

Applying Knowledge Discovery to Predict Infectious Disease Epidemics

Syed Sibte Raza Abidi and Alwyn Goh

**School of Computer Sciences
Universiti Sains Malaysia
11800 Penang, Malaysia.
sraza@cs.usm.my**

Abstract. Predictive modelling, in a knowledge discovery context, is regarded as the problem of deriving predictive knowledge from historical/temporal data. Here we argue that neural networks, an established computational technology, can efficaciously be used to perform predictive modelling, i.e. to explore the intrinsic dynamics of temporal data. Infectious-disease epidemic risk management is a candidate area for exploiting the potential of neural network based predictive modelling—the idea is to model time series derived from bacteria-antibiotic sensitivity and resistivity patterns as it is believed that bacterial sensitivity and resistivity to any antibiotic tends to undergo temporal fluctuations. The objective of epidemic risk management is to obtain forecasted values for the bacteria-antibiotic sensitivity and resistivity profiles, which could then be used to guide physicians with regards to the choice of the most effective antibiotic to treat a particular bacterial infection. In this regard, we present a web-based Infectious Disease Cycle Forecaster (IDCF), comprising a number of distinct neural networks, that have been trained on data obtained from long-term clinical observation of 89 types of bacterial infections, being treated using 36 different antibiotics. Preliminary results indicate that IDCF is capable of generating highly accurate forecasts given sufficient past data on bacteria-antibiotic interaction. IDCF features a client-server based WWW interface that allows for remote projections to be requested for and displayed over the Internet.

1 Introduction

Electronic data repositories are vastly expanding with an upward trend towards storing continuous historical data, such as stock markets, foreign exchange rates, weather patterns, medical monitoring and so on. Usually, temporal or time series data streams embed within recurring patterns of behaviour/activity. Knowledge discovery or data mining then entails the detection of intrinsic recurring patterns from past temporal data, thereby rendering the opportunity to exploit the discovered knowledge to predict future behaviours (given the present and past states) of the

system. Mathematically, any prediction system attempts to predict future values— $x(t+\Delta t) = f(x(t), x(t-\Delta t), x(t-2\Delta t), \dots)$ given sufficient data collected over some elapsed time period.

Predictive modelling based on temporal data (or in general time-series forecasting), in a knowledge discovery or data mining context, is regarded as the problem of deriving predictive knowledge from historical data—information about past behaviour is used to automatically generate a model of the system that can be used to predict future behaviour [1] [2]. For example, a foreign exchange broker might want to predict the future currency exchange rates; a hospital administrator might want to predict the rate of admission of patients to the hospital; healthcare professionals might want to assess the future effects of certain drugs on particular infectious organisms; a marketing executive might want to predict whether a particular consumer will switch brands for a specific product and so on. In summary, time-series forecasting is widely employed for complex non-linear systems in areas ranging from financial markets to weather to medicine [3] [4] [5] [6].

Risk management is an innovative domain that can benefit from knowledge discovery activities. Modern risk management strategies advocate the use of large quantities of historical data (collected from the subject system) to build models that can assess future risk situations—to predict beforehand a possible disaster situation and to circumvent it by taking pre-emptive measures. In the realm of risk management we are particularly interested in applying predicting modelling techniques for infectious-disease epidemic risk management. In a typical infectious-disease control scenario, the strategy is to eradicate the culprit bacteria before it has the chance to spread and infect a larger population. The most widely practised method to eradicate bacterial organisms is the use of stipulated antibiotics. Each infectious disease is propagated by a particular type of bacteria that is susceptible to only a few distinct antibiotics (these antibiotics may differ in chemical composition yet they instigate the same mode of action towards the bacteria). Nevertheless, bacteria are quite resilient—they have the tendency to develop a temporary immunity towards a certain antibiotic, hence rendering it temporarily ineffective. Although, doctors know the bacterial origin of each infectious disease and the corresponding set of effective antibiotic drugs, yet they have no means of ascertaining the current effectiveness of an antibiotic towards the treatment of a certain bacteria. In practice, doctors choose an antibiotic from the set of effective antibiotics, in case the first choice antibiotic is ineffective then another antibiotic is prescribed with the hope of it being relatively more potent. In this scenario, prior knowledge of an antibiotic's effectiveness in treating a bacterial organism can potentially play a major role in controlling an infectious disease epidemic. Effective infectious-disease epidemic risk management then encompasses predictive modelling, based on past bacteria-antibiotic interactions, to determine the *future* effectiveness of candidate antibiotics towards a bacteria, and then choosing the most effective antibiotic to culminate the spread of the infectious disease.

The emergence of the artificial Neural Network (NN) paradigm has provided an innovative methodology for temporal data analysis, in that temporal data can be supplied as 'training' input—so as to produce empirically correct relationships

between past, present and predicted future values. This paper primarily introspects the possible efficacy of NNs towards knowledge discovery activities, in particular predictive modelling. The argument is extended further by demonstrating how NNs can be effectively used for predictive modelling—the problem domain addressed is epidemic risk management. In this regard this paper examines the temporal fluctuations in bacterial susceptibility towards a given antibiotic, which medical professionals have long suspected of possessing a “recurrent” pattern. Such recurrent patterns of a bacteria’s susceptibility towards antibiotics are captured by a NN system that later provides a forecast of the behaviour of a bacteria towards various antibiotics [7]. It is argued that having a predictive model for bacteria-antibiotic interactions would be enormously useful, in that doctors usually have a choice of several antibiotics with which to treat a particular bacterial infection. Reliable “future” knowledge of how one antibiotic, amongst a possible choice of several antibiotics, can be used with optimal effect at a particular time can have significant implications towards the control of spread of infectious diseases. Finally, we present a web-based *Infectious Disease Cycle Forecaster* that allows for projections to be requested for by remote healthcare practitioners and displayed over the Internet.

2 The Essence of Predictive Modelling

Knowledge discovery manifests a synergy of a diverse set of computational technologies to address a central objective: to glean information that is buried in the enormous stocks of collected data and to develop and underpin strategies for improved decision making support. Of the many facets of data mining, we are particularly interested in predictive modelling—the analysis of historical data to discover predictive patterns [8] [9].

Typically, the exercise of mining useful information from data is predicated by the specification of the needs and goals. But, it is also true to state that the nature of the data circumscribes the nature of the knowledge that can be derived from it—the available data defines the scope of the problem, more so the data *defines* the problem and in turn the *extractable* knowledge [10]. Predictive modelling is characteristic of such *data-defined* problems as the mathematical model of the system generating the time-series data is not available, rather the phenomenon/functions realising the time-series data are concealed within the collected data. Predictive modelling encompasses the formation of a descriptive model of the system in question, either by inductive or deductive means, and then exploiting the model to predict future system behaviour. For that matter, within the knowledge discovery paradigm, predictive modelling can be regarded as a *discovery-driven* data-mining operation.

Mining predictive quality information is more an inductive problem—in the absence of both an abstract specification of the system generating the data and the non-linear and non-monotonic relationships between the data items in the data set, inductive learning techniques are more appropriate to produce a ‘generalisation’ of the functions governing the system, i.e. a descriptive model of the system. The generalisation—maybe a ‘trained’ NN—interpolates between and extrapolates beyond

the data items used for its construction and hence can compute relations/values beyond those used to develop it. In a knowledge discovery context, predictive modelling of time-series data within an inductive learning paradigm can be understood as follows: The inductive learning system—a NN for that matter—is given a set of instances (derived from the data set) of the form (x, y) , where y represents the variable that needs to be predicted by the system, and x is a vector of representative features deemed relevant to determining y . The inductive learning system is to induce a general mapping from x vectors to y values by way of building (or rather implicitly learning) a prediction model, $y = f(x)$ of the unknown/inherent function f , that allows the prediction of y values from unseen x vectors. For any prediction activity, knowledge about the current state of the system is essential as the predicted value is a function of the current state(s). In sum, the inductive learning system accounts for the *regularities* hidden within a seemingly arbitrary data set and uses the ‘learnt’ generalisation to predict future values of some variables.

Given that predictive modelling involves inductive model-development activities, the computing literature offers a suite of inductive techniques that are candidates for performing predictive modelling—e.g. decision rule or decision tree induction, classifier rules, statistical linear discriminants, case based (nearest neighbour) methods, genetic algorithms and NNs [11]. Typically, predictive modelling is carried out using symbolic-induction techniques as the generated models are expressed as sets of if-then rules, and are therefore comprehensible and explainable. Yet, recently with the emergence of NNs as a powerful computational tool, with learning capabilities, there is a strong case for using NNs for predictive modelling.

3 The Efficacy of Neural Network for Effective Knowledge Discovery/Data Mining Activities

NNs have a natural propensity to learn—they learn how to solve problems from acquired/generated data (from the problem domain) as opposed to solving problems based on explicit problem specification. Furthermore, the learning characteristics of NN enable them to deal efficiently with noisy data—partial, incorrect and potentially conflicting data—and generalise well in situations not previously encountered. Hence, it can be argued that NN are well suited for data mining tasks, in particular to tackle *data defined problems* [12] [13].

Data mining literature does not seem to support the above conjecture. Despite the above-mentioned efficacy of NNs towards various knowledge discovery and data mining activities, predictive modelling being a prime candidate, NNs are not commonly used for data mining tasks. One explanation for this apparent lack of acceptance of NN by the data mining community is that *trained NNs are usually not comprehensible*—they are ‘black boxes’ with no explanation on how they solved the problem. Below we will attempt to justify the efficacy of NNs for data mining, in particular predicative modelling.

Indeed, NNs when applied for predictive modelling applications do not render any symbolic rules or explanations towards the operational characteristics or phenomenon governing the system in question. It is not the case that NN do not possess such knowledge; on the contrary, the knowledge learnt from data by the NN—a model of the system—is sub-symbolic in nature and it is encoded using real-valued parameters (connection weights) and distributed representations within the NN.

From a pragmatic point of view, the goal of most time-series forecasting/predictive modelling applications is to predict the future values to be generated by the system (based on the present state of the system), as opposed to the understanding of the phenomenon that would lead to the generation of those values. Of course, symbolic rules and explanations may be desired to get a better understanding of the system itself, and for that matter current research efforts in NNs endeavour to generate the rules learnt by the NN from the training data [14] [15]. Notwithstanding the fact that the availability of explicit ‘system defining’ rules would make NNs more favourable for knowledge discovery/data mining activities, yet it may be noted that most of the rules currently derived from NNs are primarily geared towards classification and clustering problems [16]. As much as the rules generated from learnt NNs can identify salient/discriminant features (in the data set), responsible for eventual classifications of the data items, it is yet to be seen how well NNs being trained on temporal data can (a) explicate their knowledge of the complex system they are modelling and (b) why and what attributes are significant in determining the future values of the system. It can be argued that the richness and the temporal nature of time-series data used in predictive modelling renders difficulties in generating meaningful rules.

In conclusion, we argue that maybe it is not even useful to look for ‘system defining’ rules within a NN performing predictive modelling, rather a validation of the NN results based on collected data should be used as a benchmark to determine its efficacy towards forecasting future values. In relative terms, the efficacy of NN for predictive modelling is further validated by the fact that predictions performed by NN are derived from an inherently ‘learnt’ mathematical model of the system. On the contrary, real-life explanations of temporal systems are at best speculative—experts within the area of investigation usually provide a subjective and speculative analysis of the causes for the temporal behaviours of the system—devoid of credible mathematical models of the system.

4 Neural Network Based Predictive Modelling

In general, real-life observational data is difficult to model using linear statistical models based on auto-regression or moving averages. NNs [17] have been shown to be able to decode non-linear time series data which adequately describes the characteristics of the time-series. Information contained in the NN’s weighted synaptic connections—assuming a sufficiently rich architecture—enables the NN to calculate forecasted values that fit into the non-linear trend presumably present in the past values of the time-series.

NN is an information processing paradigm inspired by the architecture and distributed processing methodology of the biological nervous system. It is composed of a large number of functionally simple, but highly interconnected, processing units (neurons) which are normally organised into layers i.e. input, output and at least one “hidden” layer interposed between the input and output layers. The inter-neuron synaptic connections constitute a connection-weight matrix which can be modified during “training” so as to better reproduce the non-linear mappings between the input pattern and the corresponding desired output pattern. Training the NN requires that it be supplied with sufficient input-output vectors, from which it will hopefully “learn” the inherent behavioural patterns. NN systems are therefore extremely useful for data-defined problems, characterised by the abundant availability of empirical data for which an analytic rule-based description is difficult or non-existent.

Knowledge discovery/Data mining, in particular predictive modelling, using NN need to be carried out according to five major steps. First, time-series data is collected. Second, the data is cleansed—the data is normalised and scaled in order to minimise noise. Third, an appropriate NN capable of capturing the hidden regularities within the time-series data is built by experimenting with various architectural and training parameters. Fourth, the NN is trained using the training data. Finally, the ‘learnt’ NN is extensively testing for accuracy using the validation and testing data. After the NN passes the validation criteria it can then be used for predictive modelling.

4.1 Data Collection and Pre-Processing

The bacterial sensitivity/resistivity data used in our research was provided by Universiti Sains Malaysia Hospital located in Kota Bharu, Malaysia. This data-set was compiled to observe the interactions of various microbiological organisms against several antibiotics prescribed as treatment. For each individual patient, sensitivity/resistivity variations in the infecting organism towards the antibiotic used were periodically recorded. In total, the original five-year (1993-97) study collected data on the sensitivity/resistivity of 89 organisms, for which 36 different antibiotics were prescribed. For our purposes, it was deemed appropriate to sum all occurrences of a particular bacteria-antibiotic interaction within a particular month. We tabulated monthly values for bacterial sensitivity (S), resistivity (R) and their difference (S-R); thereby producing a data-set with points at regular temporal intervals.

4.2 Data Cleansing

Data cleansing was performed in terms of normalising the data i.e. map the tabulated values onto the numerical range [0,1] prior to insertion into the input-layer of the forecaster. The numerical range [0,1] was important as the NN employed the binary sigmoidal function. Data normalisation was done firstly by calculating the differences between successive monthly values for a particular bacteria-antibiotic

(BA) interaction ie $d_{BA}(t_i) = y_{BA}(t_i) - y_{BA}(t_{i-1})$ where $y(t_i)$ is the sensitivity/resistivity for the i -th month

Secondly, we performed linear normalisation with respect to the maximal and minimal values for the time-series of differential values i.e.

$$x_{BA}(t_i) = \frac{d_{BA}(t_i) - \min_{BA}}{\max_{BA} - \min_{BA}} \quad (1)$$

with $\max/\min_{BA} = \max/\min\{d_{BA}(t_i), \text{ for } \forall i \text{ in time-series}\}$.

This exercise resulted in normalised vector $x_{BA}(t_i)$ in N -space for $i = -N+1, \dots, -1, 0$.

Prior to embarking on any time-series forecasting exercise it is important to establish whether the time-series in question incorporates any sort of inherent trends, otherwise by definition forecasting would be impossible. For our case, the *Random Walk Hypothesis* testing determined that the time-series of interest (for most of the BA interactions) exhibited biased random walk behaviour with a ‘‘period’’ of 3 to 4 months. This observation implies that the ‘learnt’ generalisations of the inherent trends within the time-series data can at best be used to forecast no better than three months into the future. This result allows us to determine an upper limit for the memory of the NN forecaster, i.e. $N = 3$

4.3 NN Forecaster System: Design and Training

NN, in particular backpropagation (BP) networks, can be used as effective non-linear general-purpose function approximators—the BP network is simply taught historical data of the time-series and the learnt BP network can be used to predict future outcomes.

We have used the Back-Propagation (BP) NN with sigmoidal Feed-Forward (FF) learning [18] as the basis for our forecaster. This NN model employs a supervised-learning algorithm requiring a sample time-series of form $x_i = x(t_i)$ where $t_i = t_0 + i\Delta t$ in which N past values (including the present) are used as input towards the calculation of future values for the time-series. The limits imposed by the random-walk test require three past x_{BA} values ($i = -3, -2, -1$) and the present ($i = 0$) to generate three future x_{BA} value ($i = 1, 2, 3$). Generation of the training pattern for the subsequent computational cycle requires the shifting forward of the 4-month temporal ‘‘past’’ window (now $i = -2, -1, 0, 1$) in order to generate future x_{BA} values at $i = 2, 3, 4$. The observed cyclic parameter, based on a 4 month temporal window, discussed in the previous section allows for design of a NN to model the time-series $x_{BA}(t_i)$ for $i = -3, -2, -1, 0, 1, 2, 3$; with $i = -3, -2, -1, 0$ as input and $i = 1, 2, 3$ as output. This corresponds to a NN with 4 input and 3 output units. Experiments subsequently determined the optimal network configuration to be one with 10 hidden units.

Intuitively, one expects the predicted accuracy to degrade with temporal distance, this will be demonstrated in the next section. Finally, we have considered $x_{BA}(t_i)$ and $x_{BA}(t_{i-1})$ to be independent time-series, essentially for simplicity. This means we do

not take into account the effects of interaction BA on interaction BA', although work is currently in progress to quantify this assumption.

To develop the most efficacious NN model four different time-series representation schemes were studied, namely:-

1. (S – R) time-series data used to train a single NN
2. S and R data each used to train separate NNs
3. Differential (dS – dR) data used to train a single NN
4. dS and dR data each used to train separate NN

Option four realised the best NN model as it appeared that the difference between past values of R/S reflects the magnitude of fluctuations between the R/S values with time. For forecasting purposes this is a more important indicator of the behaviour of the system as opposed to the actual values of R/S. Also, it was observed that keeping the S and R values in separate networks yielded better forecasting results. This observation is in accordance with the fact that, in theory, the sensitivity and resistivity parameters are not reciprocal to each other. Hence, in practice, they are not supposed to influence each other and can thus be modelled by separate NNs.

4.4 Discussion of Experimental Results

The major problem encountered in our work was the occasional absence of documented data. The missing data value are inserted using linear interpolation so as to reconstruct the complete time series required for NN training. When uninterrupted data does exist, the performance of the NN forecaster tends to be reasonably accurate. Fig. 1 shows, monthly (actual and forecasted) occurrences of *Staphylococcus Aureus* sensitivity towards *Cefuroxime* (Graph P) and *Acinetobacter* sensitivity towards *Amikacin* (Graph Q). As can be seen in Graphs P and Q (shown in Fig. 1), the time-series of 1-month forecasted data tends to “track” fairly well with the actual documented incidence of bacterial-antibiotic sensitivity. The 2-month and 3-month predicted time-series are successively less accurate.

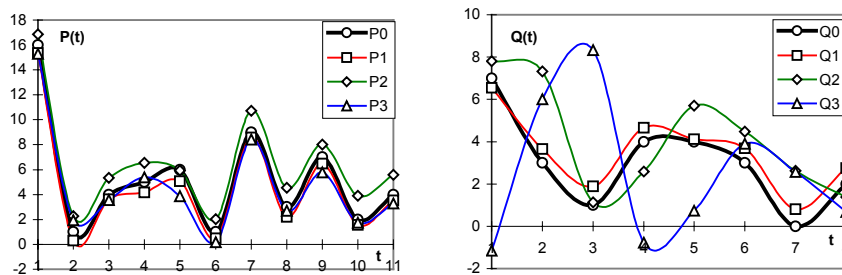


Fig. 1. Monthly actual and forecasted BA occurrences. In the legend, $(P/Q)_0$ = Actual recorded data; $(P/Q)_1$ = 1-mth; $(P/Q)_2$ = 2-mth and $(P/Q)_3$ = 3-mth predicted data

In order to ascertain the absolute accuracy of the system, we subtracted off the actual data values from each predicted time-series. Fig. 2 shows two graphs illustrating the

differential comparison of monthly predictions against actual data for *Staphylococcus Aureus* sensitivity towards *Cefuroxime* (Graph P) and *Acinetobacter* sensitivity towards *Amikacin* (Graph Q). From Fig. 2, we are thus able to conclude that the 1-month forecaster for both time-series produces output correct to within ± 1 occurrences of sensitivity, with the 2-month and 3-month predictions being successively less accurate. Note that the discrepancy between predicted and recorded data tends to peak at the extremal points on the actual time-series, and that both predicted—especially the 1-month time-series—and recorded data tend to share similar minima and maxima. The system is hence able to predict reasonably well whether BA sensitivity is increasing or decreasing, information which is probably more important than a numerical forecast. Comparable levels of accuracy are typical for the BA interactions in our analysis, leading us to conclude that NN-based forecasting is indeed a reasonably accurate tool to counter bacterial infections.

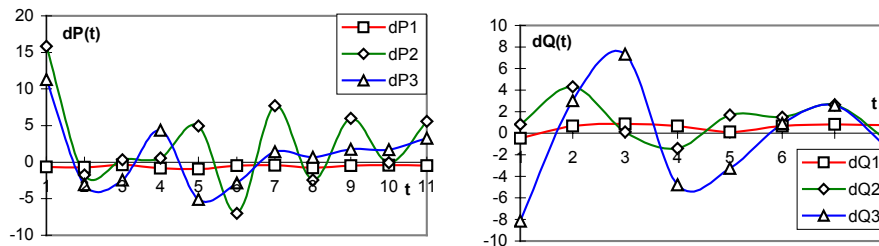


Fig. 2. Differential comparison of monthly predictions against actual data. *In the legend, $d(P/Q)_1 = 1$ -mth; $d(P/Q)_2 = 2$ -mth; $d(P/Q)_3 = 3$ -mth predicted differential*

5 Infectious Disease Cycle Forecaster

The Infectious Disease Cycle Forecaster (IDCF) is the end-product of our research efforts. It allows remote health-care practitioners to forecast the behaviour of a bacteria/organism against one or more antibiotics. With increasing user-acceptance, we migrated IDCF from a stand-alone system to a web-based client-server system. The clients (remote healthcare professionals) run the user-interface front end using any Web browser to the IDCF server. Transactions between the client and server is via HTML pages—the client’s web browser sends forecasting requests (names of bacterial organisms, antibiotics, duration of forecast and so on) and BA input values to the IDCF server, CGI programs initiate and co-ordinate all NN operations and calculations and finally results (i.e. forecast reports and graphs) are again sent to the client in the form of HTML pages.

The first panel in Fig. 3 shows the IDCF main input screen which requires specification of the organism, a list of interacting antibiotics, the nature of the forecast profile need to be generated, and the predictive time frame. The forecasted results are displayed on a dynamically generated Web-page (as shown in the second panel in Fig.

3), which illustrates the future S/R trends of the organism against the various selected antibiotics.

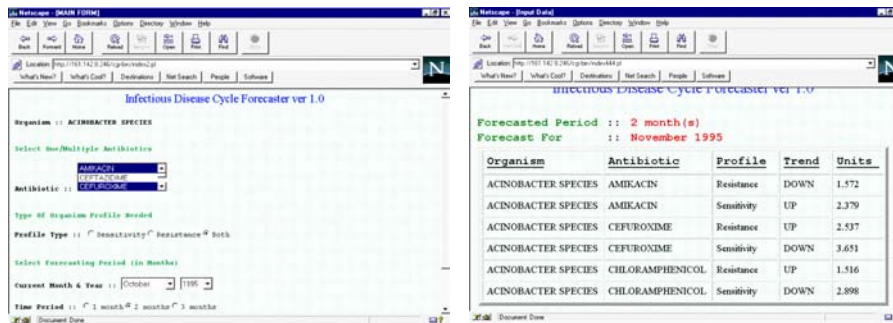


Fig. 3. The main screen of IDCF's web interface, followed by the forecast report for an exemplar bacteria-antibiotic interaction

The numerical data describing the future trend of certain bacteria-antibiotic interactions can be viewed as follows:-

- A single graph depicting either of the S or R profiles for one or more antibiotics (first panel of Fig. 4)
- A single graph showing both the S and R profiles for one antibiotic (second panel of Fig. 4)

with all graphs are generated dynamically from the NN output corresponding to one or more BA interactions.

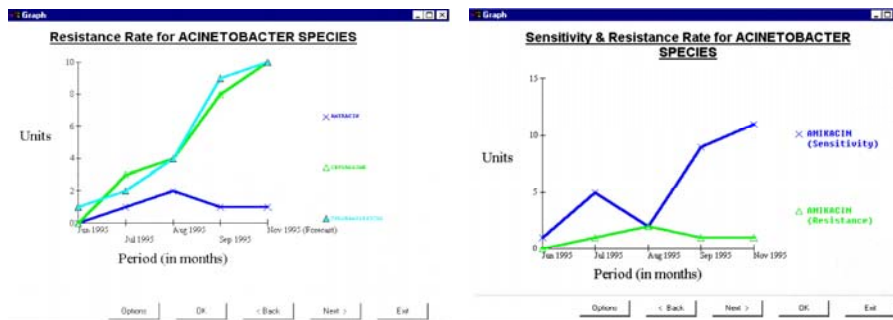


Fig. 4. Sample graphs illustrating the profiles of 3 different antibiotics against a common organism, and the S/R profile of a single antibiotic against a bacteria

6 Concluding Remarks

Typically, risk management—a scenario in a healthcare context being epidemic management—relies on human analysts to perform the necessary analysis of historical data and come up with a damage minimisation plan. However, with large databases storing continuous historical data, any attempt at real-life prediction involves the introspection of thousands of historical data items, while trying to deduce the inter-relationships between the data items. Given this reality, we have argued that NN can serve as *normative decision-support systems* to predictive modelling problems—NN can tackle *data-defined* problems by way of generating a mathematical model of the system from the collected time-series data. The NN mathematical model, i.e. a generalisation of the system, not only automatically and inductively incorporates all the inherent inter-relationships between the various data items but it also exploits the ‘learnt’ knowledge to predicts future values.

In a typical disease control scenario, the strategy is to eradicate the culprit bacteria before it has the chance to spread and infect a larger population. In this regard the utility of IDCF is paramount as it is able to systematically (and accurately) generate “future” knowledge of how one specific antibiotic—amongst a possible choice of several drugs—can be used with optimal effect at a particular time. Based on an IDCF-supplied prediction, informed decisions, such as discontinuation of less (or soon-to-be less) effective antibiotics in favour of more effective antibiotics, can subsequently be made by the respective agencies. Furthermore, major healthcare institutions will find it helpful to know ahead of time that they need to maintain sufficient quantities of the drugs projected to be most useful in dealing with certain infections, while perhaps reducing stockpiles for other less effective ones.

Finally, our results are significant because we have demonstrated that (a) NNs can offer a practical and automated solution to the problem of discovering knowledge from historical data to perform predictive modelling, and (b) NN-based learning techniques can provide ‘intelligent’ predictive modelling systems at a significantly lower cost in time and resource than traditional knowledge engineering.

References

1. Piatetsky-Shapiro, G. (ed.): Knowledge Discovery in Databases. J. Intelligent Information Systems: Integrating Artificial Intelligence and Database Technologies. 4 (1). (1995)
2. Weigend, A., Gershenfeld, N. (eds.): Predicting the Future and Understanding the Past. Addison Wesley, Redwood City (1993)
3. Tierney, W., Murray, M., Gaskins, D.L., Zhou, X.: Using Computer-Based Medical Records to Predict Mortality Risk for Inner-City Patients with Reactive Airways Disease. J. American Medical Informatics Association, 4(4) (1997) 313-321
4. Jasic, T., Poh H.L.: Financial Time-Series Prediction using Neural Networks: A Case Study for the TOPIX Data. Proc. Sixth Australian Conf. Neural Networks. Sydney (1995)
5. Refenes, A.N., Zaprani, A., Francis, G.: Stock Performance Modelling using Neural Networks: A Comparative Study with Regression Models. Neural Network. 5 (1995)

6. Liu, J., Wong, L.: A Case Study for Hong Kong Weather Forecasting. Proc. Int. Conf. Neural Info. Processing. Hong Kong (1996)
7. Abidi, S. S. R., Goh, A.: Neural Network Based Forecasting of Bacteria-Antibiotic Interactions for Infectious Disease Control. Ninth World Congress on Medical Informatics. Seoul (1998)
8. Apte, C., Hong, S.J.: Predicting Equity Returns from Securities Data with Minimal Rule Generation. In: Fayyad, U.M., Shapiro, G.P., Smyth, P., Uthurusamy, R. (eds.): Advances in Knowledge Discovery and Data Mining. AAAI Press, California (1996)
9. Berndth, D.J., Clifford, J.: Finding Patterns in Time-Series: A Dynamic Programming Approach. In: Fayyad, U.M., Shapiro, G.P., Smyth, P., Uthurusamy, R. (eds.): Advances in Knowledge Discovery and Data Mining. AAAI Press, California (1996)
10. Partridge, D.: The Specification and Implementation of Data-Defined Problems. Proc. Data Mining. London (1996)
11. Shavlik, J., Mooney, R., Towell, G.: Symbolic and Neural Net Learning Algorithms: An Empirical Comparison. Machine Learning. 6 (1991) 111-143
12. Lu, H., Setiono, R., Liu, H.: Effective Data Mining using Neural Networks. IEEE Trans. Knowledge and Data Engineering. 8(6) (1996) 319-327
13. Lu, H., Setiono, R., Liu, H.: Neurorule: A Connectionist Approach to Data Mining. Proc. VLDB. (1995) 478-489
14. Towell, G. & Shavlik, J.: Extracting Refined Rules from Knowledge-Based Neural Networks. Machine Learning. 13(1) (1993) 71-101
15. Andrews, R., Diederich, J., Tickle, A.B.: A Survey and Critique of Techniques for Extracting Rules from Trained Artificial Neural Networks. Knowledge-Based Systems. 8(6) (1995)
16. Kaski, S., Kohonen, T.: Exploratory Data Analysis by the Self-Organising Map: Structures of Welfare and Poverty in the World. Proc. Third Int. Conf. Neural Networks in the Capital Markets. London (1995)
17. Haykin, S. Neural Networks. Macmillan, New York (1994)
18. Rumelhart, D., McClelland, J.: Parallel Distributed Processing: Explorations in the Macrostructure of Cognition. MIT Press, New York (1986)