ELSEVIER

European Journal of Operational Research 138 (2002) 191–211

EUROPEAN JOURNAL OF OPERATIONAL RESEARCH

www.elsevier.com/locate/dsw

Interfaces with Other Disciplines

Bayesian neural network learning for repeat purchase modelling in direct marketing

Bart Baesens [a], Stijn Viaene [a], Dirk Van den Poel [b,*], Jan Vanthienen [a], Guido Dedene [a]

[a] *K.U.Leuven, Department of Applied Economic Sciences, Naamsestraat 69, B-3000 Leuven, Belgium*
[b] *Ghent University, Department of Marketing, Hoveniersberg 24, B-9000 Ghent, Belgium*

Received 7 November 2000; accepted 14 March 2001

Abstract

We focus on purchase incidence modelling for a European direct mail company. Response models based on statistical and neural network techniques are contrasted. The evidence framework of MacKay is used as an example implementation of Bayesian neural network learning, a method that is fairly robust with respect to problems typically encountered when implementing neural networks. The automatic relevance determination (ARD) method, an integrated feature of this framework, allows us to assess the relative importance of the inputs. The basic response models use operationalisations of the traditionally discussed Recency, Frequency and Monetary (RFM) predictor categories. In a second experiment, the RFM response framework is enriched by the inclusion of other (non-RFM) customer profiling predictors. We contribute to the literature by providing experimental evidence that: (1) Bayesian neural networks offer a viable alternative for purchase incidence modelling; (2) a combined use of all three RFM predictor categories is advocated by the ARD method; (3) the inclusion of non-RFM variables allows to significantly augment the predictive power of the constructed RFM classifiers; (4) this rise is mainly attributed to the inclusion of customer/company interaction variables and a variable measuring whether a customer uses the credit facilities of the direct mailing company.

Keywords: Neural networks; Marketing; Bayesian learning; Response modelling; Input ranking

1. Introduction

It is well established in the literature that customer retention is at least as important as customer acquisition in the current context of competitive markets, not in the least for (direct) mail-order companies. Mail-order companies typically are in the business of sending out catalogues to a selected number of prospective buyers. The

* Corresponding author. Tel.: +32-9-264-8980; fax: +32-9-264-4279.

E-mail addresses: bart.baesens @ econ.kuleuven.ac.be (B. Baesens), stijn.viaene@ econ.kuleuven.ac.be (S. Viaene), dirk.vandenpoel@rug.ac.be (D. Van den Poel), jan.vanthienen@econ.kuleuven.ac.be (J. Vanthienen), guido.dedene@econ.kuleuven.ac.be (G. Dedene).

0377-2217/02/$ - see front matter
PII: S0377-2217(01)00129-1

selection of whom to include in the mailing list rests on an assessment of the individual's propensity to buy. The prospects or customers to be mailed are typically selected following the results of statistical models including behavioural, demographic and other customer profiling predictors in order to optimise the prospective buyer response rate. Commonly used target variables for these mailing response models are purchase incidence, purchase amount and interpurchase time. In this paper, we focus on the purchase incidence, i.e. the issue whether or not a purchase is made from any product category offered by the direct mail company.

Conceptually, the purchase incidence response modelling issue reduces to the general problem category of binary classification: repurchase or not. Among the traditional (statistical) techniques that have been widely used are logistic regression, linear and quadratic discriminant analysis models. However, their pre-determined functional form and restrictive (often unfounded) model assumptions limit their usefulness [4,58]. In this paper, we use neural networks (NNs) for response modelling. Their universal approximation property makes them a very interesting alternative for pattern recognition purposes. Unfortunately, many practical problems still remain when implementing NNs, e.g. *How to choose the appropriate number of hidden neurons? What is the impact of the initial weight choice? How to set the weight decay parameter? How to avoid the network from fitting noise in the training data?* These issues are often dealt with in an ad hoc way [3]. Nevertheless, they are crucial to the success of the NN implementation. A Bayesian learning paradigm has been suggested as a way to deal with these problems during NN training [4,33,34,42]. Here, all prior assumptions are made explicit and the weights and hyperparameters are determined by applying Bayes' theorem to map the prior assumptions into posterior knowledge after having observed the training data. In this paper, we use the evidence framework of MacKay as an example implementation of Bayesian learning [33–36]. An interesting additional feature of this framework is the automatic relevance determination (ARD) method which allows us to assess the relative importance of the various inputs by adding weight regularisation terms to the objective function. In this paper, it is shown that training NNs using the evidence framework (with the ARD extension) is an effective and viable alternative for the response modelling case at hand when compared to the three benchmark statistical techniques mentioned above.

The empirical study consists of two subexperiments. Initially, only standard Recency, Frequency and Monetary (RFM) predictor categories will underly the purchase incidence model. This choice is motivated by the fact that most previous research cites them as being most important and because they are internally available at very low cost [1,15,29]. It is shown for this case that, from a predictive performance perspective, Bayesian NNs are statistically superior when compared to logistic regression, linear and quadratic discriminant analysis classifiers. Predictive performance is quantified by means of the percentage correctly classified (PCC) and the area under the receiver operating characteristic curve (AUROC). The latter basically illustrates the behaviour of a classifier without regard to class distribution or error cost, so it effectively decouples classification performance from these factors [20,59,60]. The ARD method is used to shed light upon the relative importance of all variables operationalising the RFM response model. In a second experiment, the response model is extended with other potentially interesting customer profiling variables. It is illustrated that the Bayesian NNs still perform significantly better than the three statistical classifiers. Again, the relative importance of the inputs is assessed using the ARD method.

This paper is organised as follows. In Section 2 we provide a concise overview of response modelling issues in the context of direct marketing. Section 3 discusses the theoretical underpinnings of NNs for pattern recognition purposes. The Bayesian evidence framework for classification is presented in Section 4. Section 5 presents the ARD extension of the evidence framework. The design of the study, including data set description, experimental setup and used performance criteria are presented in Section 6. Results and discussion of

the basic and extended RFM experiment are covered in Sections 7 and 8.

2. Response modelling in direct marketing

For mail-order response modelling, several alternative problem formulations have been proposed based on the choice of the dependent variable. The first category is purchase incidence modelling [9]. In this problem formulation, the main question is whether a customer will purchase during the next mailing period, i.e. one tries to predict the purchase incidence within a fixed time interval (typically half a year). Other authors have investigated related problems dealing with both the purchase incidence and the amount of purchase in a joint model [32,64]. A third alternative perspective for response modelling is to model interpurchase time through survival analysis or (split-)hazard rate models which model whether a purchase takes place together with the duration of time until a purchase occurs [16,63].

This paper focuses on the first type of problem, i.e. purchase incidence modelling. More specifically, we consider the issue whether or not a purchase is made from any product category offered by the direct mail company. This choice is motivated by the fact that the majority of previous research in the direct marketing literature focuses on the purchase incidence problem [41,69]. Furthermore, this is exactly the setting that mail-order companies are typically confronted with. They have to decide whether or not a specific offering will be sent to a (potential) customer during a certain mailing period.

Cullinan is generally credited for identifying the three sets of variables most often used in response modelling: (R)ecency, (F)requency and (M)onetary [1,15,29]. Since then, the literature has accumulated so many uses of these three variable categories, that there is overwhelming evidence both from academically reviewed studies as well as from practitioners' experience that the RFM variables are an important set of predictors for modelling mail-order repeat purchasing. However, the beneficial effect of including other variables into the response model has also been investigated. In Table 1, we present a literature overview of the operationalisations of both the independent and dependent variable(s) in direct marketing response modelling studies. It shows that only few studies include non-RFM variables. Moreover, these studies typically include only one operationalisation per variable.

The substantive relevance of response modelling comes from the fact that an increase in response of only one percentage point can result in substantial profit increases, as the following real-life example of an actual mail-order company illustrates. Suppose that the mail-order company decides to mail to 75% of its current mailing list of 5 million customers, i.e. 3,750,000 mailings are sent out. Suppose that the overall response rate when mailing to all of their current customers is 10% during a particular mailing period, i.e. if everyone would be mailed, 500,000 orders would be placed. Suppose further that the average contribution per customer amounts to 100 Euro, which is the typical real-life situation of a large mail-order company. Table 2 compares the economics of several alternative response models. When no model is available, we can expect to obtain 75% of all potential responses (i.e. $0.75 \times 500{,}000 = 375{,}000$ responses) when 75% of 5 million people are mailed (i.e., 3.75 million mailings are sent out). The ideal model (at the specific mailing depth) is able to select the people from the mailing list in such a way that the 500,000 potential customers all receive a mailing, i.e. even though 25% of the mailing list is not mailed, not a single order is lost. Suppose further that the current response model used by the company, by mailing to 75% of their mailing list, allows to obtain 90% of the responses, i.e. even though 1,250,000 people on the list do not receive a mailing, only 10% of the 500,000 potential customers are excluded. This will result in 450,000 orders, which represents a substantial improvement over the 'null model' situation. If a better response model can be built, which achieves 91% of the responses instead of 90%, the contribution of this change will directly increase the contribution over the null model from 7.50 million Euro to 8 million Euro, i.e. by 500,000 Euro (1% of 10% of 5 million customers $\times$ 100 Euro average contribution).

Given a tendency of rising mailing costs and increasing competition, we can easily see an increasing importance for response modelling [25]. Improving the targeting of the offers may indeed counter these two challenges by lowering non-response. Moreover, from the perspective of the recipient of the (direct mail) messages, mail-order companies do not want to overload consumers with catalogues. The importance of response modelling to the mail-order industry is further illustrated by the fact that the issue of improving targeting was among the top three concerns with 73.5% of the cataloguers in the sample mentioned in [19].

In this study, we contribute to the literature by providing a thorough investigation into: (1) the suitability of Bayesian neural networks for repeat purchase modelling; (2) the predictive performance

Table 1
Literature review of response modelling papers

Reference	Independent variable						Dependent variable		
	R	F	M	Length of relationship	Other behavioural	Socio-demographic	Binary	Binary and amount	Binary and timing
Berger and Magliozzi [2]	X	X	X			X	X		
Bitran and Mondschein [5]	X	X	X					X	
Bult and Wittink [11]	X		X			X		X	
Bult [9]	X	X					X		
Bult [8]			X		X	X	X		
Bult et al. [10]	X	X	X	X	X	X	X		
Desarbo and Ramaswamy [18]							X		
Gönül and Shi [23]	X	X					X		
Kaslow [28]	X	X	X	X	X	X			X
Levin and Zahavi [32]	X	X	X		X	X	X	X	
Magliozzi and Berger [38]	X	X	X				X		
Magliozzi [37]	X	X	X				X		
Rao and Steckel [47]								X	
Thrasher [61]	X	X			X	X	X		
Van den Poel [62]	X	X	X	X		X	X		
Van der Scheer [64]			X			X		X	
Zahavi and Levin [69]	X		X		X	X	X		

Table 2
Economics resulting from performance differences among response models

Type of model	Mailing depth	No. of customers (million)	No. of mailings sent out (million)	No. of responses	Average contribution (Euro)	Total contribution (million Euro)	Additional contribution over 'No model' (million Euro)
Null model	75.00%	5.00	3.75	375,000	100.00	37.50	0.00
Ideal model	75.00%	5.00	3.75	500,000	100.00	50.00	12.50
90% Model	75.00%	5.00	3.75	450,000	100.00	45.00	7.50
91% Model	75.00%	5.00	3.75	455,000	100.00	45.50	8.00

of alternative operationalisations of RFM variables and their relative importance; (3) the issue whether other (non-RFM) variables add predictive power to the traditional RFM variables.

3. Neural networks for pattern recognition

Neural networks (NNs) have shown to be very promising supervised learning tools for modelling complex non-linear relationships [4,50,71]. NNs are designed to deal with both regression and classification tasks. This, especially in situations where one is confronted with a lack of domain knowledge. As universal approximators, they can significantly improve the predictive accuracy of an inference model compared to mappings that are linear in the input variables [26]. In what follows, the discussion will be limited to the binary classification problematic. Typical application areas include medical applications [39,44,53], business failure prediction [14,30,55,70] and customer credit scoring [17,22,45].

NNs are mathematical representations inspired by the functioning of the human brain. An NN is typically composed of an input layer, one or more hidden layers and an output layer, each consisting of several neurons (layer units). Each neuron processes its inputs and generates one output value which is transmitted to the neurons in the subsequent layer. In a multi-layer perceptron (MLP), all neurons and layers are arranged in a feedforward manner. For a binary classification problem one commonly opts for an MLP with one hidden layer and one output unit.

This neural network then performs the following non-linear function mapping:

$$y = f_2(w_2 f_1(w_1 x)), \tag{1}$$

where $x \in \mathbb{R}^n$ and $y \in \mathbb{R}$ is the MLP produced output. w_1 and w_2 are weight vectors of the hidden and output layer, respectively. The weight vectors w_1 and w_2 together make up the parameter vector $\mathbf{w}$, which needs to be estimated (learned) during a training process. f_1 and f_2 are termed transfer functions and essentially allow the network to perform complex non-linear function mappings.

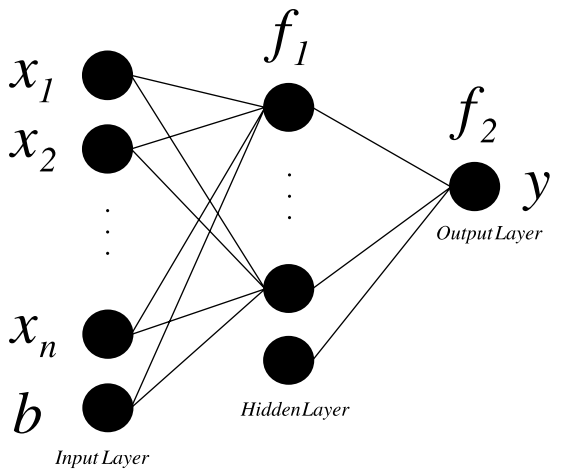

Fig. 1. A multi-layer perceptron with one hidden layer and one output unit.

An example of an MLP with one hidden layer and one output unit is presented in Fig. 1.

Given a training data set $D = \{x^{(m)}, t^{(m)} \mid m = 1, \ldots, N\}$, where $x^{(m)}$ is an n-dimensional input vector corresponding to a specific data instance m that is labelled by a target variable $t^{(m)}$, the weight vector $\mathbf{w}$ of the NN is randomly initialised and iteratively adjusted so as to minimise an objective function, typically the sum of squared errors (SSE)

$$E_D = \frac{1}{2}\sum_{m=1}^{N}(t^{(m)} - y^{(m)})^2. \tag{2}$$

The backpropagation algorithm originally proposed by Rumelhart et al. [52] is probably the best-known example of the above mechanics. It performs the optimisation by using repeated evaluation of the gradient of E_D and the chain rule of derivative calculus. Due to the problems of slow convergence and relative inefficiency of this algorithm, new and improved optimisation methods (e.g. Levenberg–Marquardt and quasi-Newton) have been suggested to deal with the latter. For an overview, see [4].

For a binary classification problematic it is convenient to use the logistic transfer function

$$f(z) = \frac{1}{1 + \exp(-z)} \tag{3}$$

as transfer function in the output layer (f_2), since its output is limited to a value within the range $[0,1]$. This allows the output $y^{(m)}$ of a neural network to be interpreted as a conditional probability of the form $p(t^{(m)}=1|x^{(m)})$ [4]. In that way, the neural network naturally produces a score per data instance, which allows the data instances to be ranked accordingly for scoring purposes (e.g. customer scoring). It has to be noticed that for classification purposes the SSE function E_D (see Eq. (2)) is no longer the most appropriate optimisation criterion because it was derived from maximum likelihood on the assumption of Gaussian distributed target data [4,7,56]. Since the target attribute is categorical in a classification context, this assumption is no longer valid. A more suitable objective function is the cross-entropy function which is based on the following rationale [4]. Suppose we have a binary classification problem for which we construct an NN with a single output representing the posterior probability $y^{(m)}=p(t^{(m)}=1|x^{(m)})$. The likelihood of observing $t^{(m)}\in\{0,1\}$ given $x^{(m)}$ is then given by

$$p(t^{(m)}|x^{(m)})=(y^{(m)})^{t^{(m)}}(1-y^{(m)})^{1-t^{(m)}}. \tag{4}$$

The likelihood of observing the training data set is then modelled as

$$\prod_m (y^{(m)})^{t^{(m)}}(1-y^{(m)})^{1-t^{(m)}}. \tag{5}$$

The cross-entropy error function G maximises this likelihood by minimising its negative logarithm:

$$G=-\sum_m \left\{t^{(m)}\ln(y^{(m)})+(1-t^{(m)})\ln(1-y^{(m)})\right\}. \tag{6}$$

It can easily be verified that this error function reaches its minimum when $y^{(m)}=t^{(m)}$ for all $m=1,\ldots,N$. Optimisation of G with respect to $\mathbf{w}$ may be carried out by using the optimisation algorithms mentioned in [4].

For decision purposes, the posterior probability estimates produced by the NN are used to classify the data instances into the appropriate (predefined) classes. This is done by choosing a threshold value in the scoring interval $[0,1]$. The optimal choice of this threshold value can be related to the probabilistic interpretation of the network outputs as follows. Suppose we have two classes, class 1 ($t^{(m)}=1$) and class 0 ($t^{(m)}=0$). As mentioned above, the output of the NN represents the estimated probability that a particular data instance m belongs to class 1 given its input vector $x^{(m)}$. The misclassification percentage is then minimised by assigning an instance $x^{(m)}$ to the class $c\in\{0,1\}$ having the largest posterior probability estimate $p(t^{(m)}=c|x^{(m)})$. This simply comes down to choosing a threshold value of 0.5. A data instance is assigned to class 1 if its output (posterior) probability exceeds this threshold and to class 0 otherwise. Notice that this reasoning is contingent on a situation in which equal misclassification costs are assigned to false positive and false negative predictions.

The ultimate goal of NN training, and eventually of every inference mechanism, is to produce a model which performs well on new, unseen test instances. If this is the case, we say that the network generalises well. To do so, we basically have to avoid the network from fitting the noise or idiosyncracies in the training data. This is most often realised by monitoring the error on a separate validation set during training of the network. When the error measure on the latter set starts to increase, training is stopped, thus effectively preventing the network from fitting the noise in the training data (early stopping). A superior alternative is to add a penalty term (weight regulariser) to the objective function as follows [4,58]:

$$F(\mathbf{w})=G+\alpha E_W, \tag{7}$$

whereby, typically

$$E_W=\frac{1}{2}\sum_i w_i^2 \tag{8}$$

with i running over all elements of the weight vector $\mathbf{w}$. This method for improving generalisation constrains the size of the network weights $\mathbf{w}$ and is referred to as regularisation. When the weights are kept small, the network response will be smooth. This decreases the tendency of the network to fit the noise in the training data.

The success of NNs with weight regularisation obviously depends strongly on finding appropriate values for the weight vector **w** and the hyperparameter α. In Section 4, we discuss the evidence framework of MacKay as our method of choice for training the NN weight vector **w** and setting the hyperparameter α [33–35].

4. The evidence framework

Bayesian learning essentially works by adapting prior probability distributions into posterior probability distributions guided by the training data [4,33–35,42]. Relying on probability distributions stresses the importance of capturing the inherent uncertainty while learning the true relationship from a finite data sample. In a Bayesian context, all implicit assumptions, i.e. prior knowledge encoded in the form of prior probability distributions, have to be made explicit and rules are provided for reasoning consistently given those assumptions. More specifically, in a Bayesian NN learning framework, the weights of the neural network are considered random variables and are characterised by a joint probability distribution. In this section, we restrict our attention to the evidence framework for Bayesian learning as introduced by MacKay in [33–35]. Other implementations of Bayesian learning have been presented in e.g. [12,42,68].

Let $p(\mathbf{w}|\alpha, H)$ be the prior probability distribution over the weight vector **w** given a neural network model H and the hyperparameter α. $p(\mathbf{w}|\alpha, H)$ expresses our initial beliefs about the weights **w** before any data have arrived. This will typically be a flat (uniform) distribution in the weight space when all weight values are a priori equiprobable. When the data D are observed, the prior distribution of the parameter vector **w** is adjusted to a posterior distribution according to Bayes' theorem (level-1 inference). This gives

$$p(\mathbf{w}|D, \alpha, H) = \frac{p(D|\mathbf{w}, H)p(\mathbf{w}|\alpha, H)}{p(D|\alpha, H)}. \tag{9}$$

In the above expression $p(D|\mathbf{w}, H)$ is the likelihood function, which is the probability of the data occurring given the weights **w** and the functional form of the neural network H. The denominator of the expression in Eq. (9), i.e. $p(D|\alpha, H)$, is the normalisation factor that guarantees that the right-hand side of Eq. (9) integrates to one over the weight space. The latter is often referred to as the evidence for α. Hence, Eq. (9) can be restated as

$$\text{posterior} = \frac{\text{likelihood} \times \text{prior}}{\text{evidence}}. \tag{10}$$

Obtaining good predictive models is dependent on the use of the right prior distributions. MacKay uses Gaussian prior distribution functions in his operationalisation of Bayesian learning to approximate the posterior $p(\mathbf{w}|D, \alpha, H)$. In e.g. [12,68] other types of prior distributions have been used. When assuming a Gaussian prior for the weights **w** with zero mean and variance equal to $1/\alpha$, the probability distributions in the numerator of the right-hand side of Eq. (9) can be written as

$$\begin{aligned} p(D|\mathbf{w}, H) &= \prod_m (y^{(m)})^{t^{(m)}} (1 - y^{(m)})^{1-t^{(m)}} \\ &= \exp(-G), \\ p(\mathbf{w}|\alpha, H) &= \frac{1}{Z_W(\alpha)} \exp(-\alpha E_W) \end{aligned} \tag{11}$$

with $Z_W(\alpha) = (2\pi/\alpha)^{l/2}$ and l standing for the number of weight parameters. By substituting these probabilities into Eq. (9), we obtain

$$\begin{aligned} p(\mathbf{w}|D, \alpha, H) &= \frac{(1/Z_W(\alpha)) \exp(-(G + \alpha E_W))}{\text{evidence}} \\ &= \frac{1}{Z_M(\alpha)} \exp(-F(\mathbf{w})). \end{aligned} \tag{12}$$

The most probable weights $\mathbf{w}^{\text{MP}}$ can then be chosen so as to maximise the posterior probability $p(\mathbf{w}|D, \alpha, H)$. This is equivalent to minimising the regularised objective function $F(\mathbf{w}) = G + \alpha E_W$, since $Z_M(\alpha)$ is independent of the weights **w**. The most probable weight values $\mathbf{w}^{\text{MP}}$ (given the current setting of α) are thus found by minimising the objective function $F(\mathbf{w})$. Standard optimisation methods may be used to perform this task [4]. This concludes the first level of Bayesian inference.

Notice how Eq. (9) assumes that the value for the hyperparameter α is known, since the probability distributions were formulated as being contingent on the values of α. The hyperparameter α may again be optimised by applying Bayes' theorem, which is typical in an optimisation framework governed by Bayesian reasoning (level-2 inference). This yields

$$p(\alpha|D,H) = \frac{p(D|\alpha,H)p(\alpha|H)}{p(D|H)}. \tag{13}$$

Starting from Eq. (13) and assuming a uniform (non-informative) prior distribution $p(\alpha|H)$, the most probable α, α^{MP}, is obtained by maximising the likelihood function $p(D|\alpha,H)$. Notice that this likelihood function performs the role of the normalising constant in Eq. (9), where it was referred to as the evidence for α. Making use of Eq. (9) and making the Gaussian prior explicit, we can rewrite the normalisation factor as

$$\begin{aligned} p(D|\alpha,H) &= \frac{p(D|\mathbf{w},H)p(\mathbf{w}|\alpha,H)}{p(\mathbf{w}|D,\alpha,H)} \\ &= \frac{\exp(-G)(1/Z_W(\alpha))\exp(-\alpha E_W)}{(1/Z_M(\alpha))\exp(-F(\mathbf{w}))} \\ &= \frac{Z_M(\alpha)\exp(-G-\alpha E_W)}{Z_W(\alpha)\exp(-F(\mathbf{w}))} = \frac{Z_M(\alpha)}{Z_W(\alpha)}, \end{aligned} \tag{14}$$

where $Z_W(\alpha)$ is known from its definition in Eq. (11). The only part we need to determine in order to be able to optimise Eq. (14) is $Z_M(\alpha)$. The latter may be estimated by demanding that the right-hand side of Eq. (12) integrates to one over the weight space and approximating $F(\mathbf{w})$ by a second-order Taylor series expansion around $\mathbf{w}^{\mathrm{MP}}$. The hyperparameter α^{MP} may then be found by setting the derivative of the logarithm of Eq. (14) with respect to α to zero yielding

$$\alpha^{\mathrm{MP}} = \frac{\gamma}{2E_W(\mathbf{w}^{\mathrm{MP}})}, \tag{15}$$

where $\gamma = l - \alpha\mathrm{Trace}(\mathbf{H}^{\mathrm{MP}})^{-1}$ is called the effective number of parameters in the neural network. For more mathematical details see MacKay [33–35]. $\mathbf{H}^{\mathrm{MP}}$ stands for the Hessian matrix of the objective function $F(\mathbf{w})$ evaluated at $\mathbf{w}^{\mathrm{MP}}$. The effective number of parameters in a trained neural network is the number of well-determined weights indicating how many parameters of the NN are effectively used in reducing the error function $F(\mathbf{w})$. It can range from 0 to l. The α parameter is randomly initialised and the network is then trained in the usual manner by using standard optimisation algorithms [4], with the novelty that training is periodically halted for the weight decay parameter α to be updated. The latter may be done at each epoch of the NN training algorithm or after a fixed number of epochs. Notice that, since no validation set is required, all data can be used for training purposes.

An aspect of Bayesian learning we have not mentioned yet is model selection (level-3 inference). It is possible to choose between network architectures in a Bayesian way by using the evidence attributed to an architecture H referred to as $p(D|H)$ in [33–35]. Network models may then be ranked according to their evidence. However, in [51] it was empirically shown that for large data sets, the training error is as good a measure for model selection as is the evidence. For further details on Bayesian learning for neural networks we refer to [4,33–35,42]. In Section 5, we present another aspect of the evidence framework that plays an important role in the setup of this paper: input ranking using the ARD method.

5. Input ranking using automatic relevance determination (ARD)

Selecting the best subset of a set of n input variables as predictors for a neural network is a non-trivial problem. This follows from the fact that the optimal input subset can only be obtained when the input space is exhaustively searched. When n inputs are present, this would imply the need to evaluate $2^n - 1$ input subsets. Unfortunately, as n grows this very quickly becomes computationally infeasible [27]. For that reason, heuristic search procedures are often preferred. A multitude of input selection methods have been proposed in the context of neural networks [40,48,49,54]. These methods generally rely on the use of sensitivity heuristics, which try to measure the impact of input changes on the output of the trained network. Inputs may then be ranked (soft

input selection) and/or pruned (hard input selection) according to their sensitivity values. In this paper, we focus on input ranking as a means to assess the relative importance of the various inputs for the direct marketing case at hand. This is done by using the ARD method [36,43]. The ARD model is easily integrated within the evidence framework outlined in Section 4. It allows to perform soft input selection by ranking all inputs according to their relative importance for the trained network.

The ARD model introduces a weight decay hyperparameter for each input. For an MLP with one hidden layer and one output neuron, three additional weight decay constants are introduced: one associated with the connections from the input bias to the hidden neurons, one associated with the connections from the hidden neurons to the output neuron and one associated with the connection from the hidden bias neuron to the output neuron. This means that $n+3$ weight classes, each associated with one weight decay parameter α_k, are considered when n inputs are present. This setup is illustrated in Fig. 2. All weights of weight class k are then assumed to be distributed according to a Gaussian prior with mean 0 and variance $\sigma_k^2 = \frac{1}{\alpha_k}$ (see Eq. (11)). The evidence framework is thereupon applied to optimise all $n+3$ hyperparameters α_k by finding their most probable values α_k^{MP}.

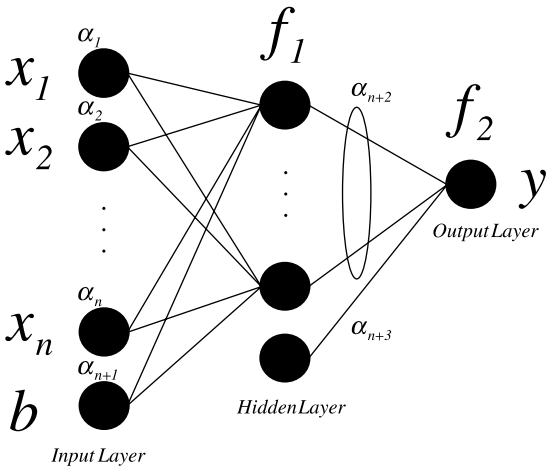

Fig. 2. Overview of the α_k hyperparameters introduced by the ARD method.

The most probable weights $\mathbf{w}^{\mathrm{MP}}$ are found by minimising the altered objective function

$$F(\mathbf{w}) = G + \sum_k \alpha_k E_{W(k)}, \tag{16}$$

where $E_{W(k)} = (1/2)\sum_i w_i^2$, with i running over all weights of weight class k. Analogous to the results obtained in the previous section, one obtains (level-2 inference)

$$\alpha_k^{\mathrm{MP}} = \frac{\gamma_k}{\sum_i (w_i^{\mathrm{MP}})^2}, \tag{17}$$

where γ_k is the number of well-determined parameters for weight class k with $\gamma_k = l_k - \alpha_k \mathrm{Trace}_k(\mathbf{H}^{\mathrm{MP}})^{-1}$. l_k is the number of parameters (weights) in weight class k and the trace is taken over those parameters only. All inputs may eventually be ranked according to their optimised α_k values. The most relevant inputs will have the lowest α_k values, since α_k is inversely proportional to the variance around 0 of the corresponding Gaussian prior.

One of the main advantages of ARD is that it allows to include a large number of potentially relevant input variables in the model without damaging effects [43]. Furthermore, it is integrated into the optimisation mechanism and completely rests upon the inspection of the optimised α_k parameters. Illustrations of ARD for input ranking can be found in [6,13,36,42,43,67].

6. Design of the study

6.1. Data set

From a major European mail-order company, we obtained data on past purchase behaviour at the order-line level, i.e. we know when a customer purchased what quantity of a particular product at what price as part of what order. This allowed us, in close cooperation with domain experts and guided by the extensive literature (see Section 2), to derive all the necessary purchase behaviour variables for a total sample size of 100,000 customers. For each customer, these variables were measured in the period between July 1st 1993 and

June 30th 1997. The goal is to predict whether an existing customer will repurchase in the observation period between July 1st 1997 and December 31st 1997 using the information provided by the purchase behaviour variables. This problem boils down to a binary classification problem: Will a customer (data instance m) repurchase ($t^{(m)} = 1$) or not ($t^{(m)} = 0$)? Again notice that the focus is on customer retention and not on customer acquisition. Of the 100,000 customers, 55.18% actually repurchased during the observation period.

6.2. Experimental setup

The experiment consists of two subexperiments. In Section 7, we start by concentrating on RFM variables only. Using these variables, we compare the performance of NNs trained using the evidence framework with that of three benchmark statistical classification techniques i.e. logistic regression, linear and quadratic discriminant analysis. We then discuss the relevance of the RFM variables using the ARD method presented in Section 5. In an attempt to further enrich the RFM response model, the same experiment is repeated with the input of other, potentially interesting customer profiling predictors which were handpicked by domain experts.

All performance assessments are computed on 10 bootstrap resamples generated from the original data set. Each bootstrap consists of 100,000 instances which are divided into a training set (50,000 instances) and a test set (50,000 instances). The former is used to train the classifier and the latter is used to estimate its generalisation behaviour. As a form of preprocessing, the inputs are statistically normalised to zero mean and unit variance by subtracting their mean and dividing by the standard deviation [4]. This is needed in order to be able to compare the relative importance of the various inputs by means of the ARD hyperparameters.

All neural network classifiers have one hidden layer with hyperbolic tangent transfer functions. A logistic transfer function is used in the output layer. The architecture of the MLP is determined by varying the number of hidden units between 2 and 14 in steps of 2. The hidden units have connections to all input units and also have a bias input. The single output is connected to all hidden units and again has a bias input. The number of epochs is set to 1000. The hyperparameter α is initialised to 0.2. All neural networks are trained with the quasi-Newton method to minimise a regularised cross-entropy error function. The hyperparameter α is updated every 100 epochs. All trained classifiers are evaluated by looking at their performance assessed on the independent test sets of all 10 bootstraps. All neural network analyses were done using the Netlab toolbox for Matlab implemented by Bishop and Nabney [4]. In the following section we provide an overview of the performance measures which were used in this paper.

6.3. Performance criteria for classification

The percentage correctly classified (PCC) cases, also known as the overall classification accuracy, is undoubtedly the most commonly used measure of the performance of a classifier. It simply measures the proportion of correctly classified cases on a sample of data D. Formally, it can be described as

$$\mathrm{PCC} = \frac{1}{N}\sum_{m=1}^{N}\delta(y_{0,1}^{(m)}, t^{(m)}), \tag{18}$$

where $y_{0,1}^{(m)}$ is the predicted class for instance m, $t^{(m)}$ is its true class label and $\delta(\cdot,\cdot)$ stands for the Kronecker delta function which equals 1 if both arguments are equal, 0 otherwise.

In a number of cases, the overall classification accuracy may not be the most appropriate performance criterion. It tacitly assumes equal misclassification costs for false positive and false negative predictions. This assumption is problematic, since for most real-world problems (e.g. fraud detection, customer credit scoring) one type of classification error may be much more expensive than the other. Another implicit assumption of the use of PCC as an evaluation metric is that the class distribution (class priors) among examples is presumed constant over time and relatively balanced [46]. For example, when confronted with a situation characterised by a very skewed class distribution in which faulty predictions for the underrepresented class are very costly, a model evaluated on PCC

alone may always predict the most common class and, in terms of PCC, provide a relatively high performance. Thus, using PCC alone often proves to be inadequate, since class distributions and misclassification costs are rarely uniform. However, taking into account class distributions and misclassification costs proves to be quite hard, since in practice they can rarely be specified precisely and are often subject to change [21]. In spite of the above, comparisons based on classification accuracy often remain useful because they are indicative of a broader notion of good performance [46].

Descriptive statistics such as the false positives, false negatives, sensitivity and specificity can provide more meaningful results. Class-wise decomposition of the classification of cases yields a confusion matrix as specified in Table 3. The following performance metrics can readily be distilled from Table 3:

$$\text{sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{19}$$

$$\text{specificity} = \frac{\text{TN}}{\text{FP} + \text{TN}}. \tag{20}$$

The sensitivity (specificity) measures the proportion of positive (negative) examples which are predicted to be positive (negative). Using the notation of Table 3, we may now formulate the overall accuracy as follows:

$$\text{PCC} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}}. \tag{21}$$

Note that sensitivity, specificity and PCC vary together as the threshold on a classifier's continuous output is varied between its extremes within the interval $[0, 1]$. The receiver operating characteristic curve (ROC) is a 2-dimensional graphical illustration of the sensitivity ('true alarms') on the Y-axis versus (1-specificity) ('false alarms') on the X-axis for various values of the classification threshold. It basically illustrates the behaviour of a classifier without regard to class distribution or error cost, so it effectively decouples classification performance from these factors [20,59,60].

Table 3
The confusion matrix for binary classification

Predicted	Actual	
	+	−
+	True positive (TP)	False positive (FP)
−	False negative (FN)	True negative (TN)

Fig. 3 provides an example of several ROC curves. Each ROC curve passes through the points $(0, 0)$ and $(1, 1)$. The former represents the situation whereby the classification threshold exceeds the highest output posterior probability value (meaning all instances are classified in class 0). In the latter case, the classification threshold is lower than the lowest posterior probability value (meaning all instances are classified in class 1). A straight line through $(0, 0)$ and $(1, 1)$ represents a classifier with poor discriminative power, since the sensitivity always equals (1-specificity) for all possible values of the classification threshold (curve A). It is to be considered as a benchmark for the predictive accuracy of other classifiers. The more the ROC curve approaches the $(0, 1)$ point, the better the classifier will discriminate (e.g. curve D dominates curves A, B and C). ROC curves of different classifiers may however intersect making a performance comparison less obvious (e.g. curves B and C). To overcome this problem, one often calculates the area under the receiver operating characteristic curve (AUROC). The AUROC provides a simple figure-of-merit for the performance of the constructed classifier. An intuitive interpretation of the AUROC is that it provides an estimate of the probability that a randomly chosen instance of class 1 is correctly rated (or ranked) higher than a randomly selected instance of class 0 [24].

In what follows, we consistently multiply AUROC values by a factor of 100 to give a number that is similar to PCC, with 50 indicating random and 100 indicating perfect classification.

7. Basic RFM experiment

7.1. Predictors used in the basic RFM experiment

We used two time horizons for all RFM variables. The Hist horizon refers to the fact that the

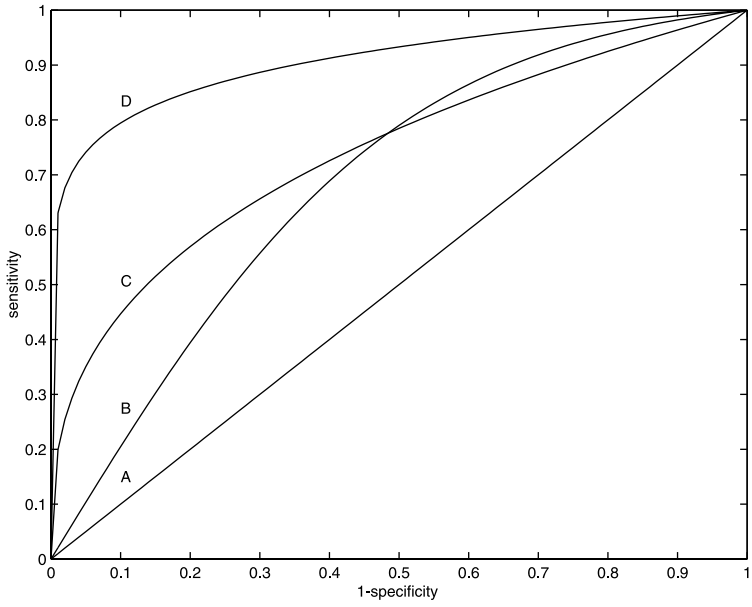

Fig. 3. The receiver operating characteristic curve (ROC).

variable is measured between the period July 1st 1993 until June 30th 1997. The Year horizon refers to the fact that the variable is measured over the last year. Including both time horizons allows us to check the argumentation that more recent data are much more relevant than historical data. All RFM variables are modelled both with and without the occurrence of returned merchandise, indicated by R and N in the variable name, respectively. The former is operationalised by including the counts of returned merchandise in the variable values, whereas in the latter case these counts are omitted. Taking into account both time horizons (Year versus Hist) and inclusion versus exclusion of returned items (R versus N), we arrive at a 2×2 design in which each RFM variable is operationalised in four ways.

For the Recency variable, many operationalisations have already been suggested. In this paper, we define the Recency variable as the number of days since the last purchase within a specific time window (Hist versus Year) and including or excluding returned merchandise (R versus N) [1]. Recency has been found to be inversely related to the probability of the next purchase, i.e. the longer the time delay since the last purchase the lower the probability of a next purchase within the specific period [15].

In the context of direct mail, it has generally been observed that multi-buyers (buyers who already purchased several times) are more likely to repurchase than buyers who only purchased once [1,57]. Although no detailed results are reported because of the proprietary nature of most studies, the Frequency variable is generally considered to be the most important of the RFM variables [41]. Bauer [1] suggests to operationalise the Frequency variable as the number of purchases divided by the time on the customer list since the first purchase. We choose to operationalise the Frequency variable as the number of purchases made in a certain time period (Hist versus Year) while including or excluding returned merchandise (R versus N).

In the direct marketing literature, the general convention is that the more money a person has spent with a company, the higher his/her likelihood of purchasing the next offering [31]. Nash [41] suggests to operationalise monetary value as the highest transaction sale or as the average order size. Levin and Zahavi [31] propose to use the average amount of money per purchase. We model the Monetary variable as the total accumulated

monetary amount of spending by a customer during a certain time period (Hist versus Year) while including or excluding returned merchandise (*R* versus *N*). Table 4 gives an overview of the different operationalisations of the RFM variables.

7.2. Results and discussion of the basic RFM experiment

The upper three rows of Table 5 contain the results for three benchmark statistical techniques: logistic regression, linear and quadratic discriminant analyses. The mean and standard deviation for the PCC and AUROC performance criteria are reported for training and test set over all 10 bootstrap resamples. The logistic regression classifier yields a mean classification accuracy of 70.3% and the mean area under the receiver operating characteristic curve amounts to 77.4% on the test set. It is clearly dominating both the linear and quadratic discriminant analysis classifiers when looking at the performance in terms of PCC and AUROC. This is confirmed by a series of paired student's *t*-tests using a significance level of 0.01. In all cases the resulting *P*-values proved to be smaller than 0.01. Notice the small difference between the test set and training set results for all statistical classifiers.

Table 4
Operationalisations of RFM variables used in the basic RFM experiment

Recency	Frequency	Monetary
RecHistN	FrHistN	MonHistN
RecHistR	FrHistR	MonHistR
RecYearN	FrYearN	MonYearN
RecYearR	FrYearR	MonYearR

Results for the Bayesian NN classifiers are presented in the second part of Table 5. The performance increases only slightly as the number of hidden neurons is varied between 2 and 6. From that point on, adding more hidden neurons seems to have no extra beneficial effect on both performance measures. Again, notice the small differences between the training and test set performances. This is a clear indication of the fact that no significant overfitting on the training set occurs while learning the NN (hyper)parameters. This may be attributed to the Bayesian way of learning the NN parameters, the weight regularisation mechanism, and the fact that both training and test sample size are rather large. Note that an NN with two hidden neurons already gives quite satisfying results. As noted above, we perform model selection using the training set error. Hence, we choose an NN with six hidden neurons yielding a mean PCC of 71.3% and a mean AUROC of 78.6% on the test set. Both the PCC and AUROC are significantly better for the Bayesian NN than for the logistic regression classifier. This is confirmed by the corresponding paired student's *t*-tests. The 1% point difference between the mean

Table 5
Performance assessment of all classifiers for the basic RFM experiment

	PCC		AUROC	
	Train	Test	Train	Test
Logistic regression	70.3 ± 0.1	70.3 ± 0.2	77.5 ± 0.1	77.4 ± 0.2
Linear discriminant analysis	68.9 ± 0.1	68.9 ± 0.2	76.0 ± 0.1	75.9 ± 0.2
Quadratic discriminant analysis	63.6 ± 0.4	63.2 ± 0.4	74.4 ± 0.2	74.3 ± 0.2
NN 2 hidden neurons	71.2 ± 0.1	71.2 ± 0.1	78.5 ± 0.1	78.4 ± 0.2
NN 4 hidden neurons	71.3 ± 0.2	71.2 ± 0.1	78.8 ± 0.2	78.5 ± 0.2
NN 6 hidden neurons	71.4 ± 0.2	71.3 ± 0.2	78.9 ± 0.2	78.6 ± 0.2
NN 8 hidden neurons	71.4 ± 0.2	71.3 ± 0.2	78.9 ± 0.2	78.6 ± 0.2
NN 10 hidden neurons	71.4 ± 0.2	71.3 ± 0.1	78.9 ± 0.2	78.6 ± 0.2
NN 12 hidden neurons	71.4 ± 0.2	71.3 ± 0.1	78.9 ± 0.2	78.6 ± 0.2
NN 14 hidden neurons	71.4 ± 0.2	71.3 ± 0.1	78.9 ± 0.2	78.6 ± 0.2
NN ARD six hidden neurons	71.4 ± 0.2	71.2 ± 0.1	78.7 ± 0.2	78.5 ± 0.2

PCC of both classifiers is important from a direct marketing perspective as discussed in Section 2. The final part of Table 5 depicts the performance results of an NN ARD classifier with six hidden neurons. The ARD method yields a mean PCC of 71.2% and a mean AUROC of 78.5% on the test set, which is comparable to the NN non-ARD results reported in the second part of the table.

Fig. 4 reports the error bars representing the 95% confidence intervals for the α_k (on a logarithmic scale) of the various inputs over the 10 bootstrap resamples. The α_k coefficients of the NN ARD classifiers are also used to obtain a ranking of the importance of all 12 weight classes corresponding to the RFM inputs. All 12 α_k coefficients are mapped into a ranking from 1 to 12 for each of the 10 runs of the ARD experiment. The weight class corresponding to the lowest α_k is ranked first because it is considered most important according to the ARD semantics. Insight into the rankings produced by all 10 runs of the experiment is then obtained in the following way. We created a 12×12 matrix R with elements $R(i,j)$ indicating how many times weight class i was ranked at the jth position aggregated over all 10 runs. We visualise the matrix R in the form of the contour plot presented in Fig. 5.

There is broad agreement between both plots concerning the relative importance of the inputs. The dark zone in the contour plot at the intersection of rank 1 and the RecHistR variable clearly indicates its importance. This variable was 10 times ranked first. The RecYearR and RecYearN variables seem to be very useful as well. Note that the RecYearR variable was always ranked second over all 10 runs. These findings are confirmed by the relatively low mean $\log(\alpha_k)$ values and narrow confidence intervals for the RecHistR, RecYearR and RecYearN variables as depicted in Fig. 4. The rankings of the variables belonging to the Frequency category are concentrated in the zone covering ranks 4 to 8. This suggests that these variables are of medium importance to the NN prediction. The ranking of the MonYearN variable is concentrated around rank 4. The other Monetary variables are ranked between ranks 8 and 12. Fig. 4 also indicates that the MonYearN variable is the most important among the set of Monetary predictors. Notice that neither plot clearly indicates the irrelevance of predictors included in the study. Therefore, we conclude that a combined use of predictors of all three categories is desirable for response modelling. Moreover, it can be stated that the way a variable is opera-

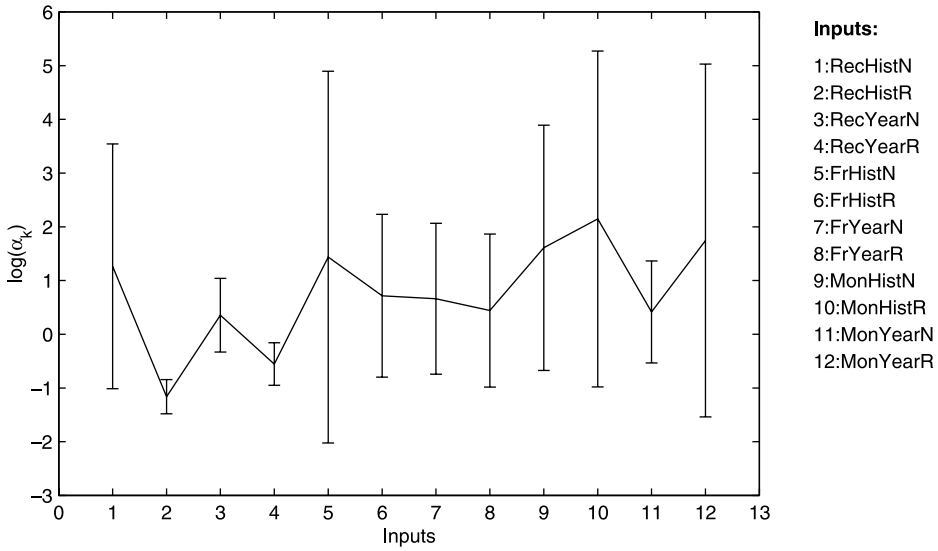

Fig. 4. Error bars for the $\log(\alpha_k)$ parameters for the basic RFM experiment.

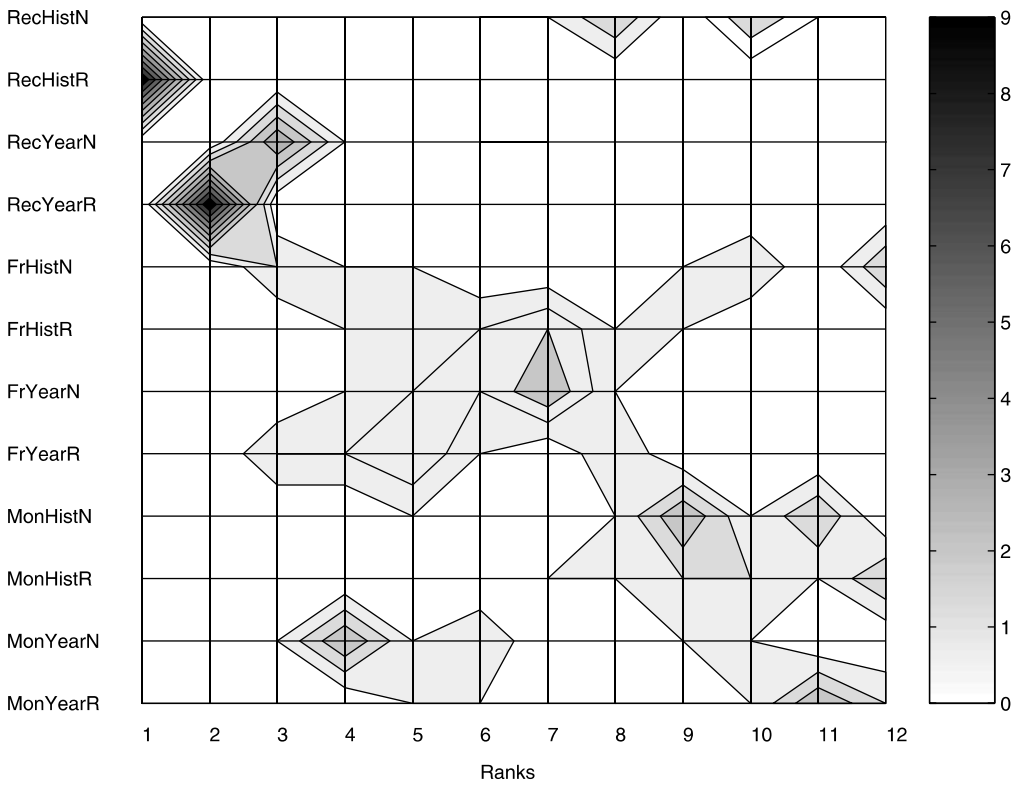

Fig. 5. Contour plot of the matrix R for the basic RFM experiment.

tionalised has a substantial impact on its predictive performance.

8. Extended RFM experiment

8.1. Predictors used in the extended RFM experiment

Apart from the RFM variables discussed in Section 7.1, we now include 10 other customer profiling features (referred to as 'Other' in Table 6) [62].

The type and frequency of contact which customers have with the mail-order company may yield important information about their future purchasing behaviour. The GenInfo and GenCust are binary customer/company interaction variables indicating whether the customer asked for general information, respectively, filed general complaints. Since customer (dis)satisfaction may not only be revealed by general complaints but also by returning items, we included two extra variables.

Table 6
Operationalisations of both RFM and non-RFM variables used in the extended RFM experiment

Recency	Frequency	Monetary	Other		
RecHistN	FrHistN	MonHistN	RetPerc	Ndays	IncrHist
RecHistR	FrHistR	MonHistR	RetMerch	Credit	IncrYear
RecYearN	FrYearN	MonYearN	ProdclaT	GenInfo	
RecYearR	FrYearR	MonYearR	ProdclaM	GenCust	

The RetMerch variable is a binary variable indicating whether the customer has ever returned an item that was previously ordered from the mail-order company. The RetPerc variable measures the total monetary amount of returned orders divided by the total amount of spending. The Ndays variable models the length of the customer relationship in days. It is commonly believed that consumers/households with a longer relationship with the company have a higher probability of repurchase than households with shorter relationships. IncrHist and IncrYear are operationalisations of a behavioural loyalty measure. We propose to perform a median split of the length of the relationship (time since the household became a customer). This enables us to compare the number of purchases (i.e. frequency) between the first and last half of the time window. The following formula is used:

$$\frac{\text{purchases second half} - \text{purchases first half}}{\text{purchases first half}}. \tag{22}$$

If the above measure is positive, this may give us an indication of increasing loyalty by that customer to the (mail-order) company, and ipso facto satisfaction with the current level of service. Remember that the suffix Hist reflects that the whole purchase history is used, whereas in the case of the suffix Year, only transactions from the last year are included. The mail-order company has internal records whether a customer uses the credit facilities. This may function as an indicator of the extent to which the customer values the financial convenience of mail-order buying. Therefore, we also include the binary Credit variable. The ProdclaT, respectively, ProdclaM variables represent the total (T), respectively, mean (M) forward-looking weighted productindex. The weighting procedure represents the 'forward-looking' nature of a product category purchase derived from another sample of data.

8.2. Results and discussion of the extended RFM experiment

The performance measurements of the extended RFM experiment are presented in Table 7. The setup of this table is analogous to that of Table 5. In general, both the PCC and AUROC of the three benchmark statistical classifiers rise about 1% point due to the inclusion of the 10 extra variables. Again, this is a result which may not be underestimated in terms of profit increases for the direct marketing company (see Section 2). As confirmed by the corresponding paired student's t-tests, the logistic regression classifier still yields the best performance of all three statistical classifiers.

In this case, we opt for a Bayesian NN with eight hidden neurons as our NN model of choice,

Table 7
Performance assessment of all classifiers for the extended RFM experiment

	PCC		AUROC	
	Train	Test	Train	Test
Logistic regression	71.4 ± 0.2	71.4 ± 0.2	78.7 ± 0.2	78.6 ± 0.2
Linear discriminant analysis	69.7 ± 0.2	69.7 ± 0.2	76.9 ± 0.2	76.8 ± 0.2
Quadratic discriminant analysis	65.0 ± 0.8	64.9 ± 0.7	73.3 ± 1.2	73.2 ± 1.2
NN 2 hidden neurons	72.3 ± 0.3	72.2 ± 0.2	79.7 ± 0.2	79.5 ± 0.2
NN 4 hidden neurons	72.5 ± 0.2	72.4 ± 0.2	80.0 ± 0.2	79.7 ± 0.2
NN 6 hidden neurons	72.6 ± 0.3	72.3 ± 0.2	80.2 ± 0.2	79.7 ± 0.2
NN 8 hidden neurons	72.8 ± 0.3	72.4 ± 0.2	80.4 ± 0.2	79.8 ± 0.2
NN 10 hidden neurons	72.8 ± 0.3	72.4 ± 0.2	80.4 ± 0.2	79.8 ± 0.2
NN 12 hidden neurons	72.8 ± 0.3	72.4 ± 0.2	80.4 ± 0.2	79.8 ± 0.2
NN 14 hidden neurons	72.8 ± 0.3	72.4 ± 0.2	80.4 ± 0.2	79.8 ± 0.2
NN ARD eight hidden neurons	72.5 ± 0.3	72.4 ± 0.3	80.0 ± 0.3	79.7 ± 0.2

since at this point adding more hidden neurons seems to provide no extra performance gains. Also notice that for the Bayesian NN, performance again increases about 1% point when compared to the basic RFM experiment. This results in a mean PCC of 72.4% and a mean AUROC of 79.8% on the test set. Training an NN ARD with eight hidden neurons yields a mean PCC of 72.4% and mean AUROC of 79.7% on the test set, a result that is comparable to the NN non-ARD results.

Again, Fig. 6 presents the 95% confidence intervals for the α_k values on a logarithmic scale. The matrix R associated with the weight class rankings is depicted in the contour plot given by Fig. 7. Among the RFM variables, the same relevance patterns are present as for the basic RFM experiment, thus essentially confirming the latter. The rankings of the RetPerc, RetMerch, ProdclaT, ProdclaM, Ndays, IncrHist and IncrYear variables are concentrated in the region of lesser importance in the contour plot. When looking at both plots, it can be observed that the Credit, GenInfo and GenCust variables are definitely more relevant to the trained networks. The 1% point performance rise may thus be especially attributed to the inclusion of these three variables in the extended RFM response model.

When comparing the results of this study to those on similar data sets from the same anonymous company, reported in [62,65,66], we observe that the insight gained using Bayesian neural network methods generally confirms previous findings. Most noticeably they also highlight: (1) the importance of a combined use of all three RFM predictor categories in predicting mail-order repeat purchase behaviour; (2) the performance gains by including non-RFM variables into the response model. However, there is some disagreement considering the relative importance of some of the RFM and non-RFM variables. These differences may be due to: (1) different data sets from different countries, resulting in a.o. diverging class proportions (i.e., 38% buyers in [65,66] compared to 55% buyers in this study); (2) inclusion of other predictors or alternative transformations (e.g. logarithmic transformation to reduce the skewness in [65,66]); (3) the use of other classification techniques (e.g. support vector machines in [66]) and

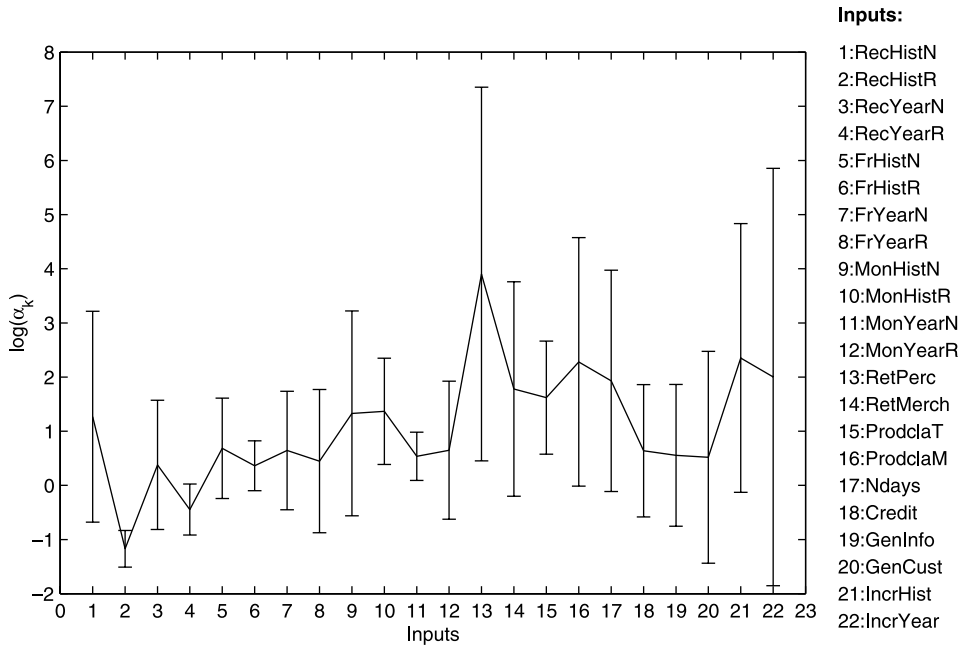

Fig. 6. Error bars for the $\log(\alpha_k)$ parameters for the extended RFM experiment.

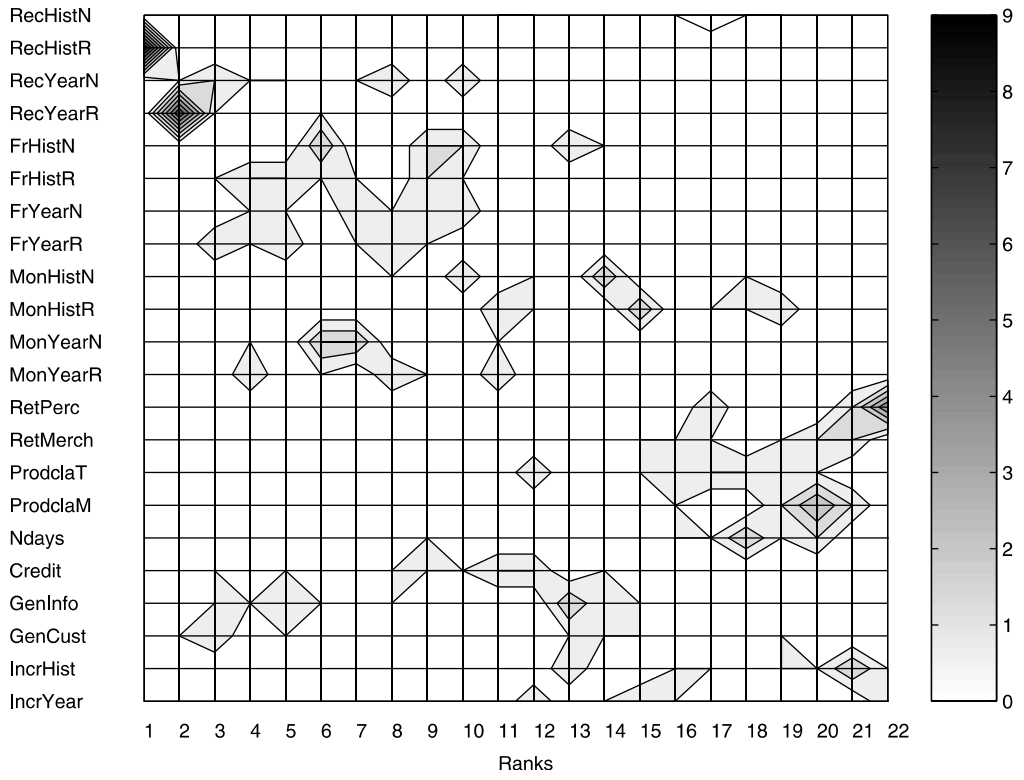

Fig. 7. Contour plot of the matrix R for the extended RFM experiment.

input selection heuristics (e.g. hard sensitivity based pruning in [65,66]).

9. Conclusion

In this paper, we focus on purchase incidence modelling for a major European direct mail company. The case boils down to a binary classification problem: Will the customer repurchase or not? Response models based on statistical and neural network techniques are developed and contrasted. The latter are trained using Bayesian neural network learning, a method that is fairly robust with respect to the problems of overfitting and (hyper)parameter choice, problems that are typically encountered when implementing neural networks. The evidence framework of MacKay is used as an example implementation of Bayesian learning. The automatic relevance determination (ARD) method is an additional feature of this framework that allows us to assess the relative importance of the inputs. The basic response models use operationalisations of the traditionally discussed Recency, Frequency and Monetary (RFM) predictor categories. In a second experiment, the RFM response framework is enriched by the inclusion of other (non-RFM) customer profiling predictors. In this study, we contribute to the literature by providing a thorough investigation into: (1) the suitability of Bayesian neural networks for repeat purchase modelling; (2) the predictive performance of alternative operationalisations of RFM variables and their relative importance; (3) the issue whether other (non-RFM) variables add predictive power to the traditional RFM variables. By means of experimental evaluation, it is illustrated that, from a performance perspective, Bayesian neural networks offer an interesting and viable alternative for purchase incidence modelling. Performance of the trained classifiers is measured using the percentage cor-

rectly classified (PCC) and the area under the receiver operating characteristic curve (AUROC). The ARD results advocate a combined use of all three RFM predictor categories for response modelling. Finally, as illustrated by a second experiment, the inclusion of non-RFM variables allows to further augment the predictive power of the constructed classifiers. The ARD results mainly attribute this rise to the inclusion of customer/company interaction variables and to a variable measuring whether a customer uses the credit facilities of the direct mail company.

Acknowledgements

This work was partly carried out at the Leuven Institute for Research on Information Systems (LIRIS) of the Department of Applied Economic Sciences of the K.U.Leuven in the framework of the KBC Insurance Research Chair, set up in September 1997 as a pioneering cooperation between LIRIS and KBC Insurance. We thank Prof. J.A.K. Suykens (K.U.Leuven) for his suggestions and comments. We are also grateful to Prof. Marnik Dekimpe (K.U.Leuven) and Prof. Joseph Leunis (K.U.Leuven) for their numerous useful comments on earlier versions of parts of this document.

References

[1] A. Bauer, A direct mail customer purchase model, Journal of Direct Marketing 2 (3) (1988) 16–24.
[2] P. Berger, T. Magliozzi, The effect of sample size and proportion of buyers in the sample on the performance of list segmentation equations generated by regression analysis, Journal of Direct Marketing 6 (1) (1992) 13–22.
[3] M.J.A. Berry, G. Linoff, Mastering Data Mining, Wiley, Chicago, 2000.
[4] C.M. Bishop, Neural Networks for Pattern Recognition, Oxford University Press, Oxford, 1995.
[5] G.R. Bitran, S.V. Mondschein, Mailing decisions in the catalog sales industry, Management Science 42 (9) (1996) 1364–1381.
[6] B.V. Bonnlander, Nonparametric selection of input variables for connectionist learning, Ph.D. thesis, University of Colorado, Department of Computer Science, 1996.
[7] J.S. Bridle, Neuro-Computing: Algorithms, Architectures and Applications, Springer, Berlin, 1989 (Chapter: Probabilistic interpretation of feedforward classification network output, with relationships to statistical pattern recognition).
[8] J.R. Bult, Semiparametric versus parametric classification models an application to direct marketing, Journal of Marketing Research 30 (1993) 380–390.
[9] J.R. Bult, Target selection for direct marketing, Ph.D. thesis, Groningen University, 1993.
[10] J.R. Bult, H. Van der Scheer, T. Wansbeek, Interaction between target and mailing characteristics in direct marketing, with an application to health care fund raising, The International Journal of Research in Marketing 14 (1997) 301–308.
[11] J.R. Bult, D.R. Wittink, Estimating and validating asymmetric heterogeneous loss functions applied to health care fund raising, The International Journal of Research in Marketing 13 (1996) 215–226.
[12] W.L. Buntine, A.S. Weigend, Bayesian back-propagation, Complex Systems 5 (1991) 603–643.
[13] F.R. Burden, M.G. Ford, D.C. Whitley, D.A. Winkler, The use of automatic relevance determination in qsar studies using Bayesian neural networks, Journal of Chemical Information and Computer Sciences 40 (2000) 1423–1430.
[14] L.K. Chang, I. Han, Y. Kwon, Hybrid neural network models for bankruptcy predictions, Decision Support Systems 18 (1) (1996) 63–72.
[15] G.J. Cullinan, Picking them by their batting averages' recency–frequency–monetary method of controlling circulation, Manual release 2103, Direct Mail/Marketing Association, NY, 1977.
[16] M.G. Dekimpe, Z. Degraeve, The attrition of volunteers, European Journal of Operational Research 98 (1997) 37–51.
[17] V.S. Desai, J.N. Crook, G.A. Overstreet Jr., A comparison of neural networks and linear scoring models in the credit union environment, European Journal of Operational Research 95 (1) (1996) 24–37.
[18] W.S. Desarbo, V. Ramaswamy, Crisp: Customer response-based iterative segmentation procedures for response modeling in direct marketing, Technical Working Paper 94–102, Marketing Science Institute, 1994.
[19] DMA(1998), Statistical Fact Book 1998, 20th ed., Direct Marketing Association, New York, 1998.
[20] J.P. Egan, Signal Detection Theory and ROC analysis. Series in Cognition and Perception, Academic Press, New York, 1975.
[21] T. Fawcett, F. Provost, Adaptive fraud detection, Data Mining and Knowledge Discovery 1–3 (1997) 291–316.
[22] L.W. Glorfeld, B.C. Hardgrave, An improved method for developing neural networks: The case of evaluating commercial loan creditworthiness, Computers and Operations Research 23 (10) (1996) 933–944.
[23] F. Gönül, M.Z. Shi, Optimal mailing of catalogs: A new methodology using estimable structural dynamic programming models, Management Science 44 (9) (1998) 1249–1262.

[24] J.A. Hanley, B.J. McNeil, The meaning and use of the area under a receiver operating characteristic (ROC) curve, Radiology 148 (1983) 839–843.
[25] B. Hauser, The Direct Marketing Handbook, 1992, pp. 233–247 (Chapter: List segmentation).
[26] K. Hornik, M. Stinchcombe, H. White, Multilayer feedforward networks are universal approximators, Neural Networks 2 (5) (1989) 359–366.
[27] G. John, R. Kohavi, K. Pfleger, Irrelevant features and the subset selection problem, in: H. Hirsh, W. Cohen (Eds.), Machine Learning: Proceedings of the 11th International Conference, Morgan Kaufmann, San Francisco, 1994, pp. 121–129.
[28] G.A. Kaslow, A microeconomic analysis of consumer response to direct marketing and mail order, Ph.D. thesis, California Institute of Technology, 1997.
[29] R.D. Kestnbaum, The Direct Marketing Handbook, 1992, pp. 588–597 (Chapter: Quantitative database methods).
[30] R.C. Lacher, P.K. Coats, C. Sharma Shanker, L.F. Fant, A neural network for classifying the financial health of a firm, European Journal of Operational Research 85 (1) (1995) 53–65.
[31] N. Levin, J. Zahavi, Segmentation analysis with managerial judgment, Journal of Direct Marketing 10 (3) (1996) 28–47.
[32] N. Levin, J. Zahavi, Continuous predictive modeling: A comparative analysis, Journal of Interactive Marketing 12 (2) (1998) 5–22.
[33] D.J.C. MacKay, Bayesian interpolation, Neural Computation 4 (3) (1992) 415–447.
[34] D.J.C. MacKay, Bayesian methods for adaptive models, Ph.D. thesis, Computation and Neural Systems, California Institute of Technology, Pasadena, CA, 1992.
[35] D.J.C. MacKay, The evidence framework applied to classification networks, Neural Computation 4 (5) (1992) 720–736.
[36] D.J.C. MacKay, Bayesian non-linear modelling for the prediction competition, ASHRAE Transactions 100 (Part 2) (1994) 1053–1062.
[37] T.L. Magliozzi, An empirical investigation of regression meta-strategies for direct marketing list segmentation models, Ph.D. thesis, Boston University, 1989.
[38] T.L. Magliozzi, P.D. Berger, List segmentation strategies in direct marketing, Omega International Journal of Management Science 21 (1) (1993) 61–72.
[39] B.A. Mobley, E. Schechter, W.E. Moore, P.A. McKee, J.E. Eichner, Predictions of coronary artery stenosis by artificial neural network, Artificial Intelligence in Medicine 18 (3) (2000) 187–203.
[40] J.E. Moody. Note on generalization, regularization and architecture selection in nonlinear learning systems, in: J.E. Moody, S.J. Hanson, R.P. Lippmann (Eds.), First IEEE-SP Workshop on Neural Networks for Signal Processing, IEEE Computer Society Press, Los Alamitos, CA, 1991, pp. 1–10.
[41] E.L. Nash, Direct Marketing: Strategy Planning Execution, 3rd ed., McGraw-Hill, New York, 1994.
[42] R.M. Neal, Bayesian Learning for Neural Networks, Lecture Notes in Statistics, vol. 118, Springer, New York, 1996.
[43] R.M. Neal, Neural Networks and Machine Learning, Springer, Berlin, 1998, pp. 97–129 (Chapter: Assessing Relevance Determination Methods using Delve).
[44] D. Pantazopoulos, P. Karakitsos, A. Ioakim-Liossi, A. Pouliakis, K. Dimopoulos, Comparing neural networks in the discrimination of benign from malignant low urinary lesions, British Journal of Urology 81 (1998) 574–579.
[45] S. Piramuthu, Financial credit-risk evaluation with neural and neurofuzzy systems, European Journal of Operational Research 112 (2) (1999) 310–321.
[46] F. Provost, T. Fawcett, R. Kohavi, The case against accuracy estimation for comparing classifiers, in: J. Shavlik, (Ed.), Proceedings of the 15th International Conference on Machine Learning, Morgan Kaufmann, San Francisco, CA, 1998, pp. 445–453.
[47] V.R. Rao, J.H. Steckel, Selecting, evaluating and updating prospects in direct mail marketing, Technical Working Paper 94–121, Marketing Science Institute, 1994.
[48] R. Reed, Pruning algorithms – A survey, IEEE Transactions on Neural Networks 4 (5) (1993) 740–747.
[49] A.P.N. Refenes, A.D. Zapranis, Neural network model identification, variable selection and model adequacy, Journal of Forecasting 18 (1999) 299–332.
[50] B.D. Ripley, Pattern Recognition and Neural Networks, Cambridge University Press, Cambridge, 1996.
[51] S.J. Roberts, W.D. Penny, Bayesian neural networks for classification: How useful is the evidence framework?, Neural Networks 12 (1998) 877–892.
[52] D.E. Rumelhart, G.E. Hinton, R.J. Williams, Parallel Distributed Processing: Explorations in the Microstructure of Cognition, vol. 1, MIT Press, Reprinted in Anderson and Rosenfeld, Cambridge, MA, 1986, pp. 318–362 (Chapter: Learning internal representations by error propagation).
[53] L. Salchenberger, R. Venta Enrique, A. Venta Luz, Using neural networks to aid the diagnosis of breast implant rupture, Computers and Operations Research 24 (5) (1997) 435–444.
[54] R. Setiono, H. Liu, Neural-network feature selector, IEEE Transactions on Neural Networks 8 (3) (1997) 654–662.
[55] R. Sharda, R. Wilson, Neural network experiments in business failures prediction: A review of predictive performance issues, International Journal of Computational Intelligence and Organizations 1 (2) (1996) 107–117.
[56] S.A. Solla, E. Levin, M. Fleisher, Accelerated learning in layered neural networks, Complex Systems 2 (1988) 625–640.
[57] B. Stone, Successful Direct Marketing Methods, Crain Books, Chicago, 1984.
[58] J.A.K. Suykens, J. Vandewalle, Nonlinear Modeling: Advanced Black-Box Techniques, Kluwer Academic Publishers, Boston, 1998.
[59] J.A. Swets, Roc analysis applied to the evaluation of medical imaging techniques, Investigative Radiology 14 (1989) 109–121.

[60] J.A. Swets, R.M. Pickett, Evaluation of Diagnostic Systems: Methods from Signal Detection Theory, Academic Press, New York, 1982.
[61] R.P. Thrasher, Cart: A recent advance in tree-structured list segmentation methodology, Journal of Direct Marketing 5 (1) (1991) 35–47.
[62] D. Van den Poel, Response modeling for database marketing using binary classification, Ph.D. thesis, K.U. Leuven, 1999.
[63] D. Van den Poel, J. Leunis, Database marketing modeling for financial services using hazard rate models, International Review of Retail Distribution and Consumer Research 8 (2) (1998) 243–257.
[64] H.R. Van der Scheer, Quantitative approaches for profit maximization in direct marketing, Ph.D. thesis, Groningen University, 1998.
[65] S. Viaene, B. Baesens, D. Van den Poel, G. Dedene, J. Vanthienen, Wrapped input selection using multilayer perceptrons for repeat-purchase modeling in direct marketing, International Journal of Intelligent Systems in Accounting, Finance & Management 10 (2) (2001) 115–126.
[66] S. Viaene, B. Baesens, T. Van Gestel, J.A.K. Suykens, D. Van den Poel, J. Vanthienen, B. De Moor, G. Dedene, Knowledge discovery using least squares support vector machine classifiers: A direct marketing case, International Journal of Intelligent Systems, 2001 (forthcoming).
[67] F. Vivarelli, C.K.I. Williams, Using Bayesian neural networks to classify segmented images, in: Proceedings of the Fifth IEE International Conference on Artificial Neural Networks, 1997.
[68] P.M. Williams, Bayesian regularization and pruning using a laplace prior, Neural Computation 7 (1) (1995) 117–143.
[69] J. Zahavi, N. Levin, Issues and problems in applying neural computing to target marketing, Journal of Direct Marketing 11 (4) (1997) 63–75.
[70] G. Zhang, M.Y. Hu, B.E. Patuwo, D.C. Indro, Artificial neural networks in bankruptcy prediction: General framework and cross-validation analysis, European Journal of Operational Research 116 (1999) 16–32.
[71] J.M. Zurada, Introduction to Artificial Neural Systems, PWS Publishing Company, Boston, 1995.