Multiple Algorithms for Fraud Detection

Richard Wheeler and Stuart Aitken

Artificial Intelligence Applications Institute,

The University of Edinburgh,

80 South Bridge,

Edinburgh EH1 1HN, Scotland.

Tel. (44) 131 650 2732

Fax. (44) 131 650 6513

Email {richardw,stuart}@aiai.ed.ac.uk

Abstract

This paper describes an application of Case-Based Reasoning to the problem of reducing the number of final-line fraud investigations in the credit approval process. The performance of a suite of algorithms which are applied in combination to determine a diagnosis from a set of retrieved cases is reported. An adaptive diagnosis algorithm combining several neighbourhood-based and probabilistic algorithms was found to have the best performance, and these results indicate that an adaptive solution can provide fraud filtering and case ordering functions for reducing the number of final-line fraud investigations necessary.

Keywords: fraud detection, case-based reasoning, adaptive algorithms

1 Introduction

Artificial intelligence techniques have been successfully applied to credit card fraud detection and credit scoring, and the field of AI as applied to the financial domain is both well-developed and well documented. As an emerging methodology, case-based reasoning (CBR) is now making a significant contribution to the task of fraud detection. CBR systems are able to learn from sample patterns of credit card use to classify new cases, and this approach also has the promise of being able to adapt to new patterns of fraud as they emerge. At the forefront of research in this field is the application of adaptive and hybrid learning systems to problems which previously were considered too dynamic, chaotic, or complex to accurately model and predict.

As applied to the financial domain, CBR systems have a number of advantages over other AI techniques as they:

- provide meaningful confidence and system accuracy measures,
- require little or no direct expert knowledge acquisition,
- are easily updated and maintained,
- articulate the reasoning behind the decision making clearly,
- are flexible and robust to missing or noisy data,
- may take into account the cost effectiveness ratio of investigating false positives and advise accordingly, and
- are easily integrated into varying database standards.

And the addition of adaptive CBR components may allow the system to:

- optimise the accuracy of classification by dynamically adjusting and updating weighting structures,
- use multiple algorithms to enhance final diagnostic accuracy, and
- better differentiate between types of irregularities and develop a diagnostically significant sense of abnormality which aids in the first-time detection of new irregularity types.

In this paper we describe the background of a complex fraud-finding task, and then describe the development of an adaptive proof-of-concept CBR system which is able to achieve very encouraging results on large, noisy real-world test sets. In specific, this paper addresses the problem of making a diagnostic decision given a set of near-matching cases. Finally, the results of the investigation are summarised and considered in the light of other work in the field.

2 Background

At the request of one of the UK's most successful fraud detection system software providers, AIAI undertook an investigation into methods of applying new AI technologies to increase the accuracy of the already highly advanced systems presently in use. While the firm's software presently reduces the number of necessary fraud investigations by several orders of magnitude, our investigation showed that utilising adaptive algorithms and fuzzy logic results in significant diagnostic improvement on the most difficult sub-section of cases.

The focus of our investigation was to reduce the number of applications referred for expert investigation after the existing detection systems had been utilised. These well-proven systems take advantage of many person years of expert knowledge elicitation and encoding, and are able to reduce the initial volume of applications by roughly 2500 times (400 referred from every million analysed). It was on a database consisting solely of these hardest cases that the CBR system attempted to make diagnostically significant decisions.

The source data comprised pairs of database records, the first of which was tagged as the application, the second as the evidence found by previous analysis suggesting fraud. Two files of data were provided: one set of nearly 2000 application-match pairs which were initially flagged as fraud but later cleared (the non-fraud set), and a second set of 175 application-match pairs which were judged to be fraudulent.

The application records consisted of applicant data (personal code, name of the lender, amount of loan, name and address of applicant, etc.), and employer data (type of business, time employed, etc.). Evidence provided followed a similar format, and the final test sets consisted of 584 cases of non-fraud and 96 cases of fraud. Each case consisted of an application and one or more evidence records both of which contained application and employer data.

Pre-processing was kept to a minimum, excluding only those fields which might be construed as being false indicators, such as database tags generated by the company's selection process. All other fields remained in their original state, and omissions formed a high percentage of total information encoded.

In order to capture more general patterns in application-match pairs *within* a case, the type of match that existed between fields was introduced into the case description. A small number of terms were defined to describe these matches, and this information was added into the cases after parsing. These general descriptions of matches were given simple descriptive labels: `exact- match`, `near-match`, `dissimilar` and added as a third component to each application-match pair. As such, the additional information was intended to act as a general fuzzy classifier of match fitness. The conjecture was that there are patterns of values for match types that might be exploited by an adaptive system. Similarity measures were assessed for all field types: strings, dates, addresses, numerical values, etc., and it was these final three-part sets: application, evidence, fuzzy match descriptor, that were presented to the proof-of-concept system for analysis. After all pre-processing, each case was described by 128 attributes.

3 Approach

Statistical investigations of the test sets suggested that the nature of the problem is inherently non-linear, noisy, contradictory, and not addressable using a simple similarity matrix and CBR decision system. This is unsurprising as the test sets were composed of the most difficult and intractable sub-set of the credit approval data, and as such did not cluster into identifiable fraud/non-fraud regions. However, highly localised phenomena and patterns appeared fairly common, suggesting that a hybrid or adaptive system within a CBR methodological structure might be able to focus upon and effectively exploit these characteristics.

The proof-of-concept system design has two essential decision-making components familiar to all CBR frameworks: retrieval and diagnosis. Retrieval utilises a weight matrix and nearest neighbour algorithm, while diagnosis utilises a suite of algorithms which analyse the data recalled by the retrieval mechanism as being significant. A learning mechanism was also implemented in the proof-of-concept system.

In this section we present the investigation of weighting matrix approaches, nearest neighbour strategies employed, and multi-algorithm final analysis, which is the focus of this work.

3.1 Weighting Matrix Approach

Case-based reasoning systems function by defining a set of features within a data or case base, and then generating a similarity score that represents the relationship between a previously seen case and the test case. Generally, this comparison is flat, that is, each field matching according to predefined operators (such as exact and fuzzy matching) adds one point to the total similarity score of the comparison. Of course, not all fields (or features) within a database are equally meaningful in divining a classification or sound decision, so a weighting matrix is often employed - a method by which a single feature's importance may be raised or lowered, giving certain features more diagnostic significance than others.

The first set of experiments performed were to experimentally assess the effect on total diagnostic accuracy of the raising and lowering of individual field weights. This was performed with the AIAI CBR Shell System[1] which supports the automatic polling of fields for sensitivity to goal finding and the stochastic hill-climbing of ever-fitter combinations of field weights. Disappointingly, these investigations only demonstrated that any simple relationships between field values and fraud occurrence had already been exploited by the rule-based filtering that had been applied to the data prior to our analysis of it. In consequence, a flat weighting structure was used in all subsequent testing.

3.2 Case Retrieval

Nearest neighbour matching is common to many CBR systems. Again using the basic exploratory facilities of AIAI's CBR test bed, a set of cases which were considered to be very similar, i.e., above a certain percentage of similarity, were retrieved. These retrieved cases are designated as being appropriate to include in the final diagnostic analysis for fraud.

This approach to nearest-neighbour recall should be differentiated from **k** nearest-neighbour method, where a fixed number of cases are recalled for consideration in the diagnosis or adaptation of a solution. Whereas **k** nearest-neighbour recall will always result in a constant number of returned cases for consideration (by expanding the neighbourhood to capture the desired number of nearby cases), *thresholded retrieval* retrieves cases from a specified neighbourhood. The threshold can be modified dynamically, and one case where we might change this parameter is where no cases are recalled.

[1] see: `http://www.aiai.ed.ac.uk/~richardw/cbrshell.html`

3.3 Diagnostic Algorithms

Applying the general principle of thresholded retrieval, a multi-algorithmic approach to final match analysis was developed as a result of the design and testing of a variety of single discrimination algorithms. It has been suggested [13] that no single algorithm may perform equally well on all search and classification tasks, and that an algorithm's improved performance in one learning situation may come at the expense of accuracy in another[9]. The idea of applying multiple algorithms within the CBR methodology is also gaining currency within the CBR community [1] and the techniques used in CBR continue to mature.

The algorithms described here further analyse the sub-set of significant cases retrieved by the matching system (with a flat weighting structure) and attempt to reach a final diagnosis of fraud or non-fraud. Each was designed and tested separately for performance before integration into the larger suite, and finally, resolution strategies were implemented to resolve conflicting diagnoses by the individual algorithms.

If all cases retrieved by a CBR system as being significant support a single correct analysis (here, fraud and non-fraud), then the algorithm required to perform the final analysis may be fairly simplistic; such as best match or simple percentage averaging (here referred to as probabilistic curve) methods. However, in highly dynamic, chaotic, and noisy environments it may be beneficial to apply or combine more complex algorithms to make a final decision that considers conflicting information and which is more accurate than simple similarity or summative analyses. These diagnostic operators may be combined dynamically within the final analysis system to increase accuracy and soundness of decision making.

3.3.1 Diagnosis Resolution Strategies

If more than one algorithm is asked to diagnose the set of cases retrieved for a unknown credit request, it is possible that the algorithms may disagree on the result, and resolution strategies were implemented to resolve the varying diagnoses into a single result. The present proof-of-concept system's diagnostic algorithms are able to assess their own confidence (as a function of cases recalled, position in the case base, etc.), and these confidence measures play an important role in enhancing overall system accuracy. Three resolution strategies were tested with varying results: sequential resolution, best guess, and combined confidence.

The sequential resolution strategy, the system's default, follows a simple sequential design that whichever algorithm finds sufficient evidence first (fires first) makes the decision. The algorithms were then ordered to analyse the final matches in this order: Density Selection, Negative Selection, Probabilistic Curve, Best Match, and Default. In this manner it may be seen that if the first algorithm is unable to make a firm decision, the algorithm following it will be given the task; finally falling to the default if all other algorithms fail to reach a definite conclusion.

The best guess resolution strategy simply chooses the algorithm that reports the highest confidence in its decision making, and disregards all others. While it is beyond the scope of this paper to detail the calculation of confidence measures for each algorithm (or the system as a whole) it may be noted that the factors which influence confidence include number and percentage of cases recalled, value of the threshold (size or extent of the neighbourhood), and overall system confidence derived from previous solutions and results.

The confidence-driven resolution approach is similar to best guess—differing in that it combines the differing algorithms' diagnoses according to their confidence levels into a combined recommendation. While the best guess strategy will report a final analysis solely based on the most confident algorithm, this approach will report a combined analysis of all algorithms reporting.

At present, the system provides the best results with a shifting or adapting measure of algorithmic confidence (raising or lowering individual algorithms' relative confidence according to their success rates in particular portions of the case base), and the sequential resolution strategy appeared to perform the best and maintain its stability - that is, did not significantly change its behaviour or accuracy across differing test sets. Whether this simple resolution strategy is inherently more accurate for this task or is simply better designed and tuned for this particular application is not known. It should also be noted that the CBR system was able to adapt the distance measure on the fly as cases were analysed to increase accuracy.

While the system may utilise as many as nine separate algorithms in making a final decision about fraud risk, four of these deserve further description as they form the final basis for high system accuracy.

3.3.2 Probabilistic Curve Algorithm

Built into the functionality of the test bed are several common CBR diagnostic algorithms that were tested on the data sets with predictable results.

The probabilistic curve algorithm classifies a case as either fraudulent or clear by expressing the dominant decision within the cases recalled as a percentage of the total. For example, if two cases are recalled, one supporting the classification of fraud with 60% similarity, and the other suggesting a decision of non-fraud with 40% similarity, the probabilistic curve algorithm will recommend a decision of fraud by 60% confidence/accuracy. This analysis is similar without regard to the number of cases recalled, as the final decision is decided by the Bayesian tallying of accuracy or similarity scores of similar match designations (fraud/non-fraud), and is highly sensitive to scope of the neighbourhood retrieved. Another algorithm (not detailed here) achieved similar results with a one-case one-vote approach, which, rather than combining confidence scores, simply assigns each similar recalled case result the value of one.

This algorithm may become the default decision-making device when other algorithms in the suite have failed to exploit stronger relationships. This algorithm results in 60/30 diagnostic split (60% recognition on non-fraud cases, 30% on fraud), but scores more highly on those cases which have not been exploited by other algorithms (70/35).

3.3.3 Best Match Algorithm

The best match algorithm, common to most CBRs, classifies a case as either fraudulent or clear by selecting the result of the closest matching case. As expected, this algorithm is highly sensitive to case base density and population, and generally chooses the dominant overall type (non-fraud) suggesting that fraud cases are not tightly clustered near to each other. Prior statistical analysis had already suggested this relationship within the data.

The best match algorithm results in a 90/10 diagnostic split (90% recognition on non-fraud cases, 10% on fraud), but again, curiously scores more highly on those cases which have not been exploited by other algorithms in the suite (60/35).

3.3.4 Negative Selection Algorithm

The negative selection algorithm relies upon the exploitation of highly regional relationships within the data that may not be valid at other points within the case base. Following the axiom that one bad apple may spoil the barrel, it functions by recalling all cases over the similarity threshold (all those within the neighbourhood), and if a user-defined or machine-derived number (or percentage) of cases of fraud are found, the application is automatically tagged as fraud for further investigation. The algorithm assumes that fraud cases form difficult to define clusters, and that if a new case falls within a certain radius of a fraud case then that case is more likely to be fraudulent as well, within a certain degree of probability for the case base considered. However, we made no assumption that fraud clusters taken together form a distinct region, only that they are more likely to occur nearer each other than the standard distribution within the larger case base.

While anecdotal evidence in the domain abounds in favour of this algorithm's guiding principle, we believe that the distribution of fraud cases near each other in lightly populated regions or pockets of non-fraud space account for the algorithms heightened accuracy in certain regions of the case base.

The negative selection algorithm results in a 70/55 diagnostic split (70% recognition on non-fraud cases, 55% on fraud), and the algorithm may only reach a negative decision if sufficient evidence for fraud is not found, as default algorithms in the suite are given priority over final decision making.

3.3.5 Density Selection Algorithm

The density selection algorithm uses the natural tendency of the case recall method (threshold or fixed-neighbourhood nearest neighbour) to return implicit information about case density at a particular point in the case base using this sub-index as the basis for the probability of one classification over another. This algorithm exploits the assumption that within certain portions of the case base (but perhaps not others) the number of cases (both fraud and non-fraud) recalled using a thresholding mechanism (one in which the recall neighbourhood is fixed to density and not to the number or percentage of cases recalled) may serve in the absence of other indicators as an significant index for fraud risk. In short, that outlying cases in certain portions of the case base are very likely to be fraudulent.

While this algorithm is very sensitive to the population size and spread within the case base, it has proven fairly stable to scaling and re-population as the case base expands. This algorithm results in a 70/40 diagnostic split (70% recognition on non-fraud cases, 40% on fraud), and takes precedence over other algorithms when its certainty (here calculated as a significant under or overabundance of non-fraud cases) is very high.

3.3.6 Default Goal

In the absence of strong confidence in one decision or another, the proof-of-concept system described here may also tag outlying or unusual cases arbitrarily as either fraud or non-fraud. The use of a simple default has been shown to slightly increase over-all accuracy when used in concert with a properly trained

INSERT FIGURE 1 HERE

Figure 1: Comparison of Algorithms

system, and may be used with smaller recall neighbourhoods to cause the system to be additionally critical of suspected fraud cases.

4 Results

System performance is difficult to measure due to sensitivities to case and testing base sizes, and distributions of fraud and non-fraud cases within each. However, the principles that guide the learning process have proven to be robust to larger sample sizes. The results in Figure 1 show a comparison of the performance of each algorithm. These results represent an average performance sample for the proof-of-concept with a flat weighting structure and without using learning.

Figure 1 shows that each of the four single algorithms has very different performance characteristics. (Note that the multi-algorithm system used negative selection, adaptive thresholding, density selection, probabilistic curve, best match, and fraud as the default goal). Through experimentation, we noted that the best match algorithm has very high accuracy in regions of high case density and consistency, while the density selection algorithm seems to perform best in areas of very high or very low case population. The negative selection algorithm appears to excel in diagnosing borderline or grey area cases. It should be remembered that the nature of the problem of fraud detection is a trade off between investigating (or identifying) too many non-fraud cases versus losing too many cases of genuine fraud. While ideal performance would be 100% recognition on both fraud and non-fraud cases, standard algorithms often only achieve a ratio of 60/30. When combined using even a simple diagnostic resolution strategy, the multi-algorithmic approach may be seen to outperform the sum of its parts (80% fraud, 52% non-fraud).

Due to the adaptive nature of the diagnostic algorithms, the proof-of-concept is very sensitive to small changes in the threshold (case matching neighbourhood), weights, and search method employed. Larger case and testing bases and the combination of learning with the multi-algorithm approach should result in more stable and accurate performance.

5 Related Work

The use of case-based reasoning both as an inference engine and as the framework for hybrid fraud-detection systems is well documented (see [2] for an authoritative example). While the broader and more fundamental ideas and applications within the field of CBR are well treated in a number of recent works [5, 6, 10], the following overview of the application domain provides useful points of comparison of CBR performance in similar applications to the fraud detection task addressed here.

While only 15 cases of actual fraud might be found in 1,000,000 applications or transactions, these cases may account for up to 10% of the revenue lost by a bank or lending agency, and in some cases may seriously effect profitability. Early detection is plainly the key to curbing lost revenues to fraud, and the nature and scope of the financial domain (the volume of applications, scope of purchases and prevalence of credit cards) dictates that expert review of more than a small minority of cases is impractical. The task of reducing the number of cases tagged for expert investigation is one that requires fine-grain analysis and pattern recognition on very large volumes of data. Fraud detection tasks in the financial domain fall into two categories: credit card fraud, and application fraud, although for the purposes of this investigation similarities between the two categories lend themselves to a single approach to the topic.

The decision to authorise or deny transactions made to a credit account by a centrally located controlling database system is one of the foundations of modern banking, and significant precedent exists for a set of rules which have been derived by experts as strong indicators of fraud. These include purchases out of the normal geographical region, large and unusual cash withdrawals, and other common indicators of malfeasance. Most, if not all, modern credit accounts which contact a central branch for approval use some form of expert rules to prevent fraudulent card use. More recently, hybrid systems have begun to emerge that utilise more advanced AI techniques and methodologies to enhance fraud detection accuracy. While the most common advance has been the use of neural networks (NNs) [4, 7] to learn and predict purchase and usage patterns and detect fraudulent uses through training, a number of case based reasoning systems and hybrids have been developed and deployed with excellent results. We provide here a brief description of several such systems that are typical of CBR development within this domain.

Hybrid approaches, or multiple-technique approaches within a common CBR framework, are becoming increasingly common. One study found that a CBR/NN system which divides the task of fraud detection into two separate components was more effective than either tactic on its own[8]. In this case, a neural net learns patterns of use and misuse of credit cards while a CBR looks for best matches in the case base. The case base contained information such as transaction dates and amounts, theft date, place used, type of transaction, and type of shop or company seeking approval. The combined CBR/NN system reported a classification accuracy of 89% on a case base of 1606 cases. The system was 92% accurate in its classification of those transactions to be denied authorisation (1479 cases), and was 50% accurate in granting authorisation (127 cases) [8]. The total accuracy was comparable with that of human specialists who are expected to achieve 90% accuracy.

Another recent application of CBR to the task of fraud detection was designed to grade loan applications according to credit-worthiness [11, 12]. A case base of 600 cases was constructed from the balance sheets of customers of a Swiss bank. The CBR was able to correctly classify 48.6% of the solvent companies as solvent and 93.6% of insolvent companies as insolvent[2]. The CBR was then optimised for decision accuracy and also for decision cost. The cost associated with incorrectly classifying insolvent companies is based on the loss of credit volume (i.e. the money lent), while the cost of incorrectly classifying solvent companies (and hence not authorising the loan) is the loss of interest on the loan. The cost of misclassifying a bad customer was ten times that of misclassifying a good one. It was shown that the decision cost was reduced by the learning cycle which optimised for decision accuracy, and was then further reduced by learning accounting for cost. The value of this adaptive approach to CBR fine-tuning is becoming increasingly apparent.

Similarly, the application of CBR to car insurance premium estimation has been reported with similar results. In one system, called RICAD, the case base was derived from the database of an Australian insurance company which contained 2 million entries [3]. This system calculated the risk cost of an application based on the average cost of claims made on similar policies. The risk cost therefore depends on the attributes that characterise cases and the weight given to each attribute. Each case has 30 attributes, including, e.g. post-code and vehicle code. The attribute driver-age had the highest weight value, and weights and risk factors were derived from insurance experts and statistical analyses of the data. Qualitative attributes such as make of car and post-code were organised taxonomically or

[2] It is also reported that 36.4% of solvent companies and 27.4% of insolvent companies are incorrectly classified - see Wilke *et al.* [11, 12] for details.

spatially, and a measure of distance was defined on the spatial knowledge structure. The RICAD system was able to identify the indexes which should be given high weights by learning, and the system was capable of updating the weighting structure as new policies are issued and claims processed.

The classification accuracies reported in the credit authorisation applications are comparable with that reported here. A second common feature is the interrelation of decision accuracy and decision cost: Classification systems operating in these domains have different accuracies for false positive/false negative decisions, and the costs of these errors can be very different. An important advantage of CBR techniques is the potential to use learning to improve decision making, taking costs into account.

6 Conclusions

Investigations into the financial data provided has proven that, though highly chaotic, it has properties that allow multi-algorithmic and adaptive CBR techniques to be used for fraud classification and filtering. The data set could not be partitioned into fraud and non-fraud regions. Instead, the occurrence and distribution of fraud cases in the neighbourhood of an unknown application was observed to be diagnostically significant, and these relationships were effectively exploited by a multi-algorithmic proof-of-concept system. Neighbourhood-based and probabilistic algorithms have been shown to be appropriate techniques for classification, and may be further enhanced using additional diagnostic algorithms for decision making in borderline cases, and for calculating confidence and relative risk measures.

While more accurate performance metrics and more thorough testing is required to appropriately quantify peak precision, the initial testing results of 80% non-fraud and 52% fraud recognition strongly suggest that a multi-algorithmic CBR will be capable of high accuracy rates. A comparison with related work shows that CBR techniques can achieve similar performance in comparable problem areas.

We believe that these results are very promising and supportive of a multi-algorithmic approach to classifying and assessing large, noisy data sets, and future work will focus upon testing the algorithms and resolution strategies on similarly complex data sets from other real-world domains.

Acknowledgements

Richard Wheeler would like to thank Jim Howe, Ann Macintosh, and everyone at AIAI for their support. Special thanks go to Peter Ross.

References

[1] Aha, D. See `http://www.aic.nrl.navy.mil/~aha/` for more information.

[2] Curet,O. and Jackson, M. Tackling cognitive biases in the detection of top management fraud with the use of case-based reasoning. *Proceedings of Expert Systems '95* ed. Macintosh, A. and Cooper, C. pp.223-236, 1995.

[3] Daengedej, J., Lukose, D., Tsui, E., Beinat, P., and Prophet, L. Dynamically creating indeces for two million cases: A real world problem. *Advances in Case-Based Reasoning, Proceedings of the Third European Workshop EWCBR-96* (LNCS 1168) Smith, I. and Faltings, B. (eds.), Springer, 1996, pp. 105–119.

[4] Gately, E. *Neural Networks for Financial Forecasting*, John Wiley and Sons, 1995.

[5] Kolodner, J. *Case-based Reasoning*, Morgan Kaufmann Publishers, 1993.

[6] Leake, D. *Case-Based Reasoning Experiences, Lessons, Future Directions*, AAAI/MIT Press, 1996.

[7] Masters, T. *Neural, Novel and Hybrid Algorithms for Time Series Prediction*, John Wiley and Sons, 1995.

[8] Reategui, E.B. and Campbell, J. A Classification System For Credit Card Transactions, *Advances in Case-Based Reasoning, Proceedings of the Second European Workshop EWCBR-94* (LNCS 984) Haton, J-P., Keane, M., and Manago, M. (eds.), Springer, 1994, pp. 280–291.

[9] Schaffer, C. A Conservation Law For Generalized Performance See `http://wwwcs.hunter.cuny.edu/faculty/schaffer/papers/list.html` for more information.

[10] Watson, I. *Applying Case-based Reasoning: Techniques for Enterprise Systems*, Morgan Kaufmann, San Francisco, 1997.

[11] Wilke, W., Bergmann, R. Considering Decision Cost During Learning Feature of Feature Weights *Advances in Case-Based Reasoning, Proceedings of the Third European Workshop EWCBR-96* (LNCS 1168) Smith, I. and Faltings, B. (eds.), Springer, 1996, pp. 460–472.

[12] Wilke, W., Bergmann, R., and Althoff, K-D. Fallbasiertes Schliessen in der Kreditwurdigkeitsprufung. *KI-Themenheft: KI-Methoden in der Finanzwirtschaft 4/96*.

[13] Wolpert, D. H. *On Overfitting Avoidance as Bias.* Technical Report SFI-TR-92-03-5001, Sante Fe Institute, 1992