

Data Mining Using Neural Network Approaches Using SAS[®] and Java[™]

David Bell, State of California Genetic Disease Branch

ABSTRACT

This paper will explore the uses of two Neural Net approaches to pattern analyses based upon a neuro- evolutionary approach based upon the ID3 AI algorithm created by J. R. Quinlan and later neuro-evolutionary approaches. The first approach will use an approach devised for detecting genetic structures from genetic expression known as REVEAL (REVerse Expression evaluation Algorithm: Liang S, Fuhrman S, et al.); the second algorithm will be based upon a Neuro-dynamic Programming approach, to detecting patterns in data (D. Bertsekas, et al.: 1996). Both algorithms use relative Entropy to measure the amount of information gained by the variables used for pattern detection as well as to determine the error of pattern association.

Two examples will be presented using these algorithms. The first example will be the use of these algorithms for record matching. The second example will use these two methods to determine pattern changes (i.e., mutations) over time as would be the case in an evolutionary study or even a temporal spatial study.

Results will be presented to show the performance and accuracy merits and drawbacks of the two methods compared.

INTRODUCTION

Pattern matching and pattern evolution is becoming an emerging field in the area of data mining.

In data matching with messy data, the pursuit is that of finding correctly linked data within a certain certainty level (i.e., a valid positive). It is also quite important to rule out the possibility of linkage when valid (i.e., a valid negative). Of course one also has to be able to quantify uncertainty of false match as well.

In the area of discovering and quantifying change, the goal is that of detecting adaptive and maladaptive changes in an organism or behavioral patterns. It is also important to detect what are core versus peripheral elements in an organism or behavioral pattern. For example, in the area of criminal behavior, it has been observed that often "model prisoners" have high recidivism rates when released on parole or probation. The question becomes whether there were actually changes happening to core behavior sets while being incarcerated, or were the changes more peripheral and temporary while leaving the pathological core quite intact. In genetics, the question becomes a matter of inherited versus ecology driven mutations in mapping organism change. In terms of GIS and health services delivery maps, the questions of why certain subpopulations are underserved may lie areas sharing common barriers to accessing services such as language and/or income/insurance barriers.

Therefore the experiment using the aforementioned methods has two goals:

1. Detect linkages in a database.
2. Detect common mutational patterns within the database over time.

THE EXPERIMENT

In the interest of discovering solutions to these two problems, two methods were employed to detect linkages and model changes in a medical records database. The first goal was to detect matches over what a deterministic/non-dynamic linkage program could detect with the purpose of identifying clients for follow-up and service delivery quality. The second goal was to discover what attributes of the data set were more prone to change and error versus what attributes tended to be more stable in order to determine what factors might affect data quality. The data source was from the prenatal and newborn screening data sets of the genetic disease branch. Two data sets were selected: a "source" data set consisting of 48, 000 records; and a "target" data set containing 65,000 records.

THE METHODS

The comparison here was to determine which method was the most accurate and most time efficient.

The first method to be used (REVEAL) was based upon using generalized entropy measurements of the relative distances between variable values to establish pattern matches using Bayesian probability calculations on a maximum of 4 factors.

The second method was based upon ID3 and multilayer perceptron (MLP) methods. From ID3, it inherited the concept of inductive analysis of the data set and unsupervised learning. From MLP methods it inherited the basis Markovian decision structure. It also uses the structure inherent with Neuro-Dynamic programming. The result is an Adaptive Neuro-Evolutionary program (ANEP).

The reason ID3 was not used exclusively was that it did not perform more accurately than the REVEAL method (REVEAL had a 78.4% success rate; ID3 had a 77% success rate although it was faster). The MLP method did outperform both REVEAL and ID3 (92.3% success rate), but it was nearly as slow as REVEAL (REVEAL took approx. 3hrs. 27 min. to process the data; MLP took 3hrs 35 min to process the data). MLP also required more operator supervision to setup and run the BackPropagation learning program used to generate linkage weights.

SAS[®] was used to extract the data from the larger data sets, produce deterministically a linked dataset, generate the Metaphone values for the unlinked data, sort the data by Metaphone values, and output unlinked data sets to be processed by the REVEAL program and the Adaptive Neuro Evolutionary program (ANEP) which were written in Java[™]. Figure 1 diagrams the procedure.

Finally, the methods were compared for accuracy and efficiency by measuring the number of records matched within a 95% probability threshold to the total number of records within the source dataset.

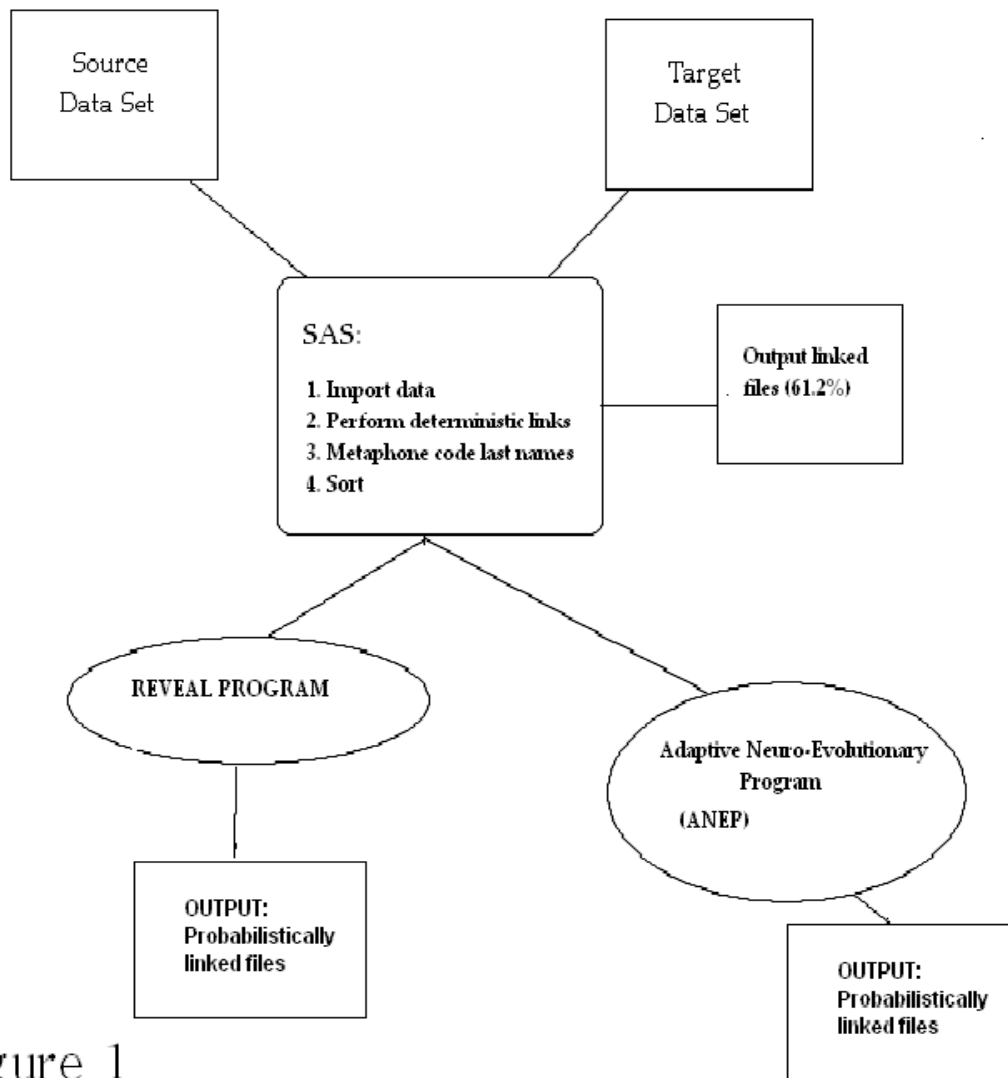


Figure 1

BRIEF METHODS DESCRIPTION:

REVEAL

REVEAL stands for REVerse Engineering Algorithm. It was developed for the analysis of genetic similarities and mutations. It uses information theory and relative distance measurements to determine genetic similarities from gene expression data. The goal is to minimize classification entropy within a classification schema thereby detecting stable classification clusters within a data set. The procedure is as follows:

```
Begin:
Get_Data(SourceData);
Get_Data(TargetData);
Get_Source_Vars(Identifiers[]);
Get_Target_Vars(Identifiers[]);
Calculate_Distances(Source_Vars,
                    Target_Vars)
                    Return(DistanceMatrix[][]);
CalculateEntropy(DistanceMatrix[][]);
                    Return(EntropyOfVars[]);
CalculateBayesProbs(EntropyOfVars[]);
End;
```

REVEAL does a good job of detecting similar patterns; however it does have two drawbacks:

1. It is limited to a maximum of four identifiers or factors due to computer intensive methods.
2. It is slow.

ANEP

ANEP stands for Adaptive NeuroEvolutionary Program. Like REVEAL, it too uses information theory to determine similarity of elements within identifier variables. The goal is to achieve convergence by minimizing information gain compared to state transition distance. This results in stable classification groups. The outline of the program is:

```
Begin:
Get_Data(SourceData);
Get_Data(TargetData);
Get_Source_Vars(Identifiers[]);
Get_Target_Vars(Identifiers[]);
LOOP:
CalculateEntropy(SourceVars,TargetVars)
                    Return(EntropyOfVars[]);
CalculateTransitionProbability(SourceVars, TargetVars);
CalculateStateTransitionEntropy(CalculateEntropy(SourceVars,TargetVars))
                    Return STE;
CalculateInformationGain(action taken,STE)
                    Return InfoGain;
CalculateControllability(InfoGain/STE)
                    Return(Controllability);
If(max(Controllability)) then End;
Else LOOP.
End;
```

RESULTS

In terms of record matching within a 95% probability threshold, the following was achieved by deterministic, REVEAL and ANEP methods:

TABLE OF RESULTS

M e t h o d	P c t M a t c h	I n c r e a s e	T i m e
D e t e r m i n i s t i c	6 1 . 2	n . a .	4 0 m i n s .
R E V E A L	7 8 . 4	1 7 . 2	3 h r . 2 7 m i n s .
A N E P	9 3 . 7	3 2 . 5	2 h r s . 5 m i n s .

From the results, it appears ANEP was faster and more accurate than REVEAL in identifying pattern similarities in the data sets.

ANEP also detected that the factors that showed the greatest stability were:

1. Metaphone Last Name
2. Date of Birth of Mother.

The variables with the least stability mostly due to missing data and incorrectly entered values were:

1. Social Security Number
2. Phone Number
3. First Name

Common mutational patterns detected were:

1. First name was aliased to same value for some subgroups
2. Phone and SSNs were often coded to a string of 9's after the 4th or 5th digit. And often the digits were interchanged in cases where they did not match exactly.

CONCLUSION

The results look promising in the use of adaptive neuro-evolutionary programming methods to detect pattern similarities and map common changes/mutations over time. The next stage should be to use a larger database to see how each performs on huge databases (e.g., approx 500,000 cases as is common for 1 year of birth data in California).

Another experiment to perform would be to see how ANEP performs as a predictive method in terms of predicting mutations/ changes in the data given a genetic or behavioral history, and to describe the dynamics of mutation in the data. Aside from possible genetic applications, there may be applications in the areas of cognitive behavioral analysis and prediction, or maybe forensic behavioral prediction.

REFERENCES

Ratitch, Bohdana and Precup, Doina, *Characterizing Markov Decision Process*, McGill University Canada. <http://www.cs.mcgill.ca/~sonce>

Liang, S., D'haeseleer, Somogyi, P., R., *Gene Expression Data Analysis and Modeling*, Session on Gene Expression and Genetic Networks.

Bertsekas, D., Tsitsiklis, J. N., *Neuro-Dynamic Programming* (Belmont: Athena Scientific, 1996)

Jaynes, E.T., *Probability Theory: the Logic of Science* (UK: Cambridge Univ. Press, 2003)

Philips, Lawrence, *Hanging on the Metaphone*, Computer Language, Dec. 1990, p.39.

Liang S, Fuhrman S, Somogyi R., ***Reveal, a General Reverse Engineering Algorithm for Inference of Genetic Network Architectures.*** SETI Institute, NASA Ames Research Center, Moffett Field, CA 94035, USA.
sliang@mail.arc.nasa.gov

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Author Name
Company
David Bell
State of California: Genetic Disease Br.
850 Marina Bay Pkwy.
Richmond, CA
Phone (510)412-6211
Email: dbell@dhs.ca.gov

Note: The author has the code used for this study and makes it available via email by contacting me at dbell@dhs.ca.gov.

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Java is a trademark of Sun Microsystems, USA .