Prediction of Commodities in Rationing System Using an Enhanced Regression Neural Network Algorithm

Capt. Dr. S Santhosh Baboo [#1], Ms. P ShanmugaPriya [*2]

[#] Department of Computer Science and Application, Dwaraka Doss Goverdhan Doss Vaishnav College
Arumbakkam, Chennai, Tamil Nadu – 600 106, India
[1] santhos2001@sify.com

[*] Department of Computer Science and Application, Dwaraka Doss Goverdhan Doss Vaishnav College
Arumbakkam, Chennai, Tamil Nadu – 600 106, India
[2] shanmusekar@gmail.com

Abstract— Predictive analytics is an area of data mining that deals with extracting information from data and using it to predict trends and behaviour patterns. The paper predicts the usage of the food commodities in public distribution system in the coming years using the general regression neural network algorithm. The algorithm is enhanced using the NTP algorithm which trains the data as per the requirements of ration system in Tamilnadu. A memory-based network that provides estimates of continuous variables and converges to the underlying (linear or nonlinear) regression surface. This general regression neural network (GRNN) is a one-pass learning algorithm with a highly parallel structure. Even with sparse data in a multidimensional measurement space, the algorithm provides smooth transitions from one observed value to another.

Keyword- NTP, NNT, PDS2, PGR, AAY

I. Introduction

The Public Distribution System in India is 50 years old. At present it is being carried on as an anti-inflationary and antipoverty system. Tamil Nadu, the southernmost State in the country, is adopting the Universal Public Distribution System covering its entire population and supplying regularly rice, wheat, sugar, kerosene and other products like pulses, edible oil etc. The PDS is a centrally sponsored scheme that entitles beneficiaries to subsidised food grains every month.

The widespread popularity of multilayer feed forward networks in many fields is mainly due to their ability 1) to approximate complex unknown nonlinear mappings directly from the input training samples and 2) to form disjoint decision regions with arbitrary shapes and to determine unknown classes. A neural network system is called a real-time learning system if it can finish a learning procedure with good generalization performance for a new application within expected fast response time defined by external requirements. Real-time learning capability of neural networks is highly expected whenever a new application is faced, where a new knowledge map has to be built.

Classical statistical methods have been applied in industry for years. Recently, Neural Network (NNs) methods have become tools of choice for a wide variety of applications across many disciplines. It has been recognized in the literature that regression and neural network methods have become competing model-building methods (Smith & Mason, 1997). For a large class of pattern-recognition processes, NNs is the preferred technique (Setyawati, Sahirman, & Creese, 2002). NNs methods have also been used in the areas of prediction and classification (Warner &Misra, 1996). Statistical methods such as regression analysis, multi-variate analysis, Bayesian theory, pattern recognition and least square approximation models have been applied to a wide range of decisions in many disciplines (Buntine &Weigend, 1991). These models are attractive to decision makers because of their established methodology, long history of application, availability of software and deep-rooted acceptance among practitioners and academicians alike. NNs are data dependent and therefore, their performance improves with sample size. Statistical methods, such as Regression perform better for extremely small sample size, and also when theory or experience indicates an underlying relationship between dependent and predictor variables (Warner &Misra, 1996).

Statistical methods such as regression analysis, multi-variate analysis, Bayesian theory, pattern recognition and least square approximation models have been applied to a wide range of decisions in many disciplines (Buntine & Weigend, 1991). These models are attractive to decision makers because of their established methodology, long history of application, availability of software and deep-rooted acceptance among practitioners and academicians alike. NNs are data dependent and therefore, their performance improves with sample size. Statistical methods, such as Regression perform better for extremely small sample size, and also when theory or experience indicates an underlying relationship between dependent and predictor variables (Warner &Misra, 1996).

II. NEURAL NETWORK MODELS

An overview of NNs models is provided by Lippmann (1987). Fausett (1994); Freeman (1994); Hertz, Krogh, and Palmer (1991); Lawrence (1994); Mehrotra, Mohan, and Ranka (1996); Rumelhart, Hinton, and Williams (1986); Smith (1993); Taylor (1999) and White (1992) conducted research involving mathematical description of NNs models, Neural Net architecture, training algorithms such as supervised/unsupervised learning and back propagation algorithm. It is also evident from the literature involving NNs models that the development and application of NNs is not limited to a specific area. The applications of NNs range from signal processing in telecommunications to pattern recognition (Lippmann, 1987) in Medicine, Business and Engineering. The following section provides a brief over-view of the articles that applied Neural Networks in various disciplines Health, Financial Marketing, Data Mining, Business, Manufacturing and Engineering.

III.COMPARISON OF STATISTICAL AND NN MODELS

Numerous authors have compared performance of statistical and neural networks models on specific problems (Sarle, 1994; Schumacher, Robner, &Vach 1999; Wu & Yen, 1992). Hruschka (1993), for example, compared econometric techniques with NNs models applied in market response functions. The authors indicated that the back-propagation NNs model that they used might lead to better model fits than achieved by comparable econometric models. However, they also cautioned that more studies were necessary to establish general conclusion regarding the strengths and weaknesses of neural networks. Lee and Jung (2000) compared the forecasting ability of logistic regression analysis with that of NNs model to predict creditworthiness of urban customers. Werbos (1991) discussed the link between Artificial Neural Networks (ANNs) and Statistical models in pattern Recognition. Warner and Misra (1996) compared the performance of regression analysis with that of neural networks using simulated data from known functions and also using real-world data. The authors discussed the situations where it would be advantageous to use NNs models in place of a parametric regression model. They also discussed difficulties related to implementation of NNs models. Zahedi (1996) compared predictability of NNs with that of discriminant analysis in financial applications. Ainslie and Dreze (1996) compared predictability of logistic regression with that of NN models.

IV.TRAINING GENERAL REGRESSION NEURAL NETWORK

Before getting into training process some initialization has to be made as a preliminary process like deciding what has to be the input and what has to be the target. In our case we are going to train the neural network to do the prediction based on the requirement of the commodities.

The consumption of food products in Public Distribution System is predicted based on the real time ration sales data collected from the Co-operative Society under the Public Distribution System2 (PDS2) department, of Medavakkam, Chennai, Tamilnadu. The data consists of requirements, allotments, movements and sales of commodities like rice, sugar, kerosene etc for 25 shops of Nanganallur circle with the total number of cards for each shop with rice cards, sugar cards, Antodaya Anna Yojana (AAY) cards given for the poorest of the poor. The actual real time data has been taken for a period of one year from May 2012 to April 2013.

If number of rice buyers in particular ration shop (or) number of ration cards eligible to get rice is *’x’* then the number of cards eligible to get Sugar ‘*y*’ will be *’x/a’* and number of AAY cards ‘*z*’ will be *’x/b’*, where constant *a=0.8962* and *b=58.21053*.

To compare the predicted results to find the deviation, we have to make the actual data ready. So from the constant study of real time data we have made approximate linear equations as follows.

For one Card the requirement of Rice will be *c=19.12511* kg per month
Rice Commodities= *N*c;*
Where, *N*→ Number of cards and *c*→ is a constant, *19.12511* Kg Rice per card.

1. For one card the requirement of Sugar will be *d=2.026138* kg per month
   Sugar Commodities= *N*d*; Where, *N*→ Number of cards and *c*→ is a constant, *2.02612* kg sugar per card.
2. For one Card the requirement of AAY will be *e=35* kg per month
   AAY Commodities= *N*e*;
   Where, N→ Number of cards and e→ is a constant, *35* Kg AAY per card.

Fig. 1. Methodology for NTP

By means of this the comparisons are made.

A. Initialization

In this process our input is going to be the number of eligible cards to buy Rice and Population Growth Rate. Target is going to be the commodity requirements of Rice, Sugar and AAY.

Also the neural networks have several parameter in which user can modify accordingly such as Transfer Function, Training function, Performance analyser. The number of neurons assigned is also one of the parameter in which user has to define, using the formula.

$$\frac{1}{2}(\text{No. of Input Layers} + \text{No. of Output Layers}) + \sqrt{\text{No. of Input Samples}}$$

The process of the transfer function used was Tan Sigmoid, the training algorithm used was Levenberg Marquardt and the performance analyser used was MSE (Mean Square Error)

B. Neural Training Preprocessor

Preprocessing the training data will often improve the learning time of NN. This Neural Training Preprocessor (NTP) algorithm will automatically prepare the input, target and number of Neurons to train the entire Artificial Neural Network. This NTP adjuster algorithm will have different formation for different types of ANN and different types of data and the initialization needed for NTP operations constitute the algorithm.

Initializing Input ➔ x
Initializing Target ➔ t
a = Ratio of sugar cards to the rice cards ‘x’.
b = Ratio of AAY cards to the rice Cards ‘x’.
c = Weight of rice per card
d = Weight of Sugar per card
e = Weight of AAY per card

$$y = \frac{x}{a}$$

$$z = \frac{x}{b}$$

RC = N x C
SC = N x d
AC = N x e

Fig. 2. NTP Algorithm

The algorithm uses X for initializing the input, T for initializing target and the adjustment factors such as y which represents number of cards eligible to get sugar, z is number of cards eligible to get AAY, RC represents the Rice Commodities, SC for Sugar commodities, AC for AAY commodities and N represents the number of cards. From this initial calculation input and targets are set

Input=[x; y; z]

Target= [RC; SC; AC]

$$\text{No. of Neurons} = \frac{1}{2}(\text{No. of Input Layers} + \text{No. of Output Layers}) + \sqrt{\text{No. of Input Samples}}$$

These are the preliminary stages given to Neural Network for training so that the prediction accuracy is good.

C. Neural Network Testing

While testing the neural network the inputs are given such as Number of cards eligible to get Rice and Population growth rate (PGR).

TABLE I
Population Growth Rate

YEAR	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010
PERCENTAGE	1.7	1.6	1.6	1.6	1.5	1.5	1.4	1.4	1.4	1.4	1.4

By means of this the eligibility card ratio for Sugar and AAY was also calculated. These card numbers was increased according to the PGR and given to Trained Network for prediction. The trained network will predict the Rice, Sugar and AAY requirement commodities for a month. To convert these commodities for year wise requirement it will be multiplied by 12.It has been found from the table 1 that the PGR from 2000 to 2011 was as follows and the population growth rate has become constant as 1.4% from 2006 onwards.

D. Dataset Used

The dataset has been obtained from the government co-operative Society of Public Distribution System 2(PDS2) department of Medavakkam, Chennai, Tamilnadu. The algorithm used is to be designed for predicting the sales data. The data contains the sales values of more or less 25 shops during the year of 2012-2013.

TABLE III
Regression Neural Network

Years	Rice Cards	Predicted Rice	Actual Rice	Sugar Cards	Predicted Sugar	Actual Sugar	AAY Cards	Predicted AAY	Actual AAY
2009	1110	21230	21235	1239	2510	2510	19	664	674
2010	1126	21528	21532	1256	2545	2545	20	667	683
2011	1142	21753	21833	1274	2571	2581	20	681	693
2012	1158	22485	22139	1292	2597	2617	20	697	703
2013	1174	22968	22449	1310	2715	2654	20	700	712

TABLE IIIII
Feed Forward Back Propagation

Years	Rice Cards	Predicted Rice	Actual Rice	Sugar Cards	Predicted Sugar	Actual Sugar	AAY Cards	Predicted AAY	Actual AAY
2009	1110	22035	21235	1239	2683	2510	19	697	674
2010	1126	22056	21532	1256	2613	2545	20	697	683
2011	1142	22088	21833	1274	2598	2581	20	698	693
2012	1158	22145	22139	1292	2621	2617	20	701	703
2013	1174	22249	22449	1310	2665	2654	20	707	712

TABLE IVV
Radial Basis

Years	Rice Cards	Predicted Rice	Actual Rice	Sugar Cards	Predicted Sugar	Actual Sugar	AAY Cards	Predicted AAY	Actual AAY
2009	1110	23786	21235	1239	2806	2510	19	405	674
2010	1126	23786	21532	1256	2806	2545	20	9	683
2011	1142	23786	21833	1274	2806	2581	20	488	693
2012	1158	23786	22139	1292	2806	2617	20	1429	703
2013	1174	23786	22449	1310	2809	2654	20	295	712

After training the Neural network with the NTP algorithm, the GRNN is ready to process the prediction. Three parameters the number of cards, year and population growth rate are considered as important and the values are predicted by varying one of the parameter and keeping the others as constant.

The above tables II, III and IV shows the actual and calculated predicted values of rice, sugar and AAY rice for the years 2009 to 2013 using Regression Neural Network, Feed forward Back Propagation and Radial Basis algorithms.

Here a line graph is used to show a comparison between predicted rice and actual rice. The various years are represented on x axis and y axis with lines representing the quantity of rice in kg. The x-axis is a horizontal line representing five different years 2009, 2010, 2011, 2012 and 2013 and y-axis is a vertical line representing the various quantities of commodities in kilogram. The *x*-axis and *y*-axis are simply just two intersections of different years and quantities of various commodities (Rice, Sugar, AAY) in kilograms. The line marked in blue colour represents the predicted amount of commodities in kilograms and brown colour indicates actual amount of commodities in kilograms. This is similar for sugar and AAY rice as shown in Fig. 2 and Fig. 3.

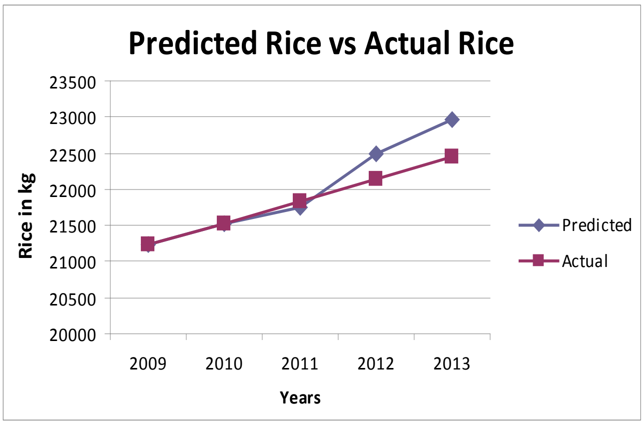

Fig. 3. Regression Neural Network

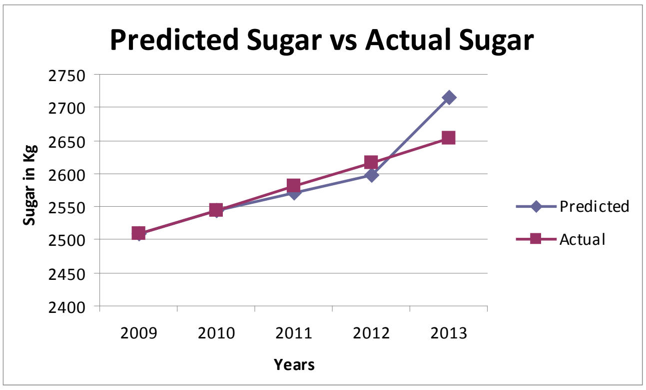

Fig. 4. Feed Forward Back Propagation Neural Network

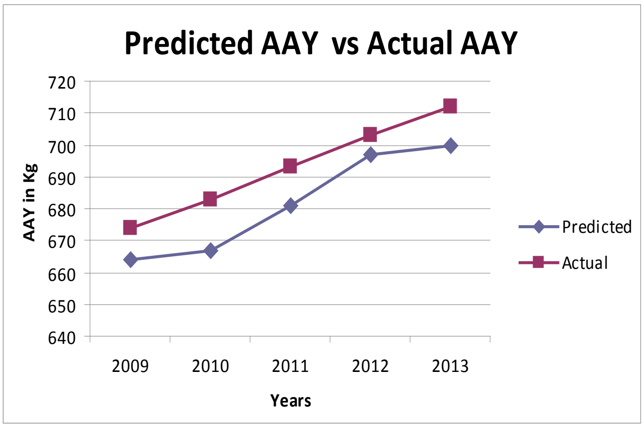

Fig. 5. Radial Basis

V. CONCLUSION

The tables and figure shows the actual and the calculated values for rice, sugar and AAY rice for the years 2009 to 2013. The calculated and the predicted values are obtained using the regression neural network, Feed Forward Back Propagation and Radial Basis algorithms. The general regression neural network graph shows clearly the deviation as less as possible when compared with other traditional Feed Forward Back Propagation and Radial Basis algorithms. The actual database will check the values with the predicted values with less deviation and less error rate.

REFERENCES

[1] B.Syam Prasad Reddy, K.VagdhanKumar,B.Musala Reddy, N.RajaNayak, “ANN Approach for Weather Prediction using Back Propagation”, International Journal of Engineering Trends and Technology ISSN: 2231-5381 Vol 3 (1), 2012.

[2] A. Pavelka and A. Proch_azka, “Linear and Non-Linear Models for Signal Prediction”, Institute of Chemical Technology, Department of Computing and Control Engineering.

[3] Makarand A Kulkarni, Sunil Patil, G V Rama and P N Sen, “Wind speed prediction using statistical regression and neural network” J. Earth Syst. Sci. Vol 117 (4), pp. 457–463 August 2008.

[4] Christian L. Dunis and JamshidbekJalilov, “Neural Network Regression and Alternative Forecasting Techniques for Predicting Financial Variables”, (Liverpool Business School and CIBEF April 2001.

[5] Guang-Bin Huang, Qin-Yu Zhu, and Chee-KheongSiew, “Real-Time Learning Capability of Neural Networks”, IEEE Transactions on Neural Networks Vol17 (4), July 2006.

[6] Muhammad A. Razi, KuriakoseAthappilly,“A comparative predictive analysis of neural networks (NNs), nonlinear regression and classification and regression tree models”, Expert Systems with Applications vol 29 pp65–74, 2005.

[7] G.-B. Huang, “Learning capability and storage capacity of two-hiddenlayerfeedforward networks,” IEEE Trans. Neural Netw., vol. 14 (2), pp. 274–281, March 2003.

[8] K. Hornik, M. Stinchcombe, and H. White, “Multilayer feedforwardnetworks are universal approximators,” Neural Netw., vol. 2, pp. 359–366, 1989.

[9] G. Cybenko, “Approximation by superpositions of a sigmoidal function,”Math. Control, Signals Syst., vol. 2, no. 4, pp. 303–314, 1989

[10] K. Funahashi, “On the approximate realization of continuous mappings by neural networks,” Neural Netw., vol. 2, pp. 183–192, 1989.

[11] A. Gallant and H. White, “There exists a neural network that does not make avoidable mistakes,” in Artificial Neural Networks: Approximation and Learning Theory, H. White, Ed. Oxford, U.K.: Blackwell, pp. 5–11 1992.

[12] M. Stinchcombe and H. White, “Universal approximation using feedforward networks with non-sigmoid hidden layer activation functions,” in Artificial Neural Networks: Approximation and Learning Theory, H. White, Ed. Oxford, U.K.: Blackwell, 1992, pp. 29–40.

[13] Y. Ito, “Approximation of continuous functions on R by linear combinations of shifted rotations of a sigmoid function with and without scaling,” Neural Netw., vol. 5, pp. 105–115, 1992.

[14] T. Chen, H. Chen, and R.-W. Liu, “Approximation capability in C(R)by multilayer feedforward networks and related problems,” IEEE Trans. Neural Netw., vol. 6, no. 1, pp. 25–30, Jan. 1995.

[15] G.-B. Huang, Y.-Q. Chen, and H. A. Babri, “Classification ability of single hidden layer feedforward neural networks,” IEEE Trans. Neural Netw., vol. 11, no. 3, pp. 799–801, May 2000.

[16] D. Erdogmus, O. Fontenla-Romero, J. C. Principe, A. Alonso-Betanzos, and E. Castillo, “Linear-least-squares initialization of multilayer perceptrons through backpropagation of the desired response,” IEEE Trans. Neural Netw., vol. 16, no. 2, pp. 325–337, Mar. 2005.

[17] D. E. Rumellhart, G. E. Hinton, and R. J.Williams, “Learning internal representations by error propagation,” Parallel Distrib. Process., vol. 1, pp. 318–362, 1986.

[18] R. P. Brent, “Fast training algorithms for multilayer neural nets,” IEEE Trans. Neural Netw., vol. 2, no. 3, pp. 346–354, May 1991.

[19] R. S. Scalero and N. Tepedelenlioglu, “A fast new algorithm for training feedforward neural networks,” IEEE Trans. Signal Process., vol. 40, no. 1, pp. 202–210, Jan. 1992.

[20] A. Sperduti and A. Starita, “Speed up learning and network optimization with extended back propagation,” Neural Netw., vol. 6, pp. 365–383, 1993.

[21] A. Parlos, B. Fernandez, A. Atiya, J. Muthusami, and W. Tsai, “An accelerating learning algorithm for multilayer perceptron networks,” IEEE Trans. Neural Netw., vol. 5, no. 3, pp. 493–497, May 1994.

[22] X. H. Yu, G. A. Chen, and S. X. Cheng, “Dynamic learning rate optimization of the backpropagation algorithm,” IEEE Trans. NeuralNetw., vol. 6, no. 3, pp. 669–677, May 1995.

[23] M. T. Hagan, H. B. Demuth, and M. H. Beale, Neural Network Design. Boston, MA: PWS, 1996.

[24] M. T. Hagan and M. B. Menhaj, “Training feedforward networks with the Marquardt algorithm,” IEEE Trans. Neural Netw., vol. 5, no. 6, pp. 989–993, Nov. 1994.

[25] S. Abid, F. Fnaiech, and M. Najim, “A fast feedforward training algorithm using a modified form of the standard backpropagation algorithm,” IEEE Trans. Neural Netw., vol. 12, no. 2, pp. 424–430, March 2001.