DECISION SUPPORT SYSTEMS LABORATORY

Data Mining with Neural Networks for CRM

DECISION SUPPORT SYSTEMS LABORATORY

WHAT ARE NEURAL NETWORKS?

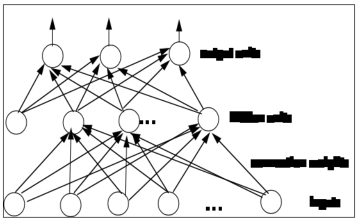
- Neural networks are parallel information processing systems with their architecture inspired by the structure and functioning of the brain
- A neural network have the ability to learn and generalise.
- Neural networks can be trained to make classifications and predictions based on historical data
- Neural nets are included in many data mining products
- Very popular and effective techniques

DECISION SUPPORT SYSTEMS LABORATORY

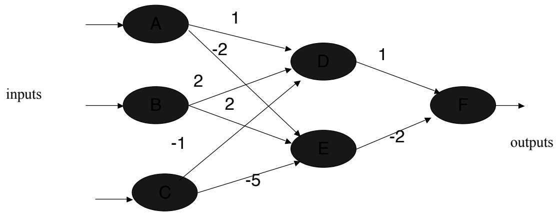
Neural Networks

- Neural networks consist of layers of connected processing units (artificial neurons).
- The strength of the connections between the neurons is expressed by their connection weights
- Typically, the neural network is trained on a set of examples
- (input, output pairs)
- Once it has learned the patterns in the data it is able to predict the output when presented with previously unseen input.

Neural Network



Neural network with a hidden layer



Neural Networks – biological inspiration

- Inspiration for the concept comes from research into the neural network structure of the brain
- The brain is massively parallel - it consists of layers of neurons

THE BRAIN

The *human brain* contains approx. 10^{10} neurons

The *neuron* is the basic processing unit of the brain

The *dendrites* act as the connections through which all the inputs to the neuron arrive

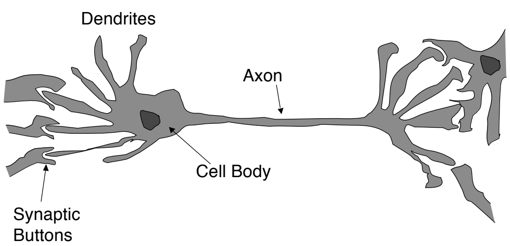
The *soma* or *cell body* performs a kind of summation of the inputs

The *axon* is the output channel of the neuron; it terminates in a synapse

The *synapse* serves as a contact with the dendrite of another cell

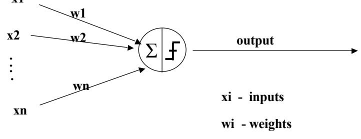
a single neuron has many synaptic inputs on its dendrites and may have many synaptic outputs connecting it to other cells.

"Real" neurons



Artificial Neuron, PERCEPTRON

Frank Rosenblatt's (1969) *perceptron* was one of the earliest neural networks models



the perceptron consists of the weights , the summation processor , and the adjustable threshold processor

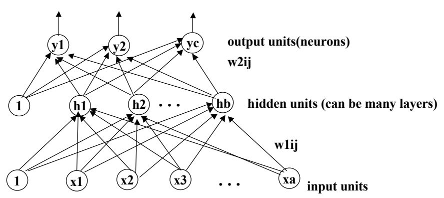
if the wighted sum is greater than the threshold value , than out put is 1 (perceptron fires), if equal or less , output is 0

Backpropagation neural network

- The backpropagation network suggested in 1986 by Rumelhart
- Most common neural network type
- It is a multilayer perceptron with a learning rule called generalised delta rule and a sigmoid transfer function
- It is an example of a supervised neural network

Backpropagation Networks

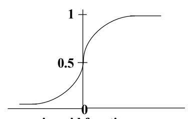
A typical back-propagation network always has an input layer, an output layer and at least one hidden layer.



fully connected, layered, feedforward network

Backpropagation

A backpropagation neuron sums up its weighted inputs and produces a real value between 0 and 1 as output, based on a sigmoid function



sigmoid function

$$\text{output} = \frac{1}{1 + e^{-sum}}$$

where *sum* is the weighted sum of the inputs to the neuron

DECISION SUPPORT SYSTEMS LABORATORY

Neural Nets Training

Backpropagation uses supervised training:

For each input example, a desired output or target is provided

- The output calculated by the neural net is compared to the target
- weights are adjusted so the next iteration will produce a closer match
- the goal is to minimise the error between the target and the current output.

DECISION SUPPORT SYSTEMS LABORATORY

Training backpropagation neural networks

Backpropagation training(learning)- general idea:

0. Randomise weights
1. Pass forward: present input example and calculate output
2. Compare the calculated output to the corresponding target output, calculate error and adjust weights (backpropagation of errors)
3. Processes 1and 2 are repeated until some criteria are met (eg. the error cannot be further reduced)

DECISION SUPPORT SYSTEMS LABORATORY

Backpropagation

Learning Algorithm (generalised delta rule)

1. initialise the weights to small random values Eg between -.01 and 0.1

2. choose an input/ output pair

X(x1,x2...xa) input vector , **D(d1,d2...dc)** desired output vector

3. propagate the input to the hidden layer using an appropriate transfer function (in our case it is the sigmoid function)

$$hj = \frac{1}{1 + e^{-s}} \quad \text{where} \quad s = \sum_{i=0}^{a} w1ijxi \quad \text{for all } j=1\ldots b$$

4. pass the vector H(hi,h2...hb) as input to the output layer and calculate the output vector Y(y1,y2...yc)

$$yj = \frac{1}{1 + e^{-s}} \quad \text{where} \quad s = \sum_{i=0}^{b} w2ijhi \quad \text{for all } j=1\ldots c$$

Backpropagation Learning Algorithm cont.

5. compute the errors of the units in the output layer

$\delta 2j = yj(1 - yj)(dj - yj)$ for all j= 1...c

6. compute the errors of the units in the hidden layer

$\delta 1j = hj\,(1 - hj) \sum_{i=1}^{c} \delta 2i\; w2ji$ for all j= 1…b

7. adjust the weights between the hidden layer and the output layer

$\Delta\, w2ij = \eta\delta\; 2j\; hi$ for all i = 0...b, j = 1...c

where η is the learning rate

8. adjust the weights between the input layer and the hidden layer

$\Delta\, w1ij = \eta\delta\; 1j\; xi$ for all i=0...a ,j = 1...b

9. go to step 2

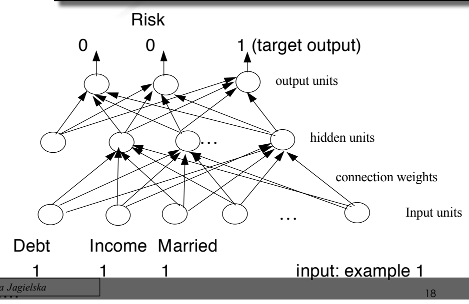
Training stops when convergence criteria are met

Example, Assessing Credit Risk

(example classification/prediction problem)

Name	Debt	Income	Married	Risk(target)
Joe	1	1	1	good
Sue	0	1	1	good
John	0	1	0	good
Mary	1	0	1	bad
Fred	0	0	1	bad
Peter	0	1	0	?

Assessing Credit Risk example



NEURAL NETWORKS
generalisation

- Properly trained neural network must be able to produce correct classification/prediction when given new (unseen) data
- This capability is called generalisation
- Overfitting – overfitted NN cannot generalise

Building Backpropagation networks

1. select input and output variables
2. prepare training and testing examples
3. select neural network configuration:
number of layers,
number of neurons,
learning parameters
4. train network if unsuccessful go to 3
5. test network
if results unsatisfactory may need to repeat from step 1

Building backpropagation neural network models

How many hidden layers?
How many neurons in each layer?
depends on application , must be determined experimentally
some tips
start with one hidden layer
backpropagation is most suited to small networks of three layers
the bigger the network the slower each pass through the training data
the more complex the relationship between input and output , the more layers and neurons required

Building neural network for models for CRM

- http://www.cheshireeng.com/Neuralyst/
- download the demo software and training manual
- build, train and test a neural network for :
  - 1. The Credit Rater: customer credit risk rating example (EZCREDIT.XLS)
  - 2. The Marketing Analyser (FIZZY.XLS): budgeting advertising money example

CRM applications

- The main areas of neural networks applications for CRM are :
- Prediction/forecasting:
  - Credit risk
  - Churn prediction
  - Sales forecasting
  - Marketing campaigns - customer response prediction

In general, where prediction or classification is required

Neural Network for Predicting the Performance of Credit Card Accounts.
(case study)

- Developed for a major financial institution
- neural network models were developed for Visa Classic, Visa Gold to predict the performance of credit card accounts based on the accounts historical data

The Input/Output Variables

- Input:
  - age, postcode, time at address, home phone number, residential status, occupation, time currently employed, time previously employed, number of loans, identification of the applicant, additional cardholder, bureau status, gender, number of dependants, martial status, spouse employed, referee phone, time bank account open, net assets, income, bank balance, surplus, number of defaults.
- Output consisted of a simple classification of good and bad accounts.

The Data

The samples of 2,430 Visa Gold and 6,000 Visa Classic cardholder data were randomly selected from the cardholder historical data bases

Segments: the population of Visa Classic accounts was additionally divided into a population of applicants under 26 years of age and over 26 years of age

.

Training and Testing

The samples were divided into training and testing sets.

Training and testing was performed using variety of architectures

The network configurations used in the experiment

	input layer	hidden layer 1	hidden layer 2	hidden layer 3	output layer
network 1	23	20	-	-	2
network 2	23	7	1	-	2
network 3	23	30	10	2	2

RESULTS

	Accuracy on testing	% good	% bad
Visa Classic under 26			
network 2	78.8	91.5	28.0
network 3	81.4	92.6	35.0
Visa Classic over 26			
network 2	70.0	83.7	15.0
network 3	73.2.	86.6	20.0
Visa Gold			
network 2	65.0	79.6	0.0
network 3	82.0	98.5	8.8

Analysis of results

- Visa Classic
  - this classification had been performed on the population of applications that had been previously assessed as good accounts by the existing credit card scoring system
  - the 35% of additionally identified bad accounts represent potential savings for the bank
- Visa Gold
  - the poor prediction of bad accounts for Visa Gold may be due to insufficient information on the application form
  - although Visa Gold targets the most affluent members of the society the application for the card carries the same information as the application for Visa Classic

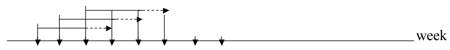
For more discussion see the article

Application to Sales Forecasting

- The problem is that of identifying the patterns in the past sales figures and understanding how various factors are influencing the formation of these patterns
- There are many factors influencing the sales of products
- The exact nature and degree of influence of these factors are hard to understand and predict
- Sales forecasting is usually done by the marketing managers and is often based on their past experience and intuition

A Neural Network Model for Sales Forecasting

- A Neural Network Model for Sales Forecasting developed for Tattersall's (Jagielska & Jacob, 1993)
- time series forecasting
- *sliding window* technique



Sliding Window size = 3

A Neural Network Model for Sales Forecasting

- In our study a neural network was built to predict sales figures
- The neural network was presented with past sales data and once it had learnt the patterns in the training data, the network was used to provide a sales forecast

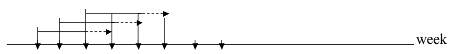
Factors of Influence

- The following factors were identified as having influence on the sales volume:

- ** Previous performance* - sales figure for the corresponding week of the previous year is considered the basis of the sales forecast for any week
- ** Draw type* - this includes super draw and special promotions such as Valentines day, Mothers day , Fathers day, Christmas Day etc.
- **Jackpot* - indicates the number of weeks the division 1 has jackpotted,
- **Division 1 prize*,
- **Cost of a single bet*
- ** Cannibalisation*- launching of new products, which could compete ** Economic indicators* - one or more suitable indicators may be considered such as CPI or Average Weekly Earnings (AWE)
- ** Advertising dollars* - total advertising dollars spent for the draw.

The Neural Network Model
Selection of input variables

- Not all of the influence factors were selected as input variables to the neural network.
- The ignored factors were considered by the marketing managers as having only a minor influence
- The selected input variables and their representation were:
  - *Past Sales* = (number) - sales in Million dollars; a sliding window of three immediately past draws each having the following parameters:



Sliding Window size = 3

The Neural Network Model
Selection of input variables

**Super Draw* = (1, 0) where 1 corresponds to "yes" for super draws

**Jackpot* - four parameters. In order to build-in the overlapping nature of these variables, it was decided to represent them as " the termometer" codes:

the 1st week jackpot was represented as (0,0,0,1), 2nd week as (0,0,1,1), 3rd as (0,1,1,1) and for more than 3 weeks jackpot the code is (1,1,1,1)

**Div1Prize* = (number) - prize amount in Millions

Average Weekly Earnings = (number), AWE figures available from Australian Bureau of Statistics were used.

The NN Architecture

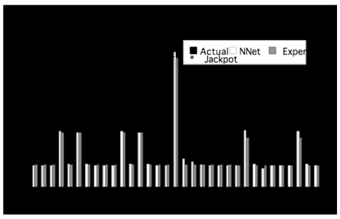
- The input layer consisted of 28 units
- The output layer consisted of one unit corresponding to the sales amount of the current draw.
- After a number of experiments with different number of hidden layers and units , a network with one hidden layer was selected

Training and Testing

- The training set consisted of data recorded between July 1989 and December 1991 (130 input/output training vector pairs - one pair for each week of this period)
- The trained network was then used to predict weekly sales volumes for the period January 1992 to September 1992 (33 vector pairs)

Analysis of Results

		Prediction Error		
Predictions	Correlation	Average	Super draw	Std Error
Neural Net	0.979	3.48%	12%	0.280
Experts	0.977	3.67%	15%	0.316



Limitations of neural networks

- In contrast to decision trees and rule extraction techniques the knowledge(patterns) “discovered” by neural networks is not represented in the form understandable by humans
- Knowledge in a trained neural network is encoded in its connection weights
- Thus NN cannot be used for descriptive data mining(exploration)
- If NN are used for decision making it is impossible to explain their decisions - often other techniques have to be combined with NN for explanation

Hybrid techniques

- Data mining tools often combine many different techniques
- For example, RightPoint’s DataCruncher(see Groth p149) combines neural network, statistical algorithms and some other techniques

Example : churn modelling

- Students are to complete exercise from Groth (149-181),
- Tool: DataCruncher
- Case study: churn modelling
- download and install from <ftp://ftp.prenhall.com/pub/ptr/c++_programming.w-050/groth/>

References

- Groth R (2000) Data Mining
- Neuralyst Training Manual, Cheshire Engineering Corporation, http://www.cheshireeng.com
- Berry, M. and Linoff, G. (2000) Data Mining Techniques for Marketing, Sales, and Customer Support, John Wiley & Sons

Berson A.., Smith S., Thearling K (2000), Building Data Mining Applications for CRM,McGrow-Hill

Jagielska I, Jaworski J.,(1996) *Neural Network for Predicting the Performance of Credit Card Accounts,* Computational Economics Vol 9, No. 1 February 1996, 77-82