EXPERIENCES WITH NEURAL NETWORK APPLICATIONS IN AGRICULTURE: TESTING EXPECTATIONS

F. Verdenius, A.J.M. Timmermans

ATO-DLO, Postbus 17, 6700 AA Wageningen, The Netherlands; phone: (+31) 317 47 50 00; fax: (+31) 317 41 22 60 e-mail: {F.Verdenius, A.J.M.Timmermans}@ato.dlo.nl

Abstract: Neural networks have been reported as promising for application in noisy, non-linear domains where human experts outperform many artificial intelligence solutions. The paper tests these expectations. Four NN applications are discussed in the light of three general purpose process models for application of learning techniques. The evaluation shows that, in spite of several practical problems, neural networks are a valuable tool for practical applications.

Keywords: Neural Networks, Classification, Regression, Agricultural Application

1. INTRODUCTION

In this paper the expectations about neural networks (NN) application are tested by discussing four projects performed at ATO-DLO. In section 2 we discuss relevant aspects for NN applications. To enable evaluation, three general process models for applying learning techniques are discussed as a reference in section 3. Section 4 discusses the four projects in relation to these references. Section 5 abstracts from the actual projects, evaluating NN application in agricultural domains, and section 6 concludes.

2. CONTEXT

2.1 Neural Networks

Neural networks are computational techniques that function in analogy with the human brain. In practice, applications of neural network are almost exclusively applying one technique: feed-forward multi-layer perceptrons, with (variants of) error back propagation (BPNN, Rumelhart et al., 1986) as the training algorithm. This NN technique is able to represent complex functions and is relatively easy to operate. The almost exclusive application of BPNN was concluded in a re-

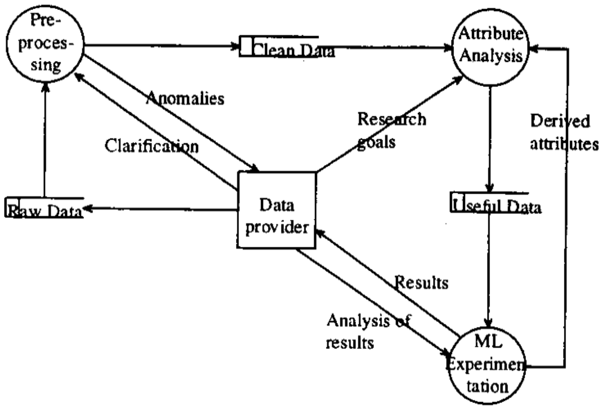

Figure 2 Process Model for Machine Learning Application, according to Garner et al. 1995

cent survey of learning applications (Verdenius 1995a).

Brodley & Smyth (1995) describe learning techniques by three aspects: the *model representation*, the *estimation criterion* and the *search method.* In NN the model representation is a non-linear mapping. It is realised with layers of activation nodes, connected by weighed links, and interacting via non-linear transfer functions. This representation offers a highly distributed, numerical representation of the knowledge of the network. It combines adequacy and efficiency of representation with a limited access for human understanding. This contrasts with symbolic learning techniques (e.g. C4.5; Quinlan 1993) where knowledge is codified explicitly, enabling humans to understand and reason about it. The estimation criterion is a likelihood-based objective function, based on cross validation or other performance measures. The crucial part for learning techniques is the training or learning algorithm, implementing the search method for optimal representation and fit to data. For NN the training algorithm defines how the weights between connections should be adapted to improve the performance characteristic of the network for a specific data-set.

2.2 *Application Modes*

If we understand *application of NN* as the process of solving a specific problem by using NN, two modes for applying NN can be distinguished:

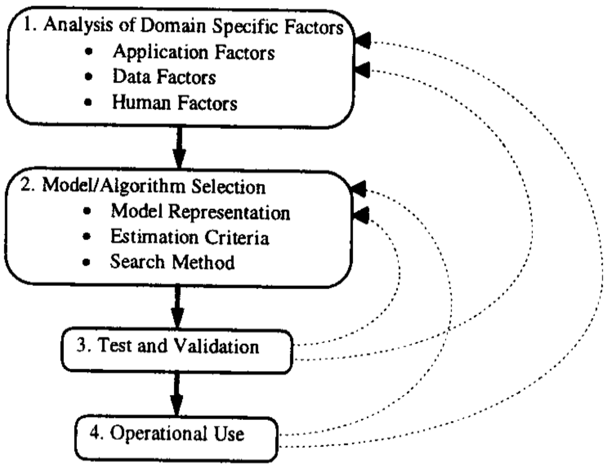

Figure 1 The application development process for machine learning **techniques according** to Brodley & Smyth (1995)

- Data Modelling (or: Data Analysis): the result of the analysis is a model, insight in the importance of data attributes, a prediction/classification for a data-set and information on the quality of that result. One conclusion of the survey on learning techniques was that many projects apply learning techniques (including NN) for these purposes.
- Task Implementation (or: Knowledge Acquisition): the NN contains knowledge to perform a problem solving (sub)task. It is the explicit goal to deliver a solution that can be used at any time the problem owner needs it. A Data Modelling phase will often be part of a Task Implementation.

2.3 *Functionality*

BPNN are used for two functions at performance time:

- **Classification**: Selecting class C_P to which input pattern P belongs; $C_P \in \{c1..cn\}$, with n finite.
- **Regression**: Establishing value x for an input pattern P, such that $x = f_{BPNN}(P)$.

Specific variants extend this functionality towards temporal dependencies, e.g. time series modelling; other networks also offer clustering functionality. Within these tasks, a finer subdivision may be applied, based on the pragmatic role the network plays. Such subdivisions include categories such as discrimination, pattern recognition, interpolation, process modelling and data compression.

2.4 *Evaluating Alternative Techniques*

Most NN applications use feed forward networks trained with back propagation. This is only one of many NN techniques that are able to realise the required functionality. If the view is broadened to other than neural techniques, the number of alternatives is even more impressive. An application designer has to select a technique from all available alternatives. It is hard to prove for any existing real world problem, whether a technique will perform well. Mostly, the suitability of a technique for a problem is assessed by iterative testing. For NN this testing leads to a parameter configuration for realising a good solution for the task. Selection of alternatives is based on the best performance, according to the estimation criterion.

There are process models that explicitly indicate the technique selection problem as a step in their cycle. However these models do not offer support in making this selection, leaving experimentation as the ultimate criterion.

3. PROCESS MODELS

Can NN application development be structured? Several authors describe process models for application of learning techniques. These process models aim at application of inductive learning techniques in general or at application of one specific technique. In this section two general purpose models and one model for neural networks are discussed.

3.1 *The WEKA Approach*

WEKA is a workbench containing several Machine Learning (ML) techniques. This workbench has been used for data modelling in agricultural domains (Garner et al., 1995). To structure the application of the workbench to real world domains a process model was developed. Figure 2 contains the relevant aspects of the WEKA approach:

- Need for data pre-processing: data-sets are never complete, consistent and sufficient. Data pre-processing solves data failures, attribute dependencies, missing values, inconsistencies, etc.
- Attribute analysis: feature selection to determine attributes that contribute to good classification,
- ML experimentation: selected data is processed with several ML techniques, which are compared by performance accuracy in an experimental cycle.

Alternative techniques (configurations) are selected through brute force experimentation. The WEKA process model supports data modelling. Provisions for task implementation are not foreseen. WEKA is suited for classification tasks. This however is not a principal limitation of the process model. WEKA is one of several tools supporting similar functionality. Some of these tools incorporate NN functionality. Specific NN workbenches are also available.

3.2 *The Process of ML Application*

Brodley & Smyth (1995) present a general model for application of ML algorithms. The approach is subdivided in 4 phases (Figure 1), starting with the problem analysis and ending with the operational use. In the second step, potentially successful techniques are selected on the basis of a match between problem- and technique characteristics. The third step contains the training and validation phase.

The process model stresses the importance of pragmatic non-technical aspects in delivering in successful applications. Unfortunately the authors have not indicated how these aspects are evaluated during application development.

The main difference between the WEKA approach and this approach is that the latter explicitly include domain-specific factors in the design process. Including such factors facilitates the ordering of techniques on the basis of pragmatic arguments.

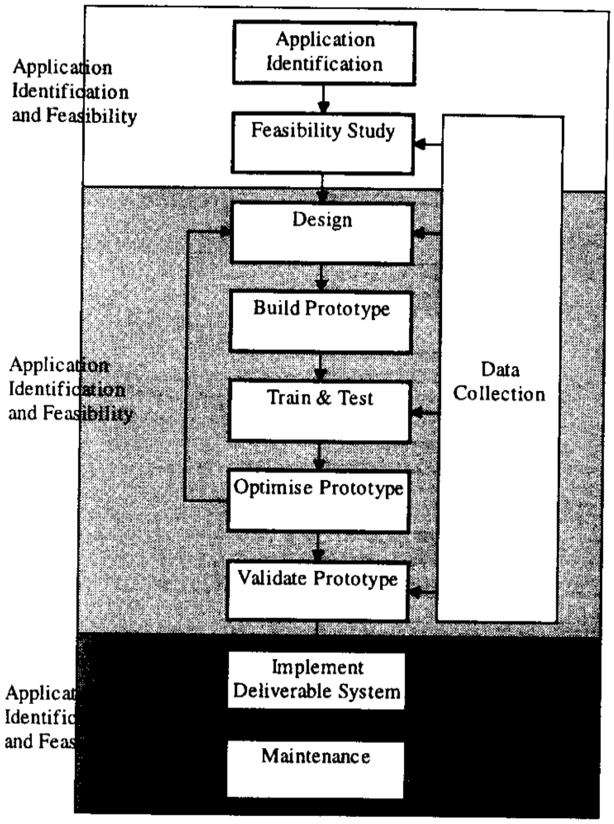

Figure 3 Process **Model** for NN application development according to DTI (1994)

3.3 DTI Approach for NN Application

The last process model is NN specific, developed by Touche Ross Management Consultants (DTI 1994). The process model contains an activity model (Figure 3). The first two phases indicate that it is a tool in a technology pull market. For each activity a set of sub-steps is given, together with guidelines on how to perform the steps. These guidelines vary in nature from pointers to other techniques and checklists to heuristics on the basis of experience.

The focus is on managing the application development process from a business point of view. It presents the development process as a waterfall approach, with minimal references to iterative cycles.

3.4 Discussion of Approaches

The presented approaches emphasise different aspects of the application of learning/NN techniques. The WEKA approach emphasises the analytical process and the interaction with the data provider and domain expert. This approach is directed towards ana-

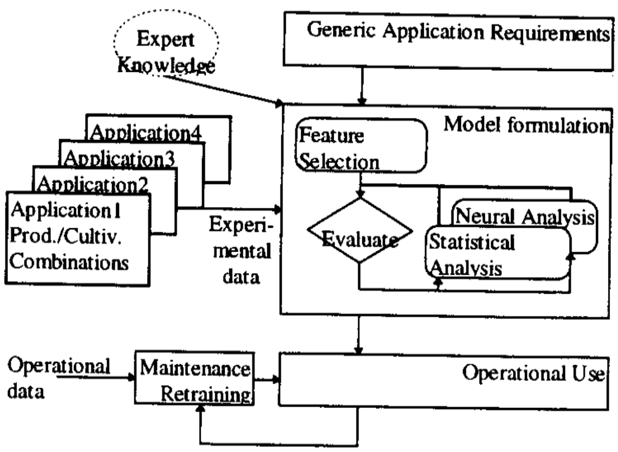

Figure 4 Schematic process model for pot plant classification **application**. This process model has emerged as a result of experiences over several projects.

lytical and experimental situations, where redesign of experiments remains possible. As a result, a formulation of the analysis problem is derived during the analysis process.

The approach of Brodley & Smyth accents the importance of problem analysis. It specifies factors to notice and structures the process for the realisation of a well-functioning application. The approach is more dedicated to the design of working solutions for real world problems.

The DTI approach is more generic in that respect that it resembles general purpose software life cycles such as SDM or KADS. Moreover it is more dedicated towards a manageable process of application.

The first two approaches aim at the development of a learning module; the latter concerns the development of a working solution for a given problem.

4. FOUR NN APPLICATIONS

Four NN application projects are described, two task implementations and two data modelling projects. The first system is currently commercially exploited; the second system resulted in prototype software. The last project is still under development.

4.1 Pot plant Sorting

Project Background: A flexible grading system for pot plants based on colour image processing has been developed (Timmermans and Hulzebosch, 1996). Pot plant growers use the system for on-line sorting of plants and seedlings, aiming at cost price reduction and objective quality sorting. Grading ornamentals is a human expert task. These experts capture grading norms for large numbers of situation. Norms differ per cultivar, client or season. In practice, inconsistency of expertise and inter-expert disagreement are observed.

Project Goals: The project aimed at a solution that can be easily adapted to specific product/cultivar requirements. This software based solution should be incorporated in a sorting system.

Project Approach: A software platform based on feature selection and learning by example offers a cost effective solution for these problems. Flexibility is realised by statistical and neural classifiers that can be easily adjusted for new products. The approach to realise a new implementation of an operational system is displayed in Figure 4. First application requirements are defined, e.g. the set of product/cultivar combinations to be classified, response time and classification accuracy.

The development of the sorting functionality is composed of two steps. First the set of relevant plant features to be calculated from the colour images is determined. These features are selected from a set containing over 35 features, e.g. surface area, height, width, flower area, flower position, colour and a number of shape and symmetry descriptors. For computational reasons and to improve the robustness and visualisation possibility, a limited feature set is used for classification. Normally between 3 and 7 features are selected. Second the product specific classifier is developed. In the product classification step a pattern recognition technique (statistical or neural based) is used to calculate discriminating functions for the quality groups in the multi-dimensional feature space.

After system delivery the client can maintain the system by retraining with operational data, adapting the system to developments in the domain.

4.2 Product Treatment Support System

Project Background: The Product Treatment Support System (PTSS; Verdenius 1995b) designs a treatment recipe for treating an agricultural product. Treatment in this respect means inducing a pre-specified (physiological) development in a pre-specified time-frame. A prescription of treatment conditions is called a treatment recipe. Currently human experts design treatment recipes. Recipes are based on standard recipes, that do not account for the variances in initial batch state.

Project Goals: A need for reactive recipe planning triggered the development of the PTSS. For the PTSS the main aspects of the treatment process are:

- State assessment, to determine the batch state before and during the treatment. The expert is supported in selecting products for state assessment by selection rules derived by C4.5 (Quinlan, 1993).
- Treatment Recipe Specification, to specify the global requirement a recipe has to fulfil to deliver the requested product quality at the end of the treatment. This function is realised by NN.
- Recipe Design, to derive, from the specification, a treatment recipe that fulfils the requirement without violating basic product constraints.

Project Approach: Recipe specifications are derived from batch data such as origin, transport history, maturity at harvest and measurement/assessment of the quality at start of the treatment. The latter includes attributes like colour and texture components.

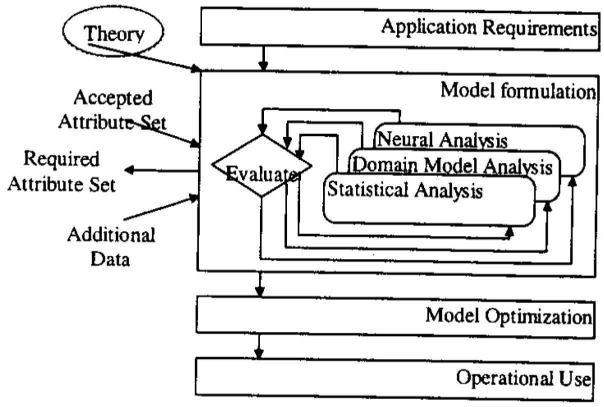

Figure 5 Process model for the PTSS neural network module.

The project approach is listed in Figure 5. During the definition of application requirements a model for assessing treatment requirements had to be derived. Data analysis initially focused on the attributes as used in practice. Outcome of the analysis process, amongst others, where directives for product research, resulting in additional attributes. On the augmented data-set regression analysis produced a fit $R^2 = 0.57$. The fundamental understanding of the relevant processes was not well enough developed to guide domain model based analysis (e.g. non-linear regression) to improve that fit. This, together with a more performance based interest of the project, triggered the analysis on the basis of NN, resulting in a model with a high quality ($R^2 = 0.9861$).

4.3 Cucumber Colour Development

Project Background: A quality parameter of cucumbers is the colour. When stored, enzymatic degradation of chlorophyll causes quality decay. As a result the colour changes from green to yellow. A research project tested the hypothesis that the duration of this decay (from an initial value to a specific level) correlates with the efficiency of the chlorophyll system. The efficiency of the chlorophyll system is measured with a Pulse Amplitude Modulated Fluorometer.

Project Goals: Goal of the project was to validate a model for the enzymatic chlorophyll decay by showing the predictability of the storage period on the basis of chlorophyll fluorescence measurements. Experimental data was fit to this model. The fit of the enzymatic model was not perfect (Pearson correlation coefficient $R_{xy} = 0.78$). This lead to the conclusion that either the de-greening process is not a result of enzymatic degradation of the chlorophyll system or that the model was incorrect.

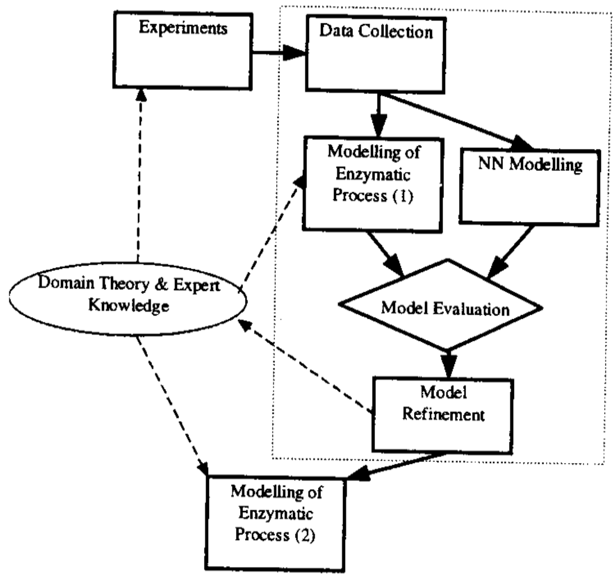

Figure 6 Process model for analysis of cucumber colour development.

Project Approach: Figure 6 illustrates the analysis process. Experiments are designed on the basis of the enzymatic model. After initial pre-processing of the data, the enzymatic model is fit to the data. Because of unsatisfactory results a NN model was developed for the same data-sets. This model resulted in a better fit ($R_{xy} = 0.93$). The results of the analysis can both be seen as a confirmation of the enzymatic model (as the data suffices for modelling the process) and as a denial of the model itself (as the current model formulation of the model does not satisfy the real behaviour). This has triggered a refinement step to improve on the current enzymatic model.

4.4 Post-processing Colour Prediction

Project Background: A main quality aspect of an agricultural raw produce for industrial food purposes is the post-processing colour. The post-processing colour depends on the raw material composition, which varies over time. Various components influence the colouring during processing. Currently raw produce quality is assessed by processing a

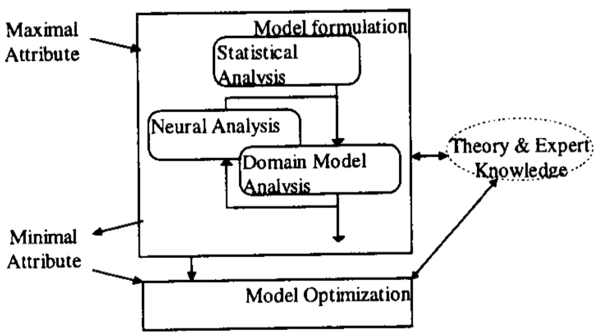

Figure 7 Schema of the analysis process for post-processing colour.

sample under standardised conditions and assess the post-processing colour. Descriptive models of the colouring process have a maximal statistical fit $R^2 = 0.82$ on training data. A full understanding of relevant mechanisms does not exist.

Project Goals: Goal of the project is to derive a model that can predict the post-processing colour from initial measurements. A sub-goal is to derive a minimal attribute set to measure.

Project Approach: Data was gathered from experimental process runs. For each experiment ca. 100 attributes were measured. The number of records that could be obtained was limited. As a result, reliable analysis on the basis of all attributes is not possible and best features have to be selected. Guided by both primary statistical analysis and expert guidelines, attribute subsets (10-14 attributes) have been composed that enable data analysis. Statistical modelling obtained the results presented above. NN resulted in a fit R^2 of 0.9955 on the training data. The difference is explained by the NN robustness against non-linearities in the data as well as by the ability of the NN to cover complex interactions. In an iterative process, NN are used to further reduce the attribute set. This will influence the process of theory formulation. In Figure 7 the process of analysis is depicted.

5. DISCUSSION

5.1 Process Model

The projects group according to application mode:

1. The data modelling projects follow process models resembling the WEKA approach. There is strong involvement of domain experts in setting up experiments and selecting attributes for analysis. Furthermore, the feedback loops confirm the WEKA process model. Finally, the incremental derivation of the domain/application requirements is an indication thereof.
2. The task performance differ from the data modelling projects in several aspects. They include phases to obtain domain or application requirements. These requirements refer to non-NN application aspects and human factors. These include experimental iteration for model development, however limited due to time constraints. Finally, the projects may also attend use and maintenance phases. Here, data aspects and human factors again play an important role.

A difference between the two groups is that for the task performance projects, NN is a last option, i.e. when other techniques, such as statistical- and domain modelling fail to meet quality requirements. Only then NN can be justified in practice. In the case of data modelling, the NN is also use as a benchmark a priori. In this case the NN sets a quality norm. Other techniques will be considered acceptable, if they outperform the NN results.

5.2 Alternative Techniques

An important issue in NN projects is the consideration of alternative techniques. In the WEKA approach alternative techniques are evaluated by brute force comparison by using available techniques in parallel. Though the Brodley & Smyth and DTI mention technique selection without providing guidelines on doing so. Both approaches advice comparative tests. DTI indicates that for specific tasks, BPNN is preferred. Brodley and Smyth aim at a general level and indicate no specific directives.

For pot plant sorting and the PTSS alternative techniques have been considered. For the pot plant sorting application linear discriminant functions (LDF) have been considered for classification. The current system offers the choice between LDF and NN. In the case of the PTSS multiple linear regression (MLR) was assessed as alternative. NN outperformed MLR, resulting in the choice of NN for implementation. Non-linear regression was rejected because an hypotheses model does not exist.

The process of technique selection (NN- , statistical - and learning techniques) is hard to formalise. Apart from performance aspects, which are described by the approaches in section 3, other more pragmatic criteria play a role. From the pot plant sorting application for instance we learn that performance is not the only reason to prefer a specific technique. Robustness, easiness to adjust and learn, verification and visualisation possibilities, education level of the end-user are important aspects to consider when taking the decision what classification technique to use. A disadvantage of applying NN classifiers is the optimisation process. There is no universal network structure that solves all possible classification problems. The construction of training sets seems more important for NN than for discriminant analysis, specifically in the situation that outliers may occur at performance time. When applying discriminant analysis the classification space can be easily visualised. The structure of a neural network remains a black box. For objects not covered by the product examples, classification results are unpredictable and not acceptable with a NN.

6. CONCLUSIONS

What is the influence of the agricultural domain on the approach of realising NN functionality. First, data is expensive or scarce. This requires special care in setting up experiments. Second, in classification problems class definitions may be unclear. This can be due to conflicting expertise, noise etc. Special attention deserves the problem of temporal variations in data (season, off-season norms, evolution in domains, cultivar innovation).

What problems remain to be solved to support practical application of NN? Given a specific NN technique, network design (nr. layers, nr. nodes, representations) and parameter optimisation are known problems. Solutions for such problems are not yet commonly available.

How do the experiences meet the expectations? In four projects NN improve the results obtained with other techniques. Some characteristics of NN deserve attention. A technical problem is the unruliness of the training and evaluation of training results. Specifically, it remains unclear whether NN-results can be improved by adapting learning or design parameters. Solutions for these problems seem not yet available. Other problems are more related to the domain. In real world applications pragmatic arguments may lead to the choice of alternative techniques, even if NN perform better; this can lead to the low profile application of NN. Furthermore data acquisition deserves specific attention. It is not always possible to obtain large data-sets of sufficient quality, requiring a good definition on beforehand. This is especially true for task implementation projects. All in all our experiences are positive. NN are a powerful tool in the modelling the complexity of product behaviour and in solving classification and prediction problems in agricultural domains.

ACKNOWLEDGEMENTS

We thank Rob Schouten, Arnold Braaksma, Clare Wilkinson, Olaf van Kooten en Paul van Eijck for their co-operation, comments and suggestions.

REFERENCES

Brodley, C.E. & Smyth, P. (1995), The Process of Applying Machine Leaning Algorithms, paper presented at the workshop on *Applying Machine Learning in Practice* at ICML-95

DTI (1994), Neural Computing, Learning Solutions, London,

Garner, S.R., Cunningham, S.J., Homes, G., Neville Manning, C.G. & Witten, I.H. (1995), Applying a machine Learning Workbench: Experiences with Agricultural Databases, paper presented at the workshop on *Applying Machine Learning in*

Practice at ICML-95

Quinlan, J.R. (1993), C4.5: Programs for Machine Learning, Morgan Kauffman, San Mateo (CA)

Rumelhart D.E., Hinton, G.E. & Williams, R.J. (1986), Learning Internal Representations by Error Backpropagation, in: D.E. Rumelhart, J.L. McClelland, Parallel Distributed Processing, Volume 1, pp. 318-362

Timmermans A.J.M., Hulzebosch, A.A. (1996), Computer vision system for on-line sorting of pot plants using an artificial neural network classifier, Computers and Electronics in Agriculture (in press)

Verdenius, F. (1995a), Applications of Inductive Learning Techniques: State of the Art, in: J.C. Bioch & Y.-H. Tan (eds.), Proceedings of the 7th NAIC, Rotterdam 1995, pp. 231-239

Verdenius, F. (1995b), Managing Product Inherent Variance During Treatment, in: Presented at the 2nd IFAC/IFIP/EurAgEng Workshop AI in Agriculture, Wageningen, May 29-31