December 1998

A Method for Incorporating Negative Data into Rule Induction

Richard Mathieu
University of North Carolina at Wilmington

Christopher Huntley
Fairfield University

Barry Wray
University of North Carolina at Wilmington

A Method for Incorporating Negative Data into Rule Induction

Richard G. Mathieu
Barry A. Wray
University of North Carolina at Wilmington

Chris Huntley
Fairfield University

Abstract

Negative data is defined by observations of unsuccessful events or poor performance. Traditional wisdom dictates that negative data be eliminated from training data sets. This paper presents a three step method for incorporating negative data into the rule induction process. The first step is to deploy rule induction using a data set containing only positive data. This is traditionally how rule induction techniques such as ID3, C4.5 and CART are used. The second step is to create a training data set that contains all of the positive data from Step 1 and also incorporates negative data. The dependent variable from Step 1 becomes a dependent variable in the new data set, and a new performance-related independent variable is defined. Decision rules are generated using the same rule induction algorithm used in Step 1. The third and final step is to reconcile the two rule sets. A step-wise procedure for creating a final, robust rule set is proposed. An example application, related to Just-In-Time manufacturing, is presented in which decision rules are generated using the classification and regression tree (CART) technique.

Introduction

Inductive reasoning starts with observed data and cases and ultimately generalizes from them to build new rules. These rules are "a natural vehicle for what we take to be the most fundamental learning mechanism: prediction-based evaluation of the knowledge store" (Holland et al., 1986). One of the critical challenges in learning a set of rules is to derive a small number of robust rules. While it is possible to derive rules from successes (positive data) and failures (negative data), traditional wisdom dictated that negative information be eliminated from the training data set. However, in a many environments it may be desirable to learn from an archived history of data that contains negative information (Triantaphyllou and Soyster, 1996) (Hall, Hansen and Lang, 1997). The purpose of this paper is to present a method that supports inductive learning in that (1) can accurately classify and predict successful and unsuccessful performance and (2) can reconcile rules generated from training sets with just positive data and rules generated with both positive and negative data.

The development of algorithms for rule induction began with Hunt's *Concept Learning System* (Hunt et al., 1966) and was followed by Quinlan's *ID3* algorithm (Quinlan, 1979). In 1984 Breiman, Friedman, Olshen and Stone (1984) developed a non-parametric statistical procedure, classification and regression trees (CART), to analyze categorical and continuous data using exhaustive searches and computer intensive testing to select an optimal decision tree. Crawford (1989) states that in cases where data is "noisy", CART is "a remarkably sophisticated tool for concept induction". Inductive learning techniques typically generate decision rules by training on data sets that contain only positive data. However, there are a wide variety of reasons for wanting to learn from data representing less than optimal conditions. First, it is important to learn from mistakes. The precise conditions that caused poor performance can be identified and steps can be taken to rectify the situation in the future. Secondly, by incorporating negative data with the positive data it increases the number of observations in the training set. As a result, more robust classifiers can be constructed. Finally, by analyzing both good and poor performance it is possible for the analyst to uncover the predictive structure of the problem. This means that the relationship between the variables that cause negative (poor) performance can be discovered and measures to assure positive (good) performance can then be taken.

An Application of the Three Step Method to JIT Manufacturing

In Just-In-Time (JIT) manufacturing, the kanban is a visual cue that is used to signal the replenishment of goods at each stage in the production process. The number of circulating kanbans is important to the effective operation of the JIT production system. Too many kanban cards produce excess work-in-progress inventory, while too few lead to production-floor disturbances. Moreover, the number of kanbans can significantly influence the load balance between processes, and the amount of orders needed to obtain supplies from subcontractors. Quite often JIT with kanbans is used in environments not meeting the conditions for optimal performance. These conditions may be unstable product demand, highly variable processing times, or highly variable vendor supply times. When these conditions exist a buffer of inventory is necessary to smooth production flow in the shop. The result is a factory "bloated" with work-in-progress inventory often characterized by a large number of kanbans at each

workcenter. Poor shop performance may also result from machine idle times, long lead times and output shortages which produce a factory that is "starved" for work-in-progress inventory. Such a system is typically characterized by a very small number of kanbans at each workcenter.

Step 1: Rule Induction Using Positive Training Data

A factory simulation model (Wray, Rakes and Rees, 1997) was used to generate 560 data points which were randomly divided into two data sets of 280 points each. One of the data set was used to generate the decision tree, while the other data set was used to validate the decision tree. All of the data reflects good factory performance (positive data). Eight dynamic factors were chosen to study over two periods of operation. The levels for the period t-1 factors would be the observed values for each factor during the previous period. The level of the period t factors would be a forecast for each factor in the next period of operation. The dependent variable is NKT (# of kanbans in period t). Table 1 shows the rules generated.

Table 1. Rules Generated from Positive Training Data

Rule #	If	Then
1.	Demand variability period t is low **and** Process variability period t is high	2 kanbans are needed
2.	Demand variability period t is low **and** Process variability period t is low **and** Vendor variability period t is high	2 kanbans are needed
3.	Demand variability period t is low **and** Process variability period t is low **and** Vendor variability period t is low	1 kanban is needed
4.	Demand variability period t is high **and** Leadtime period t-1 more than 56 **and** Vendor variability period t-1 is high	2 kanbans are needed
5.	Demand variability period t is high **and** Leadtime period t-1 more than 74 **and** Vendor variability period t-1 is low	2 kanbans are needed
6.	Demand variability period t is high **and** Leadtime period t-1 between 56 and 74 **and** Vendor variability period t-1 is low **and** Process variability period t is high	7 kanbans are needed
7.	Demand variability period t is high and Leadtime period t-1 between 56 and 74 **and** Vendor variability period t-1 is low **and** Process variability period t is low	3 kanbans are needed
8.	Demand variability period t is high **and** Leadtime period t-1 is 56 or less **and** Vendor variability period t is low **and** Number of kanbans period t-1 is more than 3	3 kanbans are needed
9.	Demand variability period t is high **and** Leadtime period t-1 is 56 or less **and** Vendor variability period t is low **and** Number of kanbans period t-1 is 3 or less **and** Process variability period t is high	4 kanbans are needed
10.	Demand variability period t is high **and** Leadtime period t-1 is 56 or less **and** Vendor variability period t is low **and** Number of kanbans period t-1 is 3 or less **an**d Process variability period t is low	5 kanbans are needed
11.	Demand variability period t is high **and** Leadtime period t-1 is 56 or less **and** Vendor variability period t is high **and** Number of kanbans period t-1 is more than 8	4 kanbans are needed
12.	Demand variability period t is high **and** Leadtime period t-1 is 56 or less **and** Vendor variability period t is high **and** Number of kanbans period t-1 is 8 or less **and** Process variability period t is high	4 kanbans are needed
13.	Demand variability period t is high **and** Leadtime period t-1 is 24 or less **and** Vendor variability period t is high **and** Number of kanbans period t-1 is 8 or less **and** Process variability period t is low	3 kanbans are needed
14.	Demand variability period t is high **and** Leadtime period t-1 is between 24 and 56 **and** Vendor variability period t is high **and** Number of kanbans period t-1 is 8 or less **and** Process variability period t is low	4 kanbans are needed

Step 2: Rule Induction Using Integrated Positive and Negative Training Data

The factory simulation model was used to generate 672 data points. Of these 672 data points, 560 reflect efficient factory conditions (positive data), 56 reflect starved factory conditions (negative data), and 56 reflect saturated factory conditions (negative data). The dependent variable from Step 1 (number of kanbans in period t) was made an independent variable in the new data set. The new dependent variable is factory performance (efficient, starved or saturated). Table 2 shows the rule set generated.

Table 2. Rules Generated from Integrated Positive and Negative Training Data

Rule #	If	Then you have
1.	The number of kanbans used is more than 4	a saturated factory
2.	The number of kanbans used is 1 **and** [either Process variability for period t or demand variability for period t or vendor variability for period t is high]	a starved factory
3.	The number of kanbans used is 1 **and** the Process variability for period t is low **and** the demand variability for period t is low **and** the vendor variability for period t is low	an efficient factory
4.	The number of kanbans used is 2 **and** the demand variability for period t is high	a starved factory
5.	The number of kanbans used is 2 **and** the demand variability for period t is low	an efficient factory
6.	The number of kanbans used is 3 or 4 **and** the demand variability for period t is high	an efficient factory
7.	The number of kanbans used is 3 or 4 **and** the demand variability for period t is low	a saturated factory

Step 3: Reconciliation of the Two Rule Sets

An examination of Rule #1 in Table 2 shows that whenever the number of kanbans in period t is greater than 4, poor factory performance (saturated factory) can be expected. In Table 1, Rule #6 and Rule #10 predict cases where the number of kanbans is greater than 4. Closer examination of the results indicates that Rule #6 and #10 on based on low number of cases and relatively low classification rates. This suggests that these rules are not robust, and should perhaps be eliminated from the ultimate rule set. An examination of Rule #2 and #3 in Table 2 indicates that 1 kanban is only efficient if the process variability **and** demand variability **and** vendor variability are low. Otherwise 1 kanban will result in a starved factory. This result is confirmed by Rule #3 in Table 1. As demonstrated the reconciliation process is used both to prune decision rules and to validate decision rules.

Conclusion

Finally, the results of this research show that inductive learning techniques can be applied to learn from negative data. Because traditional wisdom has dictated that negative data be eliminated from the training set, much valuable knowledge is lost. In a manufacturing environment it may be desirable to learn from an archived history of data that contains information that reflects less than optimal factory performance. CART is a technique that can accurately classify and predict factory performance based on shop factors, and can identify the important relationships between the shop factors that determine factory performance.

References

Breiman, L., Friedman, J., Olshen, R. and Stone, C.J., *Classification and Regression Trees*, Wadsworth, Belmont, CA, 1984.

Crawford, S.L., "Extensions to the CART algorithm", *International Journal of Man-Machine Studies*, **31**, (1989), 197-217.

Hall, D., Hansen, R. and Lang, D., "The negative information problem in mechanical diagnostics", *Journal of Engineering for Gas Turbines and Power*, 119, (1997), 370-377.

Holland, J.H., Holyoak, K.J., Nisbett, R.E. and Thagard, P.R., *Induction: Process of Inference, Learning, and Discovery*, The MIT Press, Cambridge, MA, 1986.

Quinlan, J.R., "Discovering rules by induction from large collections of examples", in Michie, D., Ed. *Expert Systems in the Micro Electronic Age*, The Edinburgh University Press, Edinburgh, 1979.

Triantaphyllou, E. and Soyster, A., "On the minimun number of logical clauses inferred from examples", *Computers & Operations Research*, 23, 8 (1996), 783-800.

Wray, B., Rakes, T., and Rees, L., "Identifying critical factors in a dynamic JIT environment using neural networks", *Journal of Intelligent Manufacturing*, 8 (1997) 83-96.