.4.1 Decision trees and rules

Decision trees consist of nodes and edges; each node contains a test on some attribute of the data. Decision trees and rules that use binary splits produce classifications which can be easily understood, and produce compact models. However, the restriction to a particular tree or rule representation can limit the functionality and approximation power of the model. An example of a decision tree is given in Figure 2. Decision trees use likelihood-based model-evaluation methods,

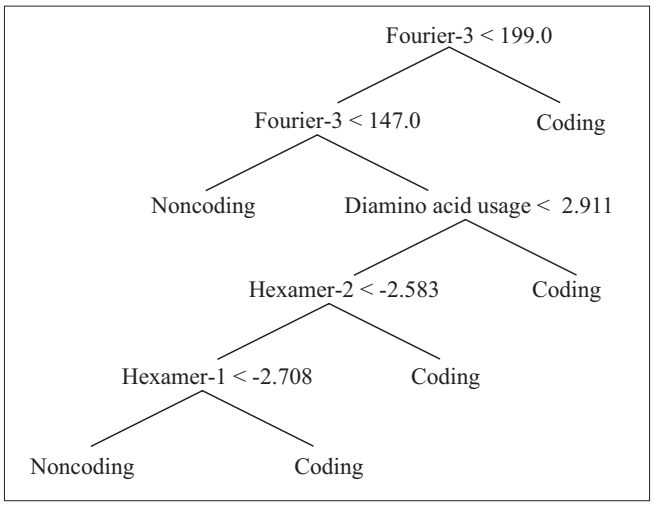

Figure 2 An example of a decision tree which predicts protein coding regions. This decision tree uses four features and contains five test nodes. (Adapted from Salzberg (1995) where detailed description of feature measures can be found.)

combined with search methods for growing and pruning tree structures. Decision trees and rules are commonly used in prediction tasks for classification, regression or summarisation.

Induction is the process in which rules are generated from sample cases. A rule induction system creates rules that fit the example cases. The rules can be used to assess other cases where the outcome is not known. An example of a rule induction system is the ID3 algorithm of Quinlan (1986), which has been extended to the C4.5 algorithm (Quinlan, 1993), and more recently to C5.0. A characteristic of induction algorithms is that the learning is based on the statistical analysis of the training set. Machine induction allows for the deduction of new knowledge. It may be possible to list all the factors influencing a decision without understanding their impacts. The rules generated can be reviewed and modified by the domain expert. Difficulties in implementing rule induction systems include:

- The generated rules are not always easy for humans to understand.
- If the attributes selected by the domain expert for defining the training set are not appropriate, it is likely that the induced rules will be of little value.
- Rule induction systems work well only with relatively small number of attributes.
- The training set should not include cases that are exceptions to the underlying rules. In biology, this requirement is difficult to fulfil.

An example of a tree-based application in biology is BONSAI Garden System (Shoudai et al., 1995). The BONSAI system uses positive and negative examples to produce decision trees and has been used to discover knowledge on transmembrane domain sequences and signal peptide sequences through the use of computer experiments. Decision trees have also been used for determination of protein coding regions (Salzberg, 1995).

2.4.2 Nonlinear regression and classification methods

Non-linear regression methods utilise non-linear functions such as polynomials, sigmoids, or splines, for finding relationships between input variables X_i and output variables Y_i, by fitting functions to the available data. Examples include methods which use (Fayyad et al., 1996): (a) feed-forward neural networks, (b) adaptive splines, or (c) projection pursuit. Non-linear regression methods, although powerful in representational power, can be difficult to interpret.

A neural network of the appropriate size can universally approximate any smooth function to any desired degree of accuracy. However, it is relatively difficult to elucidate generalised rules that characterise training data from a trained neural network. Artificial neural networks were originally designed to simulate the information processing (connectivity and signalling) within a biological brain. This consists of many self-adjusting processing elements cooperating in a densely inter-connected network. A description of neural network theory with applications in biology can be found in Baldi and Brunak (1998: Ch. 5, 6). There are many examples of neural network applications in biology starting from early 1980s. An early example is the prediction of translation initiation sites in DNA sequences (e.g., see Stormo et al., 1982). Brusic et al. (1998a) developed the PERUN system which utilises an evolutionary algorithm and artificial neural networks for prediction of immunologically interesting peptides.

Adaptive spline functions provide smooth approximations of multidimensional objects, which have the ability to capture high order interactions. This method is exemplified in MARS (Multiple Adaptive Regression Splines) model (Friedman, 1991). The MARS model utilises recursive partitioning of the input space in search of smooth basis functions that approximate multi-dimensional objects. The model is built by fitting of the splines to the overlapping input space partitions, followed by pruning using cross-validation. A particular strength of the MARS model is in its interpretation capabilities: the effects of individual variables and pairs of variables are collected together and presented graphically. This method is relatively complex for use (Elder and Pregibon, 1996), and has not been extensively used in biology. Clinical applications have been reported in Friedman and Roosen (1995).

Projection pursuit methods are useful for finding general low-dimensional structure in high-

dimensional, sparse data (Cook et al., 1995). Projection pursuit is a set of analytical techniques for finding interesting projections of multi-variate data. Early applications included visual examination of data represented as various types of plots (histograms, scatterplots or three-dimensional plots). In high dimensional models the number of views can be large; statistical measures of interestingness have been defined to help the selection of interesting views (see also Silberschatz and Tuzhilin, 1997). These measures include the deviation from normal distribution (Diaconis and Freedman, 1984) and maximal correlation index (Friedman and Stuetzle, 1981). A singular measure of interestingness is simple to use; however, structures, that are obvious to an analyst using visual inspection, are often misclassified (Elder and Pregibon, 1996). Projection pursuit was used for classification of protein structures (Klein and Somorjai, 1988).

2.4.3 Example-based methods

Example-based methods use representative examples to approximate a model. The properties of new examples are predicted by matching properties of well-known examples in the model. These methods include: (a) nearest neighbour classification, (b) regression analysis, and (c) case-based reasoning. Example-based methods have proven very useful in biology, either when dealing with sparse data or when combined with other methods.

Nearest neighbour classification (Cover and Hart, 1967) is a non-parametric (model-free) method, which examines distances between the input case and known points, using pre-defined metrics. The result returned is the closest point. Alternatively, a case will be classified according to its similarity to previously known cases. Biological sequence similarity search methods, such as BLAST or Fasta are forms of nearest neighbour methods. The metrics for individual searches are defined by selection of the comparison algorithm, of search parameters (such as gap and gap-length penalty), and of comparison matrices. The advantage of nearest neighbour methods is that they are simple to develop and easy to use. However, the accuracy of these methods is highly dependent on the selection of the distance metrics, and is often low for problems of high dimensionality. The applications of nearest neighbour methods in biology include the prediction of protein secondary structure (Levin, 1997) and the analysis of evolutionary trees (Li et al., 1996).

Regression analysis can be the goal of a data mining exercise. However, regression analysis can also be used as a data mining tool. Various forms of regression analysis include linear models (McCullagh and Nielder, 1989) or non-linear models (Bates and Watts, 1988). Regression methods have been applied to a variety of biological problems including quantitative structure-function analysis (Kubinyi et al., 1998), secondary structure content (Zhang et al., 1998) and protein/ligand interactions (Kauvar et al., 1995).

Case-Based Reasoning (CBR) is the common name for a number of techniques that use representation and reasoning from prior experience to analyse or solve a new problem. CBR may include explanations of similarities or differences between the previous examples and the present problem. It also includes techniques for adaptation of past solutions to meet the requirements of the present problem. Figure 3 indicates the case-based reasoning cycle. The characteristics of case based reasoners are:

- They can arrive at conclusions based on a number of cases, rather than the entire set of possibly contradictory and complex rules.
- They can interpret open textured concepts by using analogy.
- In sharp contrast to rule-based systems, the accuracy of a CBR system increases with the number of stored cases.
- Case based reasoners can improve the knowledge acquisition process because the notion of a case, precedent or prior experience is intuitive for knowledge engineers and domain experts alike.

Ashley (1992) has identified five case-based reasoning approaches: (a) statistically oriented, (b) model based, (c) planning/design oriented, (d) exemplar based, or (e) precedent based. Examples from biology include the use of case-based reasoning in prediction of protein secondary structure (Leng et al., 1994) and gene annotation applications (Overton and Haas, 1998).

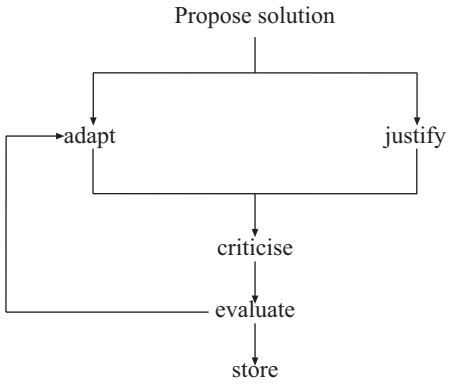

Figure 3 Case-based reasoning cycle. (Adapted from Kolodner (1993).)

2.4.4 Probabilistic models

Bayesian methods provide a formalism for reasoning about partial beliefs under conditions of uncertainty. In this formalism, propositions are given numerical values, signifying the degree of belief accorded to them. Bayes' theorem is an important result in probability theory, which deals with conditional probability. It is useful in dealing with uncertainty. Bayesian inference networks have proved very significant in the domain of information retrieval. Bayes theorem states that the probability of cause A_i given the observation of event J is equal to the joint probability of J and A_i divided by the sum of the joint probabilities of J with all terms of A_is.

$$\Pr(A_i|J) = \frac{\Pr(J|A_i)\Pr(A_i)}{\sum_{k=1}^{k=n} \Pr(J|A_k)}$$

The representative probabilistic techniques include (a) Bayesian classification, (b) probabilistic graphic dependency models, and (c) hidden Markov models.

Bayesian classification can be considered as either discovering the classes and their descriptions from a set of cases (unsupervised classification) or mapping a new case to a set of pre-defined classes (supervised classification). Bayesian prediction can be used to determine sets of attributes that define inter-class differences. The introduction of prior probabilities helps improve the integration between the data fit and the generalisation power of the model. The limitation of this approach is that the underlying conceptual model must be explicitly defined in terms of attributes and prior probabilities. Bayesian classification is described by Cheeseman and Stutz (1996). Examples of such biological applications include the estimation of evolutionary dates from sequence data (Thorne et al., 1998); classification of protein sequence families (Qu et al., 1998); determination of evolutionary distances in aligned sequences (Agarwal and States, 1996); and finding regulatory regions in DNA (Crowley et al., 1997).

Graphical models specify probabilistic dependencies using a graph structure. The model specifies the dependencies between variables, which can be categorical, discrete-valued or real-valued. Early graphic models were developed for probabilistic expert systems. The model structure and the parameters were elicited from experts. In graphical models, the model evaluation uses Bayesian probabilities, with a variety of estimation techniques or iterative search methods for parameter estimation. Various heuristics containing prior knowledge can be used to reduce the search space. Probabilistic graphical models are of interest to KDD because the graphical representation of the model facilitates human interpretation. Probabilistic graphical models are described by Whittaker (1990).

A Hidden Markov model (HMM) is a class of probabilistic graphical models. It is defined by a finite set of states, associated with a (usually multidimensional) probability distribution. Transitions between the states are governed by a set of transition and emission probabilities. An outcome of a transition from a particular state can be generated, according to the associated probability

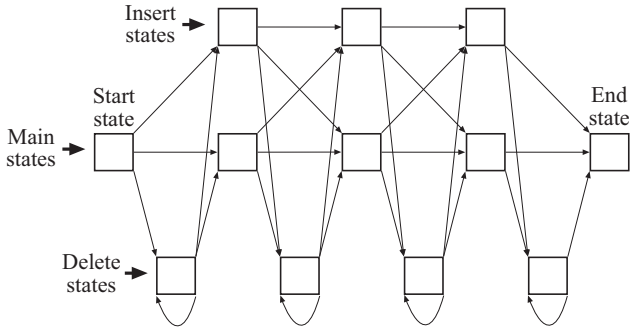

Figure 4 An architecture of HMM. (Adapted from Baldi and Brunak (1998).)

distribution. The states are not visible to an external observer, and therefore they are "hidden" to the outside; only the outcome is visible. The assumption in first-order Markov model is that the transitions depend only upon the current state. An example of the architecture of a HMM is given in Figure 4. HMMs can be trained using sets of pre-classified examples and a variety of learning algorithms. The advantages of HMMs are that they combine solid statistical basis with efficient learning algorithms. The limitations of HMMs include the need for a large number of free parameters, which in turn require a significant number of training cases. Further, a good knowledge of the domain model is required for selecting the appropriate HMM architecture for a specific task. HMMs have been extensively used in modelling biological sequence data. A detailed description of HMMs, with biological applications, can be found in Baldi and Brunak (1998: Ch. 7, 8).

2.4.5 Relational learning models

Relational learning or Inductive Logic Programming (ILP) combines the language of first order predicate calculus with machine learning and automatic programming. A relational learner can easily derive formulae such as X = Y from within data. Relational models have strong representational power, but this comes at the price of significant search costs. The detailed description of ILP can be found in Dzeroski (1996). The applications of ILP in biology include discovery structure-function relationship for drug design (King et al., 1996).

2.5 An evaluation of performances of various predictive data mining models

When representing data, we must ensure that all relevant features needed for prediction are encoded; each case will require some minimal representation. On the other hand, if the case template for the data is larger than need be, we may introduce unnecessary complexity that can adversely affect performance of the prediction system. The most commonly used performance measure of prediction system is the error rate. True error rate may be different from the computable apparent error rate and depends upon factors including the number, quality and statistical distribution of available data and estimation techniques (Weiss and Kulikowski, 1991).

Several measures are available for estimation of the accuracy of a model. The definition of terms is given in Table 1. Common measures include sensitivity (SE) and specificity (SP). SE indicates the quantity of predictions, i.e., the proportion of correctly predicted true positives. SP indicates the quality of predictions, namely the proportion of correctly predicted true negative examples. Sensitivity and specificity measures must always be used as paired measures. If a predictive model achieves SE = 100%, this could be because of the model's high accuracy, in which case SP will also be high. Alternatively, high sensitivity and low specificity indicate poor selection of the decision threshold. Selection of the decision threshold, i.e., the score used to discriminate between positive and negative examples, will influence the values of SE and SP. Lowering the decision threshold will increase SE and decrease SP of predictions.

Acc and Aroc (Table 1) provide the convenience of a single measure of the accuracy of predictive

Table 1 Definition of terms for assessing the accuracy of predictive models

	Experimental positives	Experimental negatives
Predicted positives	True positives (TP)	False positives (FP)
Predicted negatives	False negatives (FN)	True negatives (TN)

Accuracy measure	Formula	Pairs with
Sensitivity	SE = TP/(TP + FN)	SP
Specificity	SP = TN/(TN + FP)	SE
Positive predictive value	PPV = TP/(TP + FP)	NPV
Negative predictive value	NPV = TN/(TN + FN)	PPV
Accuracy	Acc = (TP + TN)/(TP + TN + FP + FN)	–
Aroc	Integration of ROC curves (see Swets, 1988)	–

models. The Acc measure is suitable when prevalence of positive and negative cases is similar and therefore is often not useful in prediction of biological effects. Using Relative Operation Characteristics (ROC) (Swets, 1988) for the integration of functions of (1-SP, SE) for various decision thresholds provides the Aroc measure. Values of Aroc = 50% indicate random-choice, Aroc > 80% good accuracy, and Aroc > 90% excellent accuracy of predictions (Swets, 1988). A variety of theoretical methods exist including splitting data into training and test sets, internal cross-validation and bootstrapping (described by Weiss and Kulikowski, 1991). Theoretical estimates of accuracy tend to be somewhat optimistic. Experimental testing of theoretical models is the best validation option, provided that the experimental method is of an acceptable accuracy.

2.6 Comparative notes on data mining methods

Logic and rule based systems are easy to build and development shells are available which can speed the process of building commercial decision support systems. However, such systems are limited in reasoning ability – they require interactive input from human experts. *We advocate the use of combined systems, which can perform analogical, inductive and deductive reasoning*. The logic of exploratory data analysis has been studied extensively – for an initial reference see Yu (1994).

A disadvantage of example-based methods, as compared with tree-based, is that when using the former, a well-defined distance metric for evaluating the distance between data points is required. Model evaluation is typically based on cross-validation estimates of a prediction error (Weiss and Kulikowski, 1991). The parameters of the model to be estimated can include the number of neighbours which are required to make a prediction as well as the distance metric itself. Case-based reasoners can be built much more quickly than rule-based reasoners and are much easier to maintain. The addition of a new rule to a rule-based system can require the modification of several other rules. The addition of cases to a case library rarely involves modification of the library. Model-based reasoning is based on knowledge of the structure and behaviour of the devices the system is designed to understand.

Non-linear regression methods are relatively easy to build and maintain, and can tolerate noisy data, however they require relatively large data sets, and it is often difficult to extract relevant rules from the model.

Example-based methods are often powerful in their approximation ability, but conversely, can be difficult to interpret because the model is implicit in the data and not explicitly formulated. This occurs in the case of neural networks. Case-based reasoning, on the other hand, offers the following natural techniques for realising expert systems goals: (a) compiling past solutions, (b) avoiding past

mistakes, (c) interpreting rules, (d) supplementing weak domain models, (e) facilitating explanation, and (f) supporting knowledge acquisition and learning.

Human knowledge acquisition often involves the use of experiences and cases; case-based reasoning often accurately models human reasoning. Compared to both rule-based systems and non-linear regression methods, case-based reasoners have disadvantages in that they are hard to build, complicated to maintain, and more likely to be research prototypes than commercially useful systems.

The advantage of probabilistic methods is that they utilise the well-defined theoretical background of Bayesian concepts. The disadvantage of probabilistic methods is that they require correctly assigned probabilities, which often cannot be clearly assigned in particular biological cases. This requires good understanding of the nature of data, which is not a requirement in non-linear regression methods.

Each data mining technique typically has a set of problems for which it is best suited. For example, decision tree classifiers can be useful for finding structure in high-dimensional spaces and in problems with mixed continuous and categorical data, because tree distances do not require distance metrics. However, classification trees might not be suitable for problems where the true decision boundaries between classes are described by a polynomial. *There is no universal data mining method and choosing a particular algorithm for a particular application is an art rather than a science. In practice, a large portion of the application effort should go into properly formulating the problem rather than into optimising the algorithmic details of a particular data mining method.*

3 Domain concepts from biological data and databases

3.1 Bioinformatics

There are ever-increasing requirements for both the speed and the sophistication of data analysis. Bioinformatics is a field emerging at the overlap between biology and computer science. Biological science provides deep understanding of this complex domain, while computer science provides an effective means to store and analyse large volumes of complex data. Combining the two fields provides the potential for great strides in understanding biological systems and increasing the effectiveness of biological research. There are many problems in ensuring the effective use of bioinformatic tools: an average biologist has a limited understanding of sophisticated data analysis methods and of their applicability and limitations; an average computer scientist lacks understanding of the depth and complexity of biological data. Bioinformaticians need to develop the understanding of both fields.

The KDD process provides a framework for the efficient use of bioinformatics resources in both defining meaningful biological questions and obtaining acceptable answers.

3.2 What do we need to know about biological data?

The four most important data-related considerations for the analysis of biological systems are understanding of: (a) the complexity and hierarchical nature of processes that generate biological data, (b) the fuzziness of biological data, (c) the biases and potential misconceptions arising from domain history, reasoning with limited knowledge, a changing domain, and methodological artefacts, and (d) the effects of noise and errors. Despite a broad awareness of the nature of biological data, biological-data-specific issues have not been extensively reported in the bioinformatics literature. This awareness is exemplified in the words of Altschul et al. (1994):

> "Surprisingly strong biases exist in protein and nucleic acid sequences and sequence databases. Many of these reflect fundamental mosaic sequence properties that are of considerable biological interest in themselves, such as segments of low compositional complexity or short-period repeats. Databases also contain some very large families of related domains, motifs or repeated sequences, in some cases with hundreds of members. In other cases there has been a historical bias in the molecules that have been chosen

for sequencing. In practice, unless special measures are taken, these biases commonly confound database search methods and interfere with the discovery of interesting new sequence similarities.''

3.2.1 Complexity underlying biological data

Biological data are sets of facts stored in databases, which represent measurements or observations of complex biological systems. The underlying biological processes are highly interconnected and hierarchical; this complexity is usually not encoded in the data structure, but is a part of the background knowledge. Knowledge of the biological process from which data are derived enables us to understand the domain features that are not contained in the data set. Raw information thus has a meaning only in the broader context, understanding of which is a prerequisite for asking ''right'' questions and the subsequent selection of the appropriate analysis tools. According to Benton (1996), the complexity of biological data is due both to the inherent diversity and complexity of the subject matter, and to the sociology of biology.

3.2.2 Fuzziness of biological data

Biological data are quantified using a variety of direct or indirect experimental methods. Even in a study of a clearly delineated biological phenomenon a variety of experimental methods are usually available. An experimental method is considered useful if a correlation can be established between its results and a studied phenomenon. This correlation is rarely, if ever, perfect. Distinct experimental methods in the study of the same biological phenomenon would generally produce sets of results that overlap, but not fully. Comparing these results involves scaling and granularity issues. Within the same experimental method, differences of results arise from our inability to reproduce identical conditions (e.g., temperature, pH, use of different cells or cell lines, use of chemicals from different suppliers, etc.). The quantification of the results is commonly a result of a human decision, which may vary due to calibration of equipment.

A reported quantitative result is typically the average value of several independent experiments. Quantitative biological data is fuzzy due to the inherent fuzziness of the biological systems themselves, and to the imprecision of the methods used to collect and evaluate data. Quantitative biological data therefore represent approximate measurements. On the other hand, the classes to which qualitative biological data are assigned are arbitrary, but objective in that they represent some biological facts. Biological research is largely driven by geographically dispersed individuals, who use unique experimental protocols and thus biological experimental data are produced with neither standard semantics nor syntax (Benton, 1996). Understanding the fuzzy nature of biological data is therefore crucial for the selection of appropriate data analysis tools.

3.2.3 Biases and misconceptions

Biological data are subject to strong biases due to ether their fundamental properties, presence of large families of related motifs or historical reasons (stated by Altshul et al., 1994). A set of biological data rarely represents a random sample from the solution space. Typically, new results are generated around previously determined data points. Some regions of the solution space are therefore explored in depth, while some regions remain unexplored. Historical reasons are a common cause of such biases, where a set of rules might be defined in an attempt to describe a biological system. If these rules gain acceptance by a research community, further research will be directed by applying these rules. If these rules describe a subset of the solution space, the consequence is the refinement of the knowledge of the subset of solutions that satisfies the rules, while the rest of the solution space will be largely ignored. Similarly, reasoning with limited knowledge can lead to either over- or under-simplification errors. A careful assessment of the relative importance of each data point is thus necessary for the data analysis. Improvements in the technology also influence biological data. Older data is often of lower granularity, both quantitatively and qualitatively, while newer data is often of higher precision, due to both expanded background knowledge and improved experimental technology.

3.2.4 Noise and errors

Sources of noise in biological data include errors of experimentation, measurement, reporting, annotation and data processing. While it is not possible to eliminate errors from data sets, a good estimate of the level of noise within the data helps selection of the appropriate method of data analysis. Due to the complexity of biological systems, theoretical estimation of error levels in the data sets is difficult. It is often possible to make a fair estimate of the error level in biological data by interviewing experimental biologists who understand both the processes that generated that data and the experimental methodology.

3.2.5 How to design KDD process?

When sufficient data are available and the biological problem is well defined, standard statistical methodology should be applied for the analysis. A field where this approach has been routinely used is epidemiology (Coggon et al., 1997). Although a statistical analysis of genes and proteins provides understanding of their bulk properties (Overton and Haas, 1998), the detailed understanding of the processes that functionally involve these genes and proteins is largely lacking. Most biological research, particularly in molecular biology, is conducted in domains characterised by incomplete background knowledge and uses data from various sources and of variable accuracy. In such cases,

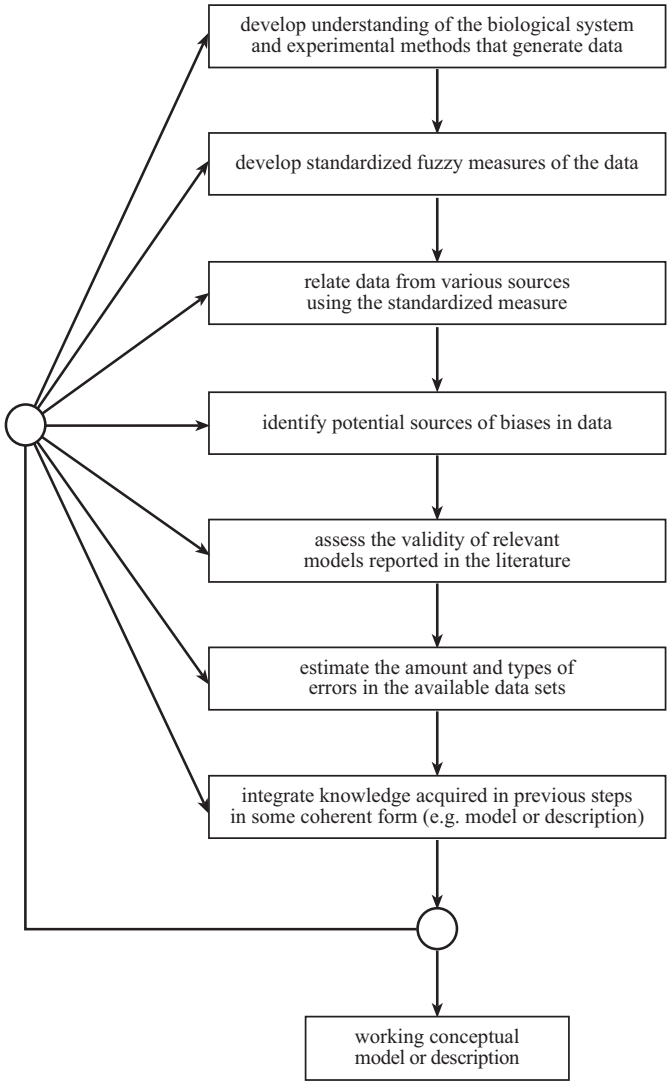

Figure 5 Data learning process. (From Brusic et al. (1998c).)

the artificial intelligence techniques are more useful than statistical techniques. To facilitate a bioinformatic analysis of biological systems, we have defined (Brusic et al., 1998c) a Data Learning Process (DLP), comprised of a series of steps (Figure 5). Iterative cycling – a refinement between any two steps (a) through (f) – can be performed. Performing the DLP steps requires significant inputs from both biologists and computer scientists and must involve two-way communication.

3.4 Database-related issues in biology

Hundreds of biological data repositories are publicly available, containing large quantities of data. A comprehensive listing of biological databases is available at Infobiogen (Discala et al., 1999). The ability to access and analyse that data has become crucial in directing biological and medical research. The Internet and World Wide Web facilitate access to data sources and also provide data analysis services. The significant research issues involved in developing and using biological databases are (a) integration of multiple data sources, and (b) flexible access to these sources.

3.4.1 Integration of heterogeneous databases

Markowitz's (1995) definition of a database is as a data repository, which provides a view of data that (a) is centralised, (b) is homogeneous, and (c) can be used in multiple applications. The data in a database are structured according to schema (database definition), which is specified in a data definition language. The data are manipulated using operations specified in a data manipulation language. Data model defines the semantics used for data definition and data manipulation languages. Biological databases are characterised by various degrees of heterogeneity in that they:

- Encode different views of the biological domain.
- Utilise different data formats.
- Utilise various database management systems.
- Utilise different data manipulation languages.
- Encode data of various levels of complexity.
- Are constantly evolving.
- Are geographically scattered.

The most popular format for distribution of biological databases is as flat files. The advances in understanding biological processes induce frequent changes in flat file formats currently being used (Coppieters et al., 1997). Popular formats for biological databases also include Sybase relational DBMS, Sybase/OPM (Chen et al., 1995), and ACeDB (Durbin and Thierry-Mieg, 1991), among others.

A comprehensive study of a particular molecular biology domain involves the analysis of data from multiple sources. Such data is often replicated at different sources. Attempts were made to overcome the problems arising from heterogeneity of the data sources and access tools (Markowitz, 1995), including:

- Consolidating databases into a single homogeneous database.
- Consolidating databases by imposing a common data definition language, data model or database management system.
- Forming *database federations* and connecting databases via the internet by maintaining hyperlinks between component databases, which preserve the individual database's autonomy.
- Forming *data warehouses* in which arbitrary subsets of data from federated databases are also loaded into a central database (e.g., Integrated Genomic Database, Ritter et al., 1994).
- Forming *multidatabase systems* which are collections of loosely coupled databases that can be queried using a common query language (e.g., in Kleisli, Davidson et al., 1997) or both described and queried by using a common data model (as in Chen et al., 1995).

Until now, the consolidating options have failed because of cost and lack of cooperation between biological database developers. Federated databases allow interactive querying of multiple

databases; however, they unfortunately offer a limited ability to perform complex queries. From the KDD perspective, data warehouses and particularly multidatabase systems are the most interesting. Multidatabase browsers which facilitate retrieval from multiple databases and cross-referencing include SRS (Etzold et al., 1996), Entrez (Schuler et al., 1996), DBGET (Migimatsu and Fujibuchi, 1996) and ACNUC (`ftp://pbil.univ-lyon1.fr/pub/acnuc`), among others. Multidatabase browsers, however, do not allow formulation of complex queries such as those required in KDD process.

3.4.2 Flexible access to biological databases

KDD requirements include both flexible access to multidatabase systems and performing complex queries. Those requirements facilitate data preparation phase of a KDD process (data preparation phase includes steps 2, 3 and 4 of Figure 1). A flexible access to the diverse biological sources is facilitated through systems such as CORBA (`http://www.mitre.org/research/domis/omg/orb.html`; Coppieters et al., 1997) or Klesli (Davidson et al., 1997).

CORBA (Common Object Request Architecture) defines a set of standards which constitute a coherent framework assessing independent data sources and their services. These standards include (a) a formal language, (b) the interface definition language (IDL) in which data and services are specified, and (c) the Object Request Broker (ORB) which is necessary to realise these services. The CORBA framework has been used for integration and interoperability of biological data resources at the European Bioinformatics Institute (Coppieters et al., 1997). However, according to Kosky et al. (1996), the IDL is not appropriate for defining database schemes and the attempts were made to combine CORBA with their OPM (Object Protocol Model). CORBA-based technology has been used for the design and implementation of genome mapping system (Hu et al., 1998) with emphasis on database connectivity and graphical user interfaces.

BioKleisli (`http://adenine.krdl.org.sg:8080./biokleisli.html`) offers high-level flexible access to human genome and other molecular biological sources. It comprises:

- A self-describing data model for complex structured data.
- A high-level query language for data transformation.
- A flexible yet precise control to enable the answering of *ad hoc* queries.

In the Kleisli environment, the typical query implementation time is reduced from weeks to days (and sometimes, hours). The architecture of the Kleisli system is given in Figure 6.

By definition, the KDD process is non-trivial and applies complex queries to data sources. The use of standards and tools such as these contained in CORBA or Kleisli systems will be essential for the future development of integrated biological applications, and consequently for the design of KDD applications in biology.

4 KDD and data mining developments in biology

Biological data accumulates exponentially in both volume and complexity. The background knowledge relevant for biological KDD system development increases continuously. The automation of the knowledge discovery is a part of this accelerating trend. The fields where the application of KDD methodologies shows an increasing importance include annotation of masses of data, structural and functional genomics, protein structure prediction and modelling, analysis of biological effects (function, signalling patterns, etc.), identification of distantly related proteins, and practical applications (e.g., drug design).

4.1 Annotation of masses of data

The current estimate of the amount of time required to double both the number of entries and the number of sequence base-pairs in DNA databases is 14–24 months. This is largely because of automated generation of Expressed Sequence Tags (EST), which now comprise more than 2/3 of the

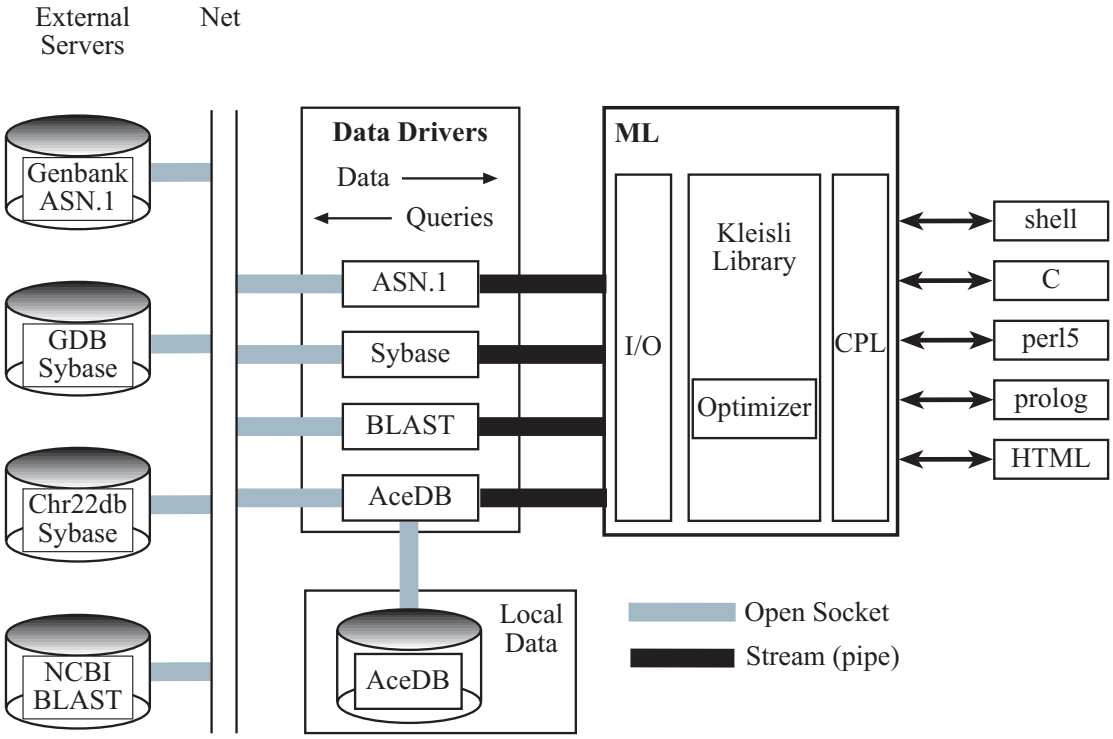

Figure 6 Architecture of the Kleisli system which facilitates access, combining and transformation of data from multiple sources. (Adapted from <http://sdmc.krdl.org.sg/kleisli/kleisli/Architecture.html>.)

database entries. Less than 10% of an estimated 10^5 human genes have been currently annotated. The components of the gene discovery include (a) gene identification, (b) gene characterisation, and (c) gene expression. A significant effort has been directed towards building computational tools for gene discovery. Tools which have been developed include GRAIL (Uberbacher et al., 1996) and the Merck Gene Index browser (Eckman et al., 1998). A detailed review on computational gene discovery can be found in Rawlings and Searls (1997). Braren et al. (1997) used information from databases to facilitate the discovery of novel genes.

4.2 Structural and functional genomics

Genomics refers to mapping, sequencing and analysis of complete set of genes and chromosomes in organisms. According to Hiether and Boguski (1997),

> "Computational biology will perform a critical and expanding role in this area: whereas structural genomics has been characterized by data management, functional genomics will be characterized by mining data sets for particularly valuable information. Functional genomics promises to rapidly narrow the gap between sequence and function and to yield new insights into the behavior of biological systems."

An initial phase of a genome analysis is construction of high-resolution genetic, physical and transcript maps of an organism – structural genomics. The advanced stage comprises the assessment of gene function by using the information and reagents provided by structural genomics. A framework for genomic analysis has been outlined by Tatusov et al. (1997).

4.3 Protein structure prediction and modelling

The structure of a protein can elucidate its function, in both general and specific terms, and its evolutionary history (Brenner et al., 1996). Numerous methods have been developed for protein

structure analysis in last two decades (e.g., see sections IV and V of *Methods in Enzymology*, Vol. 266, 1996). Nevertheless, researchers still lack the knowledge of the structure for the majority of known proteins. Secondary and tertiary structures of only 33% of all sequences in SWISS-PROT database are currently available – see the HSSP database (Dodge et al., 1998).

4.4 Analysis of biological effects

Biological systems are characterised by high degree of complexity and the processes involved are usually multi-step and involve multiple pathways. Sequence databases contain little, if any, higher level knowledge on biological systems and processes. They do, however contain voluminous amounts of low-level data. It is important to study the biological effects of the system at a high level. The relevant information is available either as expert knowledge or in the literature sources. The high-level structure can be encoded in a form of a knowledge base or as a model, which can then be used to formulate and perform complex queries. An example of a knowledge base is the RIBOWEB system (Chen et al., 1997). Promising results in modelling HIV infections were produced by rule-based cellular automaton *Cybermouse* which addresses the complexities of the immune system (Sieburg et al., 1994).

4.5 Identification of distantly related proteins

Identifying distantly related proteins is a notoriously difficult field, which is likely to continue to test the boundaries of new data mining methods. This field also provides an unifying area for the fields described in sections 4.1–4.4. Distant relations between biological sequences provide the main clues for identification and characterisation of novel sequences in the databases. The approaches include sequence similarity searches, determination of amino acid motifs, determination of conserved domains, and matching sequence patterns. The primary goal when identifying distantly related proteins is the determination of sequences that display low similarity, but which are significantly related. The discussion of issues in detection of distant similarities can be found in Catell et al. (1996). More sophisticated methods such as Hidden Markov Models (Krogh et al., 1994) are gaining popularity. Sequence pattern discovery methods are described by Brazma et al. (1998).

4.6 Practical applications

Bioinformatics is becoming an important field in drug and vaccine design. The determination of novel compounds for pharmaceutical and agricultural industries includes automated simultaneous screening of very large samples, such as compound collections and combinatorial libraries, termed High Throughput Screening (HTS). The main challenge in drug discovery research is to rapidly identify novel lead compounds. HTS produces enormous amounts of data, which are generally not matched with the ability to analyse these data, creating a bottleneck. KDD and data mining techniques will keep playing increasingly important role in this domain. Data mining techniques have been established for determination of peptide candidates for vaccines and immunotherapeutic drugs (e.g., Brusic et al., 1994, 1998a).

5 Conclusion

Current advances in biology include the development of automated methods for generation of biological data. We have been aware, that the amount, complexity and growth of genomic data will create a major challenge for bioinformatics. This growth has created a need for technologies that support automatic data handling and data interpretation. Progress in developing techniques for automatic data handling has lagged considerably behind data accumulation (Overton and Haas, 1998). The consequences of this disparity range from the persistence and spreading of erroneous information to overlooking scientific insights. KDD technology provides the means for automation

of data handling and knowledge extraction, and support for interpretation of the extracted knowledge.

Yet another problem resulting from the accumulation of data is that the selection and planning of wet-lab experiments is becoming increasingly difficult. Brusic et al. (1998b) have demonstrated that computer models can be used to complement laboratory experiments and speed up the KDD process in biology. They have provided evidence that massive scale experiments can be avoided by the judicious use of smaller-scale targeted experiments aimed at developing and validating appropriate computer models. These models can then be used for rapid and inexpensive performing of large-scale computer-simulated experiments. Computer models will grow increasingly important for biology research. The KDD technology provides the framework for effective and comprehensive use of computer models in biological research.

References

Agarwal, P and States, DJ, 1996, "A Bayesian evolutionary distance for parametrically aligned sequences" *Journal of Computational Biology* **3**(1) 1–17.

Altschul, SF, Boguski, MS, Gish, W and Wootton, JC, 1994, "Issues in searching molecular sequence databases" *Nature Genetics* **6**(2) 119–129.

Altschul, SF and Gish, W, 1996, "Local alignment statistics" *Methods in Enzymology* 2**66** 460–480.

Ashley, KD, 1992, "Case-based reasoning and its implications for legal expert systems" *Artificial Intelligence and Law* **1**(2) 113–208.

Baldi, P and Brunak, S, 1998, *Bioinformatics: the Machine Learning Approach* MIT Press.

Bates, DM and Watts, DG, 1988, *Nonlinear Regression Analysis and Its Applications* Wiley.

Benton, D, 1996, "Bioinformatics – principles and potential of a new multidisciplinary tool" *Trends in Biotechnology* **14** 261–272.

Brachman, R and Anand, T, 1996, "The process of knowledge discovery in databases: a human centered approach" In: UM Fayyad, G Piatetsky-Shapiro, P Smyth and R Uthurusamy (eds) *Advances in Knowledge Discovery and Data Mining* AAAI Press, pp 37–58.

Braren, R, Firner, K, Balasubramanian, S, Bazan, F, Thiele, HG, Haag, F and Koch-Nolte, F, 1997, "Use of the EST database resource to identify and clone novel mono(ADP-ribosyl)transferase gene family members" *Advances in Experimental Medicine and Biology* **419** 163–168.

Brazma, A, Vilo, J, Ukkonen, E and Valtonen, K, 1997, "Data mining for regulatory elements in yeast genomes" *5th International Conference on Intelligent Systems for Molecular Biology* 65–74.

Brazma, A, Jonassen, I, Eidhammer, I and Gilbert, D, 1998, "Approaches to the automatic discovery of patterns in biosequences" *Journal of Computational Biology* **5**(2) 279–305.

Brenner, SE, Chotia, C, Hubbard, TJP and Murzyn, A, (1996), "Understanding Protein Structure: Using Scop for Fold Interpretation" *Methods in Enzymology* **266** 635–643.

Brusic, V, Rudy, G and Harrison, LC, 1994, "Prediction of MHC binding peptides using artificial neural networks" <`http://www.csu.edu.au/ci/vol2/vbb/vbb.html`> In: R Stonier and XH Yu (eds) *Complex Systems: Mechanism of Adaptation* IOS Press/Ohmsha, pp 253–260.

Brusic, V, Rudy, G, Honeyman, MC, Hammer, J and Harrison, LC, 1998a, "Prediction of MHC class-II binding peptides using an evolutionary algorithm and artificial neural network" *Bioinformatics* **14** 121–130.

Brusic, V, van Endert, P, Zeleznikow, J, Daniel, S, Hammer, J and Petrovsky, N, 1998b, "A Neural Network Model Approach to the Study of Human TAP Transporter" <`www.bioinfo.de/isb/1998/01/0010/`> *Silico Biology* **1** 0010.

Brusic, V, Wilkins, JS, Stanyon, CA and Zeleznikow, J, 1998c, "Data learning: understanding biological data" In: G Merrill and DK Pathak (eds) *Knowledge Sharing Across Biological and Medical Knowledge Based Systems: Papers from the 1998 AAAI Workshop* AAAI Technical Report WS-98–04.

Cattell, K, Koop, B, Olafson, RS, Fellows, M, Bailey, I, Olafson, RW and Upton, C, 1996, "Approaches to detection of distantly related proteins by database searching" *BioTechniques* **21**(6) 1118–1125.

Cheeseman, P and Stutz, J, 1996, "Bayesian classification (AutoClass): theory and results" In: UM Fayyad, G Piatetsky-Shapiro, P Smyth and R Uthurusamy (eds) *Advances in Knowledge Discovery and Data Mining* AAAI Press, pp 153–180.

Chen, RO, Feliciano, R and Altman, RB, 1997, "RIBOWEB: linking structural computations to a knowledge base of published experimental data" *5th International Conference on Intelligent Systems for Molecular Biology* 84–87.

Chen, IA, Kosky, A, Markowitz, VM and Szeto, E, 1995, *OPM*QS: The Object-Protocol Model Multidatabase Query System* Technical Report LBNL-38181. <`http://gizmo.lbl.gov/opm.html`>

Codd, EF, 1993, *Providing OLAP (On-Line Analytical Processing) to User-Analysts: An IT Mandate* EF Codd and Associates.

Coggon, D, Rose, G and Barker, DJP, 1997, *Epidemiology for the Uninitiated. Fourth edition* <http://www.bmj.com/epidem/epid.html> BMJ Publishing Group.

Cook, D, Buja, A, Cabrera, J and Hurley, C, 1995, "Grand tour and projection pursuit" *Journal of Computational and Graphical Statistics* **4** 155–172.

Coppieters, J, Senger, M, Jungfer, K and Flores, T, 1997, *Prototyping Internet Services for Biology based on CORBA* <http://www.ebi.ac.uk/~jecop/ecoop.html> European Bioinformatics Institute.

Cover, TM and Hart, PE, 1967, "Nearest neighbor pattern classification" *IEEE Transactions on Information Theory* **13** 21–27.

Crowley, EM, Roeder, K and Bina, M, 1997, "A statistical model for locating regulatory regions in genomic DNA" *Journal of Molecular Biology* **268**(1) 8–14.

Davidson, SB, Overton, C, Tannen, V and Wong, L, 1997, "BioKleisli: a digital library for biomedical researchers" *Journal of Digital Libraries* **1**(1) 36–53.

Decker, KM and Focardi, S, 1995, *Technology Overview: A Report on Data mining* Technical Report 95–02. Swiss Scientific Computing Centre, CSCS-ETH.

Diaconis, P and Freedman, D, 1984, "Asymptotics of graphical projection pursuit" *Annals of Statistics* **12** 793–815.

Discala, C, Ninnin, M, Achard, F, Barillot, E and Vaysseix, G, 1999, "DBcat: a catalog of biological databases" <http://www.infobiogen.fr/services/dbcat>. *Nucleic Acids Research* **27**(1) 10–11.

Dodge, C, Schneider, R and Sander, C, 1998, "The HSSP database of protein structure-sequence alignments and family profiles" <http://www.sander.embl-ebi.ac.uk/hssp/> *Nucleic Acids Research* **26**(1) 313–315.

Durbin, R and Thierry Mieg, J, 1991, "A C. elegans database" Documentation, code and data available from anonymous FTP servers <lirmm.lirmm.fr>, <ncbi.nlm.nih.gov> and <cele.mrc-lmb.cam.ac.uk>.

Dzeroski, S, 1996, "Inductive logic programming and knowledge discovery in databases" In: UM Fayyad, GPiatetsky-Shapiro, P Smyth and R Uthurusamy (eds) *Advances in Knowledge Discovery and Data Mining* AAAI Press, pp 117–152.

Eckman, BA, Aaronson, JS, Borkowski, JA, Bailey, WJ, Elliston, KO, Williamson, AR and Blevins, RA, 1998, "The Merck Gene Index browser: an extensible data integration system for gene finding, gene characterization and EST data mining" *Bioinformatics* **14** 2–13.

Elder, JF and Pregibon, D, 1996, "A statistical perspective on knowledge discovery in databases" In: UMFayyad, G Piatetsky-Shapiro, P Smyth and R Uthurusamy (eds) *Advances in Knowledge Discovery and Data Mining* AAAI Press, pp 83–113.

Etzold, T, Ulyanov, A and Argos, P, 1996, "SRS: information retrieval system for molecular biology data banks" *Methods in Enzymology* **266** 114–128.

Fayyad, U, Piatetsky-Shapiro, G and Smyth, P, 1996, "From data mining to knowledge discovery" *AI Magazine* **17**(3) 37–54.

Firebaugh, MW, 1989, *Artificial intelligence. A knowledge-based approach* PWS-Kent.

Friedman, JH, 1991, "Multivariate adaptive regression splines" *Annals of Statistics* **19** 1–141.

Friedman, JH and Roosen, CB, 1995, "An introduction to multivariate adaptive regression splines" *Statistical Methods in Medical Research* **4**(3) 197–217.

Fiedman JH and Stueltzle W, 1981, "Projection pursuit regression" *Journal of the American Statistical Association* **76**(376) 817–823.

Hiether, P and Boguski M, 1997, "Functional Genomics: It's All How You Read It" *Science* **278** 601–602.

Honeyman, MC, Brusic, V, Stone, NL and Harrison, LC, 1998, "Neural network-based prediction of candidate T-cell epitopes" *Nature Biotechnology* **16**(10) 966–969.

Hu, J, Mungall, C, Nicholson, D and Archibald, AL, 1998, "Design and implementation of a CORBA-based genome mapping system prototype" *Bioinformatics* **14**(2) 112–120.

Kauvar, LM, Higgins, DL, Villar, HO, Sportsman, JR, Engqvist-Goldstein, A, Bukar, R, Bauer, KE, Dilley, H and Rocke, DM, 1995, "Predicting ligand binding to proteins by affinity fingerprinting" *Chemistry and Biology* **2**(2) 107–118.

King, RD, Muggleton, SH, Srinivasan, A and Sternberg, MJ, 1996, "Structure-activity relationships derived by machine learning: the use of atoms and their bond connectivities to predict mutagenicity by inductive logic programming" *Proceedings of the National Academy of Sciences USA* **93**(1) 438–442.

Klein, P and Somorjai, RL, 1988, "Nonlinear methods for discrimination and their application to classification of protein structures" *Journal of Theoretical Biology* **130**(4) 461–468.

Kolodner, J, 1993, *Case based reasoning* Morgan Kaufmann.

Kosko, B, 1993, *Fuzzy Thinking. The New Science of Fuzzy Logic* Harper Collins.

Kosky, A, Szeto, E, Chen, IA and Markowitz VM, 1996, *OPM data management tools for CORBA*

compliant environments Technical Report LBNL-38975. <http://gizmo.lbl.gov/DM_TOOLS/OPM/OPM_CORBA>

Krogh, A, Mian, IS and Haussler, D, 1994, "A hidden Markov model that finds genes in E. coli DNA" *Nucleic Acids Research* **21** 4768–4778.

Kubinyi, H, Hamprecht, FA and Mietzner, T, 1998, "Three-dimensional quantitative similarity-activity relationships (3D QSiAR) from SEAL similarity matrices" *Journal of Medicinal Chemistry* **41**(14) 2553–2564.

Leng, B, Buchanan, BG and Nicholas, HB, 1994, "Protein secondary structure prediction using two-level case-based reasoning" *Journal of Computational Biology* **1** 25–38.

Levin, JM, 1997, "Exploring the limits of nearest neighbour secondary structure prediction" *Protein Engineering* **10**(7) 771–776.

Li, M, Tromp, J and Zhang, L 1996, "On the nearest neighbour interchange distance between evolutionary trees" *Journal of Theoretical Biology* **182**(4) 463–467.

Markowitz, VM, 1995, "Heterogeneous Molecular Biology Databases" *Journal of Computational Biology* **2**(4) 537–538.

McCullagh, P and Nelder, JA, 1989, *Generalized Linear Models* Chapman & Hall.

Migimatsu, H and Fujibuchi, W, 1996, "Version 2 of DBGET" In: *How to Use DBGET/LinkDB* <http://www.genome.ad.jp/dbget/dbget_manual.html>

Overton, CG and Haas, J, 1998, "Case-based reasoning driven gene annotation" In: Salzberg SL, Searls DB and Kasif S (eds) *Computational Methods in Molecular Biology* pp 65–86. Elsevier.

Pearson, WR, 1998, "Empirical statistical estimates for sequence similarity searches" *Journal of Molecular Biology* **276**(1) 71–84.

Qu, K, McCue, LA and Lawrence, CE, 1998, "Bayesian protein family classifier" *ISMB* **6** 131–139.

Quinlan, JR, 1986, "Induction of decision trees" *Machine Learning* **1** 81–106.

Quinlan, JR, 1993, *C4.5: Programs for Machine Learning* Morgan Kaufmann.

Rawlings, CJ and Searls, DB, 1997, "Computational Gene Discovery and Human Disease" *Current Opinion in Genetics and Development* **7** 416–423.

Ritter, O, Kocab, P, Senger, M, Wolf, D and Suhai, S, 1994, "Prototype Implementation of the Integrated Genomic Database" *Computers and Biomedical Research* **27**(2) 97–115.

Salzberg, S, 1995, "Locating protein coding regions in human DNA using a decision tree algorithm" *Journal of Computational Biology* **2**(3) 473–485.

Schuler, GD, Epstein, JA, Ohkawa, H and Kans, JA, 1996, "Entrez: molecular biology database and retrieval system" <http://www.ncbi.nlm.nih.gov/Entrez> *Methods in Enzymology* **266** 141–162.

Shoudai, T, Lappe, M, Miyano, S, Shinohara, A, Okazaki, T, Arikawa, S, Uchida, T, Shimozono, S, Shinohara, T and Kuhara, S, 1995, "BONSAI garden: parallel knowledge discovery system for amino acid sequences" *ISMB* **3** 359–366.

Sieburg HB, Baray C and Kunzelman KS, 1993, "Testing HIV molecular biology in in silico physiologies" *ISMB* **1** 354–361.

Silberschatz, A and Tuzhilin, A, 1997, "What makes patterns interesting in knowledge discovery systems" *IEEE Transactions on Knowledge and Data Engineering* **8**(6) 970–974.

Stormo, GD, Schneider, TD, Gold, L and Ehrenfeucht, A, 1982, "Use of 'Perceptron' algorithm to distinguish translational initiation in E. coli" *Nucleic Acids Research* **10** 2997–3011.

Swets, JA, 1988, "Measuring the accuracy of diagnostic systems" *Science* **240** 1285–1293.

Tatusov, RL, Koonin, E and Lipman, DJ, 1997, "A genomic perspective on Protein Families" *Science* **278** 631–637.

Thorne, JL, Kishino, H and Painter, IS, 1998, "Estimating the rate of evolution of the rate of molecular evolution" *Molecular Biology and Evolution* **15**(12) 1647–1657.

Uberbacher, EC, Xu, Y and Mural, RJ, 1996, "Discovering and understanding genes in human DNA sequence using GRAIL" *Methods in Enzymology* **266** 259–281.

Weiss, SM and Kulikowski, CA, 1991, *Computer Systems that Learn* Morgan Kaufman.

Whittaker, J, 1990, *Graphical Models in Applied Multivariate Statistics* Wiley.

Yu, CH, 1994, "Abduction? Deduction? Induction? Is there a logic of exploratory data analysis" *The Annual Meeting of American Educational Research Association* <http://seamonkey.ed.asu.edu/~behrens/asu/reports/Peirce/Logic_of_EDA.html>.

Zeleznikow, J and Hunter, D, 1994, *Building Intelligent Legal Information Systems: Knowledge Representation and Reasoning in Law* Kluwer Computer/Law Series 13.

Zhang, CT, Lin, ZS, Zhang, Z and Yan, M, 1998, "Prediction of the helix/strand content of globular proteins based on their primary sequences" *Protein Engineering* **11**(11) 971–979.