Synapse
September 2003

Greetings from all of us at NeuralWare. Synapse is NeuralWare's newsletter with the latest information for friends and customers of NeuralWare. In this issue, we will cover the following:

- Application Story from a long time End User
- The 2003 training schedule including a new, never before offered course
- Tips from our Technical Support Group
- NeuralWare hosts and conducts training course for WISCO (Wuhan, China)
- Submit your own case study and receive credits towards purchases
- Other miscellaneous tidbits

Application Story from an End User

The following application story has been submitted by Abby Ilumoka, Ph.D, M. Sc from the University of Hartford. Dr. Ilumoka is a Professor of Electrical and Computer Engineering.

We wish to thank Dr. Ilumoka for providing the following endorsement of NeuralWare products and for the success story that follows.

"After nine years of very successful licensed access to NeuralWare's software at the University of Hartford, I would like to thank NeuralWare for giving us access to powerful tools for modeling, prediction and pattern recognition. My teaching and NSF-funded research work in the area of Microelectronics VLSI circuits have greatly benefited from the use of the Professional II/PLUS development system. In fact, many students, faculty and staff have expressed strong interest in expanding their use of neural net technology."

Application Story:

Robot Neurocontroller

This is an adaptive control project in which the predictive capability of neural networks is exploited. The objective was to design a neural network-controlled robot to patrol single level, 4-room house, detect fire location(s), set off fire alarm, turn on a fan and resume patrol. When all rooms have been visited, robot returns to start position and cycle repeats. Engineering & scientific literature report several examples of previous work in this area including the use of an associative neural network to train and control a fire fighting robot, puck state prediction in robotic air hockey and flexible link control in industrial robots using radial basis function networks. In this project, the approach taken was to use the multiplayer perceptron architecture with back propagation learning. The network used four previous and the current robot position to predict the next robot position. Additional inputs included the door positions in the house, a room counter and the fire sensor and fan status indicators. Using a neural network with 40 inputs, 25 nodes in the hidden layer and 8 outputs, the neural net was able to have the robot explore each of the 4 rooms without wall collisions, detect and extinguish each of 16 fires with good accuracy. The best results were obtained with sigmoid activation function and delta learning rule with a momentum factor of 0.92

and a learning rate of 0.3. RMS error for test was 0.0114. Project execution was in two phases – house patrol and fire detection:

Phase 1: House Patrol

Robot learns to walk around house a preset distance from walls (this avoids collision with objects). Inputs include prior and current robot positions, door location, and current room count. Outputs include next robot position, next room count. Task is to select neural net architecture, learning rule, learning parameters. Create database to train net. Test net.

Phase 2: Fire Detection

Robot sensors activated to detect temp rise as robot patrols house. Additional inputs include current sensor status, current fan status. Additional outputs include next sensor status, next fan status. Extend database to train robot to react correctly in a variety of fire situations net. To test net, allow user to override/interrupt normal flow by injection of sensor/fan stimuli.

A general purpose programming language PC Logo for Windows was employed as a graphic display tool for the project. PC Logo allowed the construction of the house, the positioning of fires and the display of robot movement in the house.

Future work should include the investigation of robot control in a dynamic environment featuring other objects in motion (e.g. falling object). It would also be interesting to exploit the parallelism available in modular neural networks to see if better results can be achieved.

Submitting an Article

If you wish to submit a case study, application note, or technical tip for publication, please contact NeuralWare sales. If we publish your contribution, you will receive a one-year Technical Assistance Program (TAP) subscription for the NeuralWare program that you use - a minimum $375 value. If you prefer, you may also apply the credit towards NeuralWare products for training! We hope you will consider this exciting opportunity.

Several of NeuralWare's customers have taken advantage of this offer. See the success story in this issue by Abby Ilumoka of the University of Hartford and watch for other customer stories in upcoming issues.

WISCO training

NeuralWare is pleased to have hosted a training session in August, 2003 for WISCO-ICC, a major steel producing corporation in Wuhan, China. Six participants traveled to our facility from Wuhan for two weeks to complete training that had begun in Wuhan earlier this year. During the first week, Dr. Anthony DeArdo and Dr. Isaac Garcia, professors at the University of Pittsburgh, instructed the engineers in using neural network technology as it applies to steelmaking. For the second week NeuralWare instructors provided advanced neural network training and continued discussions with the engineers and NeuralWare's CEO, Jack Copper, on the joint project between NeuralWare and WISCO to develop a steel production simulator. The two companies plan to continue working closely together along with the United States

Commercial Service in China to introduce neural network technology to other Chinese enterprises.

Training

NeuralWare is pleased to announce the release of our new training course, Developing Neural Network Applications. This three-day neural computing course focuses on the practical aspects of creating and deploying neural network applications using NeuralWare development tools. It provides an overview of the NeuralWare product line, and describes the software architecture of each application and library. It also identifies the functional and architecture relationship among the products. Next, the course presents an overview of the methods that can be used to deploy neural network models, including standard NeuralWare .nnd and .npr files, FlashCode™, Designer Pack, and use of NeuralWare library components. Additional information and a syllabus are available on our website at www.neuralware.com/training.jsp.

Register by October 15 for a NeuralWare course from the November –January series and receive a 10% early-bird discount. Courses are taught in Pittsburgh at our conveniently located headquarters just a few miles from downtown Pittsburgh.

Our regularly scheduled training courses ensure that our customers are knowledgeable about neural network technology and using our software effectively. See the schedule below to select the course that best fits your schedule. Make sure to visit our website www.neuralware.com/training.jsp or talk with our sales department to get complete course information before making your final selection.

Class size is limited to offer participants personalized instruction when necessary. A well-organized yet informal structure allows flexibility and provides opportunities for real-world information sharing and problem solving. When you visit the training website, look for comments from previous attendees.

If you have several individuals within your company that would like to attend our course, let us know and we will be happy to provide information on our on-site training programs. Contact us via email at training@neuralware.com.

If you would like to expand your knowledge of neural network technology using NeuralWare's state-of-the-art tools, plan to attend a course soon. Don't forget the 10% early-bird discount. Contact us today!

Current training schedule

Applying Neural Networks to Business, Industry and Government (4.5 days)
Nov 10-14
Jan 26-30

Developing Neural Network Applications (3 days)
Oct 20-22

Here are some comments from our August, 2003 Applying Neural Networks to Business, Industry and Government course.

"I was initially intimidated by the concept of neural networks. I walked away from this course with an excellent grasp of the fundamentals and a vision for a multitude of opportunities."

"The course was very practical; the instructors were very knowledgeable and everyone at the facility was kind, friendly and helpful."

Tips from our Technical Support Group

What is Case Based Reasoning and how can I use it?

Overview

Case Based Reasoning (CBR) is described in its own chapter of the NeuralWorks Predict User Guide. This paper is meant to highlight the usefulness of CBR, and show in a simplistic way how CBR works.

In some sense a trained CBR model is similar to a trained neural network model; both are created from one or more data sets, and both are capable of producing a predicted or forecasted output.

CBR and neural networks differ greatly in their approach to creating a model. While neural networks "learn" a problem by using a data sets target output, CBR creates a model using input data only. CBR groups similar records together into a 'case' (a grouping), and then each case can report statistics such as the average target output of all records in that case. Note that CBR model creation (the grouping of the data records into various cases) is accomplished without using target output data.

CBR models work best when dealing with few input fields. In Predict, CBR models use the internal input data, after Predict runs its data transformation and variable selection components. This means that if you present 100 data fields to Predict, the variable selection component may select only 15 or so variables to be part of the neural network model. Predict CBR models are created using the same 15 or so variables. CBR works best with as few variables as possible; 5 variables are much better than 10; 15 or more variables may produce poor CBR models. This is explained in more detail later.

Usefulness

Neural network models typically yield better predictions or forecasts than a CBR model. So what then is the value of CBR? The output of CBR can provide insight into the confidence of the neural network's output.

Consider a financial model that daily scans thousands of stocks and predicts 0.0 through 1.0 for each stock, where 1 represents a strong likelihood that a stock price will increase tomorrow. Suppose that the top two strongest stocks are selected, and their predicted outputs are 0.97 and 0.94. On the surface the stock whose prediction is 0.97 looks like the best stock to purchase. The same two stock records can now be run through a CBR model to determine the number of records similar to each record. CBR will produce several outputs, including the number of records in each case that these two records fall into. One might see that the 0.97 prediction was produced from a unique case; the CBR model has never seen another record similar to the stock data which generated the 0.97 prediction. The CBR model might also indicate that there are 100 records similar to the stock data which predicted 0.94. Given this additional information, there might be less risk involved with purchasing the second-best stock.

Neural network models produce predictions or forecasts. CBR models augment this information by indicating the average output of, and the number of records similar to a given record.

How CBR Works

An illustration of how CBR functions can be made using the following 10 record data set. Each of the three input variables range from 1 to 100.

input1	input2	input3	target
1	100	1	1
10	80	100	2
30	60	1	3
50	40	100	4
70	20	1	5
90	10	100	6
100	1	1	7
2	95	5	8
20	81	20	9
35	50	20	10

CBR scans each input column individually, breaking each data range into 5 bins (5 is the default and is controlled by the Granularity Factor setting in Predict's parameters). Each input ranges from 1 to 100 in this sample data, but real-world problems usually have inputs with different ranges.

A range of 100 break into 5 bins as shown below. Each bin is labeled from A through E.

```
 ------- -------- -------- -------- --------
|  a  |   b  |   c  |   d  |   e  |
|     |      |      |      |      |
| 1..20 | 21..40 | 41..60 | 61..80 | 81..100|
 ------- -------- -------- -------- --------
```

The numeric values are temporarily replaced by the appropriate bin label, yielding:

input1	input2	input3	target
a	e	a	1
a	d	e	2
b	c	a	3
c	b	e	4
d	a	a	5
e	a	e	6
e	a	a	7
a	e	a	8

a	e	a	9
b	c	a	10

This data is then sorted so that similar input records group together into cases:

input1	input2	input3	target	
a	d	e	2	!case #0 = 'ade'
a	e	a	1	!case #1 = 'aea'
a	e	a	8	!case #1 = 'aea'
a	e	a	9	!case #1 = 'aea'
b	c	a	3	!case #2 = 'bca'
b	c	a	10	!case #2 = 'bca'
c	b	e	4	!case #3 = 'cbe'
d	a	a	5	!case #4 = 'daa'
e	a	a	7	!case #5 = 'eaa'
e	a	e	6	!case #6 = 'eae'

The result of this sample data through CBR is 7 distinct cases which were created from the original 10 records. There are 3 'aea' records grouped together into a case, two 'bca' records in another case, and all other records are assigned to different cases.

CBR statistics can now be generated on a per-case basis. It is trivial to calculate, for each case, the count (number of records in the case) and the average, minimum and maximum target output value of records in each case. A CBR/Run operation runs data through the trained CBR model and outputs the count, average, minimum and maximum of the case which a record falls into. A unique case number is also output should you wish to inspect the records which were assigned to a given CBR case.

Finally, the following new data records are run through the CBR model. These are records that are not part of the sample data used to create the CBR model. Note that target outputs do not exist in this data. The resulting CBR Output shown below can be determined by inspecting the above 7 CBR cases.

new input values ->	CBR binning ->	CBR bin matching ->	CBR Output count, avg, min, max, id
18 81 5	a e a	aea	3 6 1 9 1
18 82 5	a e a	aea	3 6 1 9 1
93 14 8	e a a	eaa	1 7 7 7 5
100 100 100	e e e	no match	0 0 0 0 -1
35 50 2	b c a	bca	2 6.5 3 10 2

The reason that CBR usefulness diminishes as the number of input variables increase is that when there are so many components (dimensions) in the input space, it becomes more likely that every record is unique. Below are two records after converting them into bin labels. Although they look similar, these records will be assigned to different CBR cases.

abcdeaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
abcdeaaaaaaaaaaaaaaaaaabaaaaaaaaaaaaaaaaaaaaaaaa

Odds are very good that every record will be assigned its own CBR case if you have 1,000 data records of 50 inputs. In such a data set there are 5 to the 50th power potential cases, where 5 = granularity and 50 = number of inputs. This is far, far more possible cases than there are humans on earth. Unless you are working with an evenly-dispersed data set of over 8 [gazillion] data cases, there are major 'holes' in the 50 dimensional space.

If most records are unique, then this completely eliminates any value from per-case statistics. Of course, if Predict is able to reduce many inputs down to a few relevant variables, then CBR models as well as neural network models will not be hindered by using too many variables.

Personal License Extension Information

If you are someone who needs to use your NeuralWare software both at home and at work, you are a candidate for NeuralWare's personal license extension (PLE). With a PLE you can request a second license key for your home or laptop computer for the low one-time cost of $250. Only users who are on the technical assistance program (TAP) are eligible for this benefit. To take advantage of the TAP/PLE program, contact our sales department for more information at sales@neuralware.com.

Canceling your subscription

If you wish to discontinue your subscription to the newsletter, send email to newsletter@neuralware.com and ask to be removed from the distribution list. Please insert the word "remove" in the subject line.

Additional Information

Additional information on NeuralWare is available at www.neuralware.com! We hope you will visit us!

About this eLetter

This eLetter is copyright 2003 NeuralWare. All rights reserved. This eLetter may be freely distributed as long as the above copyright notice is included in any redistribution, electronic or printed.