Robustness of Learning Systems Subject to Noise: Case Study in Forecasting Chaos

Steven H. Kim, Churl Min Lee, and Heung Sik Oh

Graduate School of Management
Korea Advanced Institute of Science and Technology
Seoul, Korea
Email: skim@msd.kaist.ac.kr

ABSTRACT

Practical applications of learning systems usually involve complex domains exhibiting nonlinear behavior and dilution by noise. Consequently, an intelligent system must be able to adapt to nonlinear processes as well as probabilistic phenomena. An important class of application for a knowledge based systems in prediction: forecasting the future trajectory of a process as well as the consequences of any decision made by the system.

This paper examines the robustness of data mining tools under varying levels of noise while predicting nonlinear processes in the form of chaotic behavior. The evaluated models include the perceptron neural network using backpropagation (BPN), the recurrent neural network (RNN) and case based reasoning (CBR). The concepts are crystallized through a case study in predicting a Henon process in the presence of various patterns of noise.

Content Areas: Discovery, neural networks, case based reasoning.

INTRODUCTION

Intelligent systems should be competent in dealing not only with simple processes such as linear input-output functions, but stochastic and nonlinear behavior. A critical task in practical applications ranging from business to engineering, lies in the prediction of system behavior in the presence of noise.

This paper presents a study of the robustness of data mining tools under varying levels of noise while predicting nonlinear processes which embody chaotic behavior. The evaluated models include the perceptron neural network using backpropagation (BPN), the recurrent neural network (RNN), and case based reasoning (CBR). The concepts are crystallized through a case study in predicting a Henon process in the presence of disparate patterns of noise.

BACKGROUND

Real-world systems operate in complex environments. The complexity arises from novelty, nonlinearities, and the multitude of interactions which arise when attempting to predict or control various activities. In such a milieu, important process variables can remain unidentified. Even when they are identified, their interactions may remain uncertain. This complexity and the uncertainties which are often its derivatives limit the effectiveness of traditional methods.

Fortunately, this situation can be remedied by an adaptive methodology using knowledge integration [5]. Over the past decade, a popular methodology for implementing adaptive systems has lain in the neural network. Despite its many advantages such as autonomous learning in specific contexts, the neural approach has its limitations. Among the limitations are the slow rates of learning and perhaps even more importantly, the implicit nature of the learned skill. More specifically, a neural network may yield the correct response to a query but it cannot explain the result or justify its "reasoning".

In contrast, the use of explicit knowledge allows for explanation and justification for the benefit of other entities, including an interested human observer. Examples of such high-level representation lies with declarative logic or production rules.

METHODOLOGY

In real systems, the components tend to be highly nonlinear; examples of nonlinearities are found in the step functions of static friction, the cycles of hysteresis, or the gaps of dead zones. Moreover, the plant is often poorly understood and therefore inadequately modeled. Fortunately, these limitations can be addressed by knowledge based systems which learn from experience.

Neural networks are characterized by robustness and graceful degradation. The most common type of neural network and training procedure takes the form of backpropagation (BPN). A backpropagation neural network with standard connections responds to a given input pattern with exactly the same output pattern every time the input pattern is presented. In contrast, a recurrent neural network (RNN) may respond differently to the same input pattern at different times, depending upon the patterns that have been presented as inputs in the past.

A learning system should make increasingly useful decisions as it accumulates experience. This is the express goal of the work in case based reasoning (CBR). The CBR methodology can be effective even if the knowledge base is imperfect. Certain techniques of automated learning, such as explanation-based learning, work well only if a strong domain theory exists. In contrast, CBR can use many examples to overcome the gaps in a weak domain theory while still taking advantage of the fragmentary knowledge [9]. CBR can also be used when the descriptions of the cases, as well as the domain theory, are incomplete [11].

CASE STUDY

The utility of a learning approach to forecasting a complex process may be demonstrated through a case study in predicting a Henon process. The inputs into the predictive system consist of two sources: a chaotic signal and a noise source.

In the computational study, the primary data streams consisted of the Henon process H_t, the noise process ν_t and mixtures of the two. Each mixture consisted of a convex combination of the Henon and noise processes:

$$x_t = \lambda_t H_t + (1-\lambda_t)\nu_t$$

The convex weight λ_t was a function of time. As shown in Figure 1, the three mixed modes consisted of a downward step function, a square function, and a tent function. These 3 modes, plus the pure Henon and pure noise processes, constituted a total of 5 input data patterns. Each of the 5 signal modes was digested by several learning methods, then predicted out-of-sample.

The test phase involved 200 forecasts from periods 2800 to 3000. The forecast performance by signal mode and learning model is listed in Table 1 according to the metric of MAPE. Table 2 presents a similar chart for the hit rate.

An analysis of variance for the data behind Table 1 indicated that the differences due to signal mode and to learning technique were both significant. Moreover, the interaction effects were also statistically significant.

A chi-square test for independence was performed using the hit rates from Table 2. The results revealed that interaction effects were significant at level $p < 0.03$.

One interesting issue relates to the choice of a good architecture for the CBR model. The metric of MAPE was examined as a function of the locality L (the number of neighbors) for each signal mode, holding the input vector size fixed. The results indicate that the optimal architecture depends in part on the particular signal mode to be predicted.

The hit rate for CBR was also examined as a function of the locality L with the input vector size fixed. On the whole, the accuracy tended to rise with the size of the neighborhood. To be more precise, predictive performance depended both on the signal mode and the size of the locale.

Table 3 presents a set of pairwise comparisons across environmental scenarios for BPN. For instance, the first cell indicates that the MAPE for the pure Henon process using BPN was 0.011%, while that for the step mode was 0.113%. Further, the difference was significant at $p < 0.001$.

The step mode was similar to the pure Henon process, except that half of the signal was comprised of noise after $t = 1500$. Consequently, a 50% dilution of the Henon process by noise results in a significant difference in forecasting accuracy. To take another instance from Table 3, there was no significant difference in performance between the step and tent signal modes when BPN was employed.

Table 4 presents similar data for BPN according to hit rates. Subsequently, Tables 5 and 6 enumerate the pairwise differences in environmental scenarios for RNN according to the criteria of MAPE and HR, respectively. A similar pair of charts is presented in Tables 7 and 8 for CBR.

The hit rate measures the accuracy of forecasts but ignores the mistakes. The proportions of Type I and Type II mistakes for the 5 signal modes are listed for each learning technique in

Table 9. The results indicate that CBR dominated the other two techniques for the Henon signal. Moreover, Table 9 highlights the fact that all techniques performed well for the pure Henon mode. Overall, Table 9 indicates the absence of a clear-cut winner in the context of classification mistakes.

CONCLUSION

The complexity inherent in a learning system for business and engineering applications can be addressed by the judicious use of a spectrum of methodologies from data mining. The utility of these approaches in a complex, noisy environment was tested through a simulation model. The results indicate that learning systems tend to perform well in nonlinear domains even in the presence of noise.

REFERENCES

[1] Forsyth, D., J.L. Mundy, A. Zisserman, C. Coelho, A. Heller, and C. Rothwell. "Invariant Descriptors for 3-D Object Recognition and Pose." *IEEE Trans. Pattern Anal. and mach. Intel.*, v. 13(10), 1991, pp. 971-91.

[2] Hall, R.P. "Computational Approaches to Analogical Reasoning: A Comparative Analysis." *Artificial Intelligence*, v. 39(1), 1989, pp. 39-120.

[3] Holland, J.H. *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI: Univ. of Michigan Press, 1975.

[4] Kedar-Cabelli, S. "Purpose-Directed Analogy." In *Programm of the Seventh Annual Conf. of the Cognitive Science Society*, Irvine, CA, 1985, pp. 150-9.

[5] Kim, S. H. *"Learning Systems for Process Automation through Knowledge Integration". Proc. Second World Congress on Expert Systems*, Lisbon, Portugal, 1994a.

[6] Kim, S. H. *Learning and Coordination*. Dordrecht, Netherlands: kluwer, 1994.

[7] Kim, S. H. and M.B. Novick. "Using Clustering Techniques to Support Case Reasoning." *Int. J. of Computer Applications in Technology*, v. 6(2/3), 1993: 57-73.

[8] Lenat, D. "EURISKO: A Program that Learns New Heuristics and Design Concepts: The Nature of Heuristics, III: Program Design and Results." *Artificial Intelligence*, v. 21(2), 1983, pp. 61-98.

[9] Porter, B.W., E.R. Bareiss, and R.C. Holte, "Concept Learning and Heuristic Classification in Weak-Theory Domains". *Artificial Intelligence*, v. 45(1-2), 1990: 229-63.

[10] Qian, L. and J.S. Gero. "A Design Support System Using Analogy." In J.S. Gero, ed., *Artificial Intelligence in Design '92*, Dordrecht, Netherlands: Kluwer, 1992, pp. 795-816.

[11] Sycara, K. and D. Navinchandra. "Index Transformation Techniques for Facilitating Creative Use of Multiple Cases." *Proc. 12th Int. Joint Conf. on AI*, Morgan Kaufman, Los Altos, CA, 1991, pp. 347-52.

[12] Tecuci, G. and Y. Kodratoff. "Apprenticeship Learning in Imperfect Domain Theories." In Y. Kodratoff and R. S. Michalski, eds., *Machine Learning: An Artificial Intelligence Approach, v. III, San Maeto*, CA: M. Kaufmann, 1990, pp. 514-51.

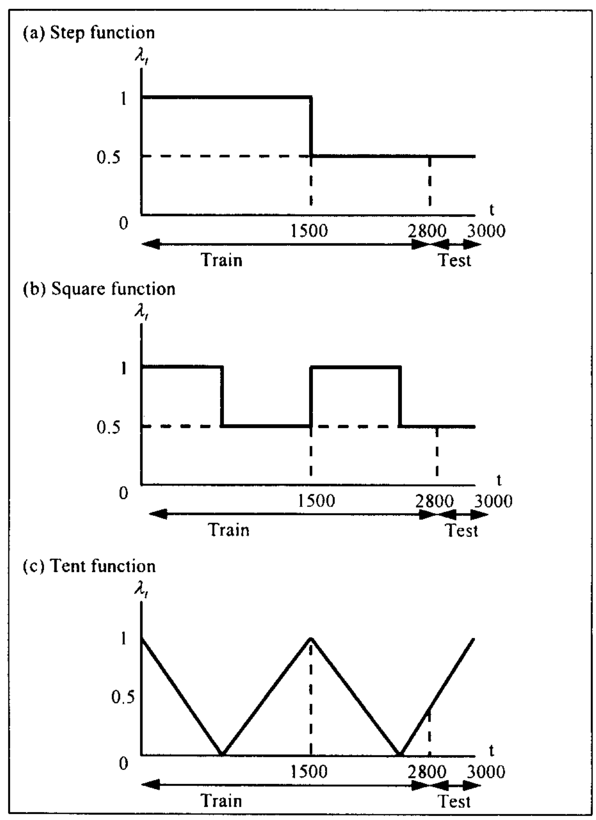

Figure 1. Variation in weighting factor as a function of time. The weighting factor λ_t is used to generate a mixed input stream composed of primary and noise sources.

Table 1. Performance by technique and mixing mode according to mean absolute percentage error (MAPE). Tables 1 and 2 examine the issue of *temporal* robustness; that is the stability of performance when a technique is trained under a particular set of conditions, then faces similar or different circumstances.

	Henon	Step	Square	Tent	Noise
BPN	0.011366	0.113030	0.112430	0.092981	0.177675
RNN	0.010869	0.132187	0.127962	0.112924	0.187926
CBR	0.001123	0.104016	0.114119	0.084794	0.214353

Table 2. Performance by technique and mixing mode according to hit rate (HR).

	Henon	Step	Square	Tent	Noise
BPN	0.995	0.780	0.770	0.945	0.725
RNN	0.985	0.730	0.760	0.835	0.730
CBR	1.000	0.800	0.810	0.925	0.670

Table 3. Stability of BPN across environmental scenarios, according to MAPE. Each cell contains 3 numbers in the format *a* : *b* (*c*). Here *a* is the performance metric in the training phase, *b* the metric in the test phase, and *c* the level of significance due to a t-test for the difference of means. Tables 3 through 8 represent an analysis of *cross-sectional* robustness for each learning technique.

	Step	Square	Tent	Noise
Henon	0.011:0.113 (.000)	0.011:0.112 (.000)	0.011:0.092 (.000)	0.011:0.177 (.000)
Step	-	0.113:0.112 (.594)	0.113:0.092 (.072)	0.113:0.177 (.000)
Square	-	-	0.112:0.092 (.021)	0.112:0.177 (.002)
Tent	-	-	-	0.092:0.177 (.000)

Table 4. Stability of BPN across environmental scenarios, according to HR. Each cell contains 3 numbers in the format *a* : *b* (*c*). Here *a* is the performance metric in the training phase, *b* the metric in the test phase, and *c* the level of significance due to a test of proportions.

	Step	Square	Tent	Noise
Henon	0.995:0.780 (1.02E-11)	0.995:0.770 (2.82E-12)	0.995:0.945 (0.003378)	0.995:0.725 (7.33E-15)
Step	-	0.780:0.770 (.810738)	0.780:0.945 (1.66E-06)	0.780:0.725 (.202505)
Square	-	-	0.770:0.945 (5.56E-07)	0.770:0.725 (.300295)
Tent	-	-	-	0.725:0.725 (3.1E-09)

Table 5. Stability of RNN across environmental scenarios, according to MAPE.

	Step	Square	Tent	Noise
Henon	0.010:0.132 (.000)	0.010:0.127 (.000)	0.010:0.112 (.000)	0.010:0.187 (.000)
Step	-	0.132:0.127 (.935)	0.132:0.112 (.182)	0.132:0.187 (.591)
Square	-	-	0.127:0.112 (.230)	0.127:0.187 (.658)
Tent	-	-	-	0.112:0.187 (.477)

Table 6. RNN across environments, using HR.

	Step	Square	Tent	Noise
Henon	0.985:0.730 (3.01E-13)	0.985:0.760 (1.53E-11)	0.985:0.835 (1.6E-07)	0.985:0.730 (3.01E-13)
Step	-	0.730:0.760 (.491268)	0.730:0.835 (.010922)	0.730:0.730 (3.01E-13)
Square	-	-	0.760:0.835 (.061998)	0.760:0.730 (.491268)
Tent	-	-	-	0.835:0.730 (.010922)

Table 7. CBR across environments, using MAPE.

	Step	Square	Tent	Noise
Henon	0.001:0.104 (.000)	0.001:0.114 (.000)	0.001:0.084 (.000)	0.001:0.214 (.000)
Step	-	0.104:0.114 (.079)	0.104:0.084 (.698)	0.104:0.214 (.000)
Square	-	-	0.114:0.084 (.200)	0.114:0.214 (.000)
Tent	-	-	-	0.084:0.214 (.000)

Table 8. CBR across environments, using HR.

	Step	Square	Tent	Noise
Henon	1.000:0.800 (2.63E-11)	1.000:0.810 (2.63E-11)	1.000:0.925 (1.4E-06)	1.000:0.670 (.000)
Step	-	0.800:0.810 (1.000)	0.800:0.925 (.012888)	0.800:0.670 (.001128)
Square	-	-	0.810:0.925 (.012888)	0.810:0.670 (.001128)
Tent	-	-	-	0.925:0.670 (2.08E-08)

Table 9. Types of error by methodology for various mode. Type I (false rejection) refers to a down prediction when the actual index rises; and Type II (false acceptance) refers to an up prediction when the actual index falls. Each entry denotes the proportion of mistakes over the trial period of 200 cases.

Method	BPN		RNN		CBR	
	Type I	Type II	Type I	Type II	Type I	Type II
Henon	0.000	0.005	0.010	0.050	0.000	0.000
Step	0.115	0.105	0.235	0.035	0.070	0.145
Square	0.115	0.115	0.200	0.040	0.115	0.115
Tent	0.050	0.005	0.150	0.015	0.045	0.045
Noise	0.200	0.075	0.205	0.065	0.255	0.125