Artificial Neural Network Prediction of Amino Acid Levels In Feed Ingredients

WILLIAM B. ROUSH and TERRI L. CRAVENER

Department of Poultry Science, Pennsylvania State University, University Park, Pennsylvania 16801

ABSTRACT Artificial Neural Networks (ANN), which are biologically inspired tools, serve as an alternative to regression analysis for complex data. Based on CP or proximate analysis (PA) of ingredients, two types of ANN and linear regression (LR) were evaluated for predicting amino acid levels in corn, wheat, soybean meal, meat and bone meal, and fish meal. The two ANN were a three layer Backpropagation network (BP3), and a General Regression Neural Network (GRNN). Methionine, TSAA, Lys, Thr, Tyr, Trp, and Arg were evaluated and R^2 values calculated for each prediction method. Artificial neural network training was completed with NeuroShell 2® using Calibration to prevent overtraining. Ninety percent of the data were used as the input for the LR and the two ANN. The remaining 10% (randomly extracted data) were used to calibrate the performance of the ANN. As compared to LR, the R^2 values were largest when PA input and GRNN were used. The BP3 did not consistently improve the R^2 values for either CP or PA inputs as compared to LR. Each neural net can be incorporated into a computer or spreadsheet program.

(*Key words*: amino acid prediction, artificial neural networks, information processing)

1997 Poultry Science 76:721–727

INTRODUCTION

Amino acid determination is expensive due to chemical analysis and laboratory turnover time required for analysis. The expense in time and money has prompted a search for alternatives to the chemical analysis of AA in feed ingredients. Although AA levels do change with CP levels, this change is not proportional (Ward, 1989) and necessitates relating inputs to outputs in a quantitative manner. Two quantitative methods of predicting AA levels have been developed using linear regression (LR) with an input of either CP (Degussa Corporation, 1990) or proximate analysis (PA) (Monsanto, 1986a,b,c).

The National Research Council (NRC, 1994) has published the CP (NRC, 1994:71, Table 9-4) and PA (NRC, 1994:72, Table 9-5) regression approaches to predicting AA. To estimate the AA level of a feed ingredient ($\hat{y}$) using the CP method, the following equation is used:

$$\hat{y} = a + bx$$

where the level of CP found in the ingredient sample analyzed is x; a is the intercept; and b the regression coefficient (slope). To use the PA (i.e., CP, moisture, fat, fiber, and ash) method of AA estimation, the following equation is used:

$$\begin{aligned}\hat{y} = {} & \text{intercept} + b_1\,(\text{percentage protein}) \\ & + b_2\,(\text{percentage moisture}) \\ & + b_3\,(\text{percentage fat}) \\ & + b_4\,(\text{percentage fiber}) \\ & + b_5\,(\text{percentage ash})\end{aligned}$$

where b_i represents the regression coefficients.

The Degussa and Monsanto AA prediction equations referenced in the NRC (1994) publication have divergent and, in some cases, low (< 0.50) R^2 values. As the R^2 value reflects the amount of input variability explained by the equation, it is clear that a more definitive method of AA prediction would be desirable.

Artificial Neural Networks (ANN) may more effectively reflect the complex relationship between inputs (ingredient composition) and outputs (nutrient level). An ANN does not require an *a priori* equation or model. This allows the ANN to build relationships between the variables by considering every interaction between the variables. Neural network predictions usually result in a tighter fit of the data than is accomplished with regression analysis. As a result, the tighter fit usually leads to better predictions (Ward Systems Group, 1993).

Successful ANN applications have been found for many areas, including medical diagnosis, stock market predictions, price forecasts, quality process control, robotics, and water resources (Ward Systems Group, 1993; Baxt, 1995). Several studies have been conducted to examine the potential for ANN use in the agricultural sciences [e.g., fertility detection, Das and Evans (1992); produce inspection, Deck *et al.* (1992); disease detection,

Received for publication August 15, 1996.
Accepted for publication December 30, 1996.

Roush *et al.* (1996); climate control, Seginer *et al.* (1994); crop yield prediction, Uhrig *et al.* (1992); product shelf life, Vallejo-Cordoba *et al.* (1995); and the grading of beef quality, Whittaker *et al.* (1991)]. The objective of this study was to evaluate the potential of ANN to predict AA levels in feed ingredients based on either CP or PA.

MATERIALS AND METHODS

Three methods of predicting AA levels in feed ingredients were evaluated based on CP and PA analysis of ingredients. The first method used was LR. The second and third approaches involved the use of ANN-Backpropagation (BP3) and General regression neural network (GRNN) algorithms. A brief description of the concept of ANN is presented in the Appendix. For a more in-depth introduction to ANN, see Caudill (1993), Caudill and Butler (1993), Lawrence (1993), and Widman *et al.* (1989).

Data

Ingredient data bases for corn, soybean meal, meat and bone meal, fish meal, and wheat with chemical analysis of CP, moisture, fat, fiber, ash, and AA levels were provided by Novus International.[1] Amino acids examined included Met, TSAA, Lys, Trp, Thr, Tyr, and Arg.

In this study, the input variables were either CP alone or PA (CP, moisture, fat, fiber, and ash). Each AA level was the output variable. A summary of the inputs and outputs are shown in Table 1.

Data bases (Lotus® 1-2-3® spreadsheets) were imported into NeuroShell 2® for ANN training with a Batch Processor Option (Release 2). The data bases were partitioned into learning sets (90%) and randomly extracted calibration sets (10%). The default value of zero (0) was used for the random seed number in the calibration set extraction. For ANN efficiency, both the BP3 and GRNN input and output data were normalized by using a linear scale function (0–1).

Linear Regression

Linear regression was used as a comparison control for the ANN procedures. Linear regression results were obtained by using prediction equations developed with CP or PA input (NRC, 1994) using SAS® Proc GLM (SAS Institute, 1990). The data base used for the linear regression method was the same as the 90% learning set used with the ANN.

Backpropagation Neural Network

A three-layer backpropagation ANN (BP3) was the first network used in this study. The initial weights for the linkages between neurons were random values between +0.3 and –0.3. The learning rate and momentum were set at 0.1 for all AA ANN. The following network training parameters were also set: rotation pattern selection, weight updates at momentum, and the events since minimum average error equaled 20,000.

The output of the ANN (AA content) was iteratively compared to the output values provided in the data base. For each pass through the data, the connecting weights of the ANN were adjusted in the direction of the data base output value. Eventually a stable set of weights evolved, providing outputs for the AA predictions (Forsström and Dalton, 1995).

To prevent overtraining, the calibration was set at 200 test intervals to evaluate the error between learning sets and independent calibration sets during the training period. At the point the error between the learning set and the calibration set was at a minimum, the ANN was saved.

General Regression Neural Network

General Regression Neural Networks (GRNN) have been shown to outperform the BP3 method in some instances (Specht, 1991). These networks are known to train quickly on sparse data sets. They are useful for continuous function approximation, as in the case of the relationship between ingredient composition (e.g., CP or PA) and nutrient level (e.g., AA level). General regression neural networks can have multidimensional input, and will fit multidimensional surfaces (Ward Systems Group, 1993).

General Regression Neural Networks are three-layer networks with one hidden neuron for each training pattern. The number of neurons in the input layer was the number of inputs (1 for CP and 5 for PA). There was one output, the individual AA prediction. This type of ANN does not need a specification of training parameters such as learning rate and momentum. Calibration was used with GRNN to prevent over-training.

Post ANN Training

After the individual ANN were trained, R^2 values and Dynamic Link Libraries (DLL) were generated for each AA in each feed ingredient. The R^2 is defined statistically as the coefficient of multiple determination and is a measure of the relation between the actual and predicted values. A DLL server enables each trained ANN to be saved in a DEF extension file, which can be accessed from a computer or spreadsheet program. The spreadsheet with the DEF extension can then be used to predict AA levels in data bases not included in training. An Excel 5™ spreadsheet with the individual DEF generated for each AA in each ingredient, based on the best R^2 values determined in this paper, is available from the authors.

RESULTS AND DISCUSSION

The results for each ANN are presented in Tables 2 to 6. Table 7 summarizes the maximum R^2 values obtained

[1]St. Louis, MO 63141.

TABLE 1. Summary of data patterns (independent and dependent variables) used for Linear Regression, Three Layer Backpropagation, and General Regression Neural Networks[1]

			Input proximate analysis mean (Var)								Output	
Ingredient	AA	n	CP		Moisture		Fat		Fiber		Ash		AA Mean (Var)
			(%)									
Corn	Met	222	8.76	(0.92)	12.60	(2.44)	4.28	(2.25)	2.03	(1.56)	1.48	(0.33)	0.192 (0.0006)
	TSAA	222	8.76	(0.92)	12.60	(2.44)	4.28	(2.25)	2.03	(1.56)	1.48	(0.33)	0.393 (0.0019)
	Lys	222	8.76	(0.92)	12.60	(2.44)	4.28	(2.25)	2.03	(1.56)	1.48	(0.33)	0.273 (0.0015)
	Trp	213	8.75	(0.90)	12.57	(2.24)	4.19	(1.49)	1.97	(1.39)	1.46	(0.31)	0.059 (0.0003)
	Tyr	213	8.75	(0.90)	12.57	(2.24)	4.19	(1.49)	1.97	(1.39)	1.46	(0.31)	0.317 (0.0023)
	Thr	216	8.73	(0.91)	12.61	(2.33)	4.18	(1.47)	1.98	(1.37)	1.46	(0.31)	0.316 (0.0009)
	Arg	213	8.75	(0.90)	12.57	(2.24)	4.19	(1.49)	1.97	(1.39)	1.46	(0.31)	0.419 (0.0023)
Wheat	Met	132	13.20	(7.61)	11.09	(1.82)	2.15	(0.84)	3.84	(8.47)	2.28	(1.17)	0.225 (0.0018)
	TSAA	132	13.20	(7.61)	11.09	(1.82)	2.15	(0.84)	3.84	(8.47)	2.28	(1.17)	0.526 (0.0089)
	Lys	132	13.20	(7.61)	11.09	(1.82)	2.15	(0.84)	3.84	(8.47)	2.28	(1.17)	0.428 (0.0118)
	Trp	129	13.26	(7.45)	11.07	(1.83)	2.16	(0.88)	3.90	(8.65)	2.30	(1.21)	0.156 (0.0012)
	Tyr	129	13.26	(7.45)	11.07	(1.83)	2.16	(0.88)	3.90	(8.65)	2.30	(1.21)	0.359 (0.0055)
	Thr	129	13.26	(7.45)	11.07	(1.83)	2.16	(0.88)	3.90	(8.65)	2.30	(1.21)	0.398 (0.0056)
	Arg	129	13.26	(7.45)	11.07	(1.83)	2.16	(0.88)	3.90	(8.65)	2.30	(1.21)	0.663 (0.0324)
SBM	Met	181	47.09	(13.69)	10.70	(6.76)	3.40	(29.15)	3.74	(1.95)	7.45	(18.86)	0.688 (0.0170)
	TSAA	181	47.09	(13.69)	10.70	(6.76)	3.40	(29.15)	3.74	(1.95)	7.45	(18.86)	1.406 (0.026)
	Lys	181	47.09	(13.69)	10.70	(6.76)	3.40	(29.15)	3.74	(1.95)	7.45	(18.86)	2.930 (0.0880)
	Trp	161	47.45	(8.37)	10.73	(6.95)	3.43	(30.51)	3.62	(1.73)	7.56	(20.95)	0.653 (0.0080)
	Tyr	162	47.46	(8.36)	10.74	(6.91)	3.42	(30.34)	3.61	(1.72)	7.55	(20.83)	1.697 (0.0245)
	Thr	170	47.45	(8.38)	10.77	(6.65)	3.33	(29.05)	3.61	(1.71)	7.52	(19.91)	1.871 (0.0247)
	Arg	162	47.46	(8.36)	10.74	(6.91)	3.42	(30.34)	3.61	(1.72)	7.55	(20.83)	3.432 (0.0768)
MBM	Met	236	51.16	(44.93)	6.35	(6.71)	12.91	(71.47)	2.03	(7.04)	24.84	(86.21)	0.707 (0.045)
	TSAA	236	51.16	(44.93)	6.35	(6.71)	12.91	(71.47)	2.03	(7.04)	24.84	(86.21)	2.542 (0.364)
	Lys	236	51.16	(44.93)	6.35	(6.71)	12.91	(71.47)	2.03	(7.04)	24.84	(86.21)	1.539 (0.317)
	Trp	217	50.92	(43.85)	6.39	(7.08)	13.07	(76.38)	2.08	(7.59)	24.74	(87.26)	0.356 (0.014)
	Tyr	217	50.92	(43.85)	6.39	(7.08)	13.07	(76.38)	2.08	(7.59)	24.74	(87.26)	1.254 (0.105)
	Thr	220	50.97	(43.49)	6.36	(7.03)	13.08	(75.43)	2.07	(7.51)	24.72	(86.16)	1.792 (0.171)
	Arg	217	50.92	(43.85)	6.39	(7.08)	13.07	(76.38)	2.08	(7.59)	24.74	(87.26)	3.526 (0.188)
FSM	Met	169	60.75	(33.81)	8.93	(9.39)	9.41	(18.85)	0.81	(2.04)	18.96	(24.99)	1.593 (0.1210)
	TSAA	169	60.75	(33.81)	8.93	(9.39)	9.41	(18.85)	0.81	(2.04)	18.96	(24.99)	2.339 (0.2081)
	Lys	169	60.75	(33.81)	8.93	(9.39)	9.41	(18.85)	0.81	(2.04)	18.96	(24.99)	4.204 (0.927)
	Trp	160	60.87	(34.08)	9.03	(9.42)	9.47	(19.73)	0.82	(2.15)	18.63	(22.76)	0.607 (0.0187)
	Tyr	160	60.87	(34.08)	9.03	(9.42)	9.47	(19.73)	0.82	(2.15)	18.63	(22.76)	1.868 (0.1001)
	Thr	167	60.79	(33.42)	8.92	(9.31)	9.42	(19.07)	0.82	(2.06)	18.94	(24.78)	2.456 (0.1202)
	Arg	160	60.87	(34.08)	9.03	(9.42)	9.47	(19.73)	0.82	(2.15)	18.63	(22.76)	3.617 (0.2816)

[1]AA = amino acids; Var = variance; SBM = soybean meal; MBM = meat and bone meal; FSM = fish meal.

TABLE 2. Linear regression and neural network R^2 values for corn amino acid levels[1]

Training method[2]	Met	TSAA	Lys	Trp	Tyr	Thr	Arg
	(Proximate analysis, R^2 values[3])						
LR	0.32	0	0.46	0.34	**0.53**	0.55	0.54
BP3	0.20	0.43	0.56	0.33	0.45	0.32	0.58
GRNN	**0.78**	**0.81**	**0.96**	**0.44**	0.50	**0.85**	**0.77**
	(Protein, R^2 values)						
LR	0.25	0.47	0.18	0.27	0.50	0.50	0.43
BP3	0.26	0.44	0.17	0.33	0.47	0.38	0.41
GRNN	0.32	0.46	0.54	0.34	0.49	0.66	0.62

[1]Bold-faced values indicate the highest R^2 level (and associated regression or neural network architecture) obtained for an amino acid within each column.

[2]LR = linear regression; BP3 = three-layer backpropagation; GRNN = general regression neural network.

[3]Proximate analysis (%) = CP (%), moisture (%), fat (%), fiber (%), ash (%).

TABLE 3. Linear regression and neural network R^2 values for wheat amino acid levels[1]

Training method[2]	Met	TSAA	Lys	Trp	Tyr	Thr	Arg
	(Proximate analysis, R^2 values[3])						
LR	0.87	0.93	0.87	0.65	0.73	**0.90**	0.89
BP3	0.83	0.89	0.77	0.66	0.71	0.89	0.89
GRNN	**0.96**	**0.97**	**0.98**	**0.76**	**0.86**	**0.90**	**0.91**
	(Protein, R^2 values)						
LR	0.82	0.92	0.50	0.59	0.66	0.76	0.67
BP3	0.77	0.86	0.44	0.51	0.66	0.76	0.70
GRNN	0.79	0.85	0.46	0.57	0.77	0.77	0.72

[1]Bold-faced values indicate the highest R^2 level (and associated regression or neural network architecture) obtained for each amino acid within a column.

[2]LR = linear regression; BP3 = three-layer backpropagation; GRNN = general regression neural network.

[3]Proximate analysis (%) = CP (%), moisture (%), fat (%), fiber (%), ash (%).

TABLE 4. Linear regression and neural network R^2 values for soybean meal amino acid levels[1]

Training method[2]	Met	TSAA	Lys	Trp	Tyr	Thr	Arg
	(Proximate analysis, R^2 values[3])						
LR	0.54	0.60	**0.62**	0.53	0.50	0.62	0.46
BP3	0.68	0.45	0.61	0.52	0.45	0.63	0.46
GRNN	**0.90**	**0.78**	0.49	**0.77**	**0.58**	**0.70**	**0.53**
	(Protein, R^2 values)						
LR	0.27	0.41	0.47	0.11	0.10	0.48	0.20
BP3	0.28	0.40	0.43	0.14	0.23	0.47	0.32
GRNN	0.42	0.45	0.45	0.42	0.36	0.46	0.51

[1]Bold-faced values indicate the highest R^2 level (and associated regression or neural network architecture) obtained for each amino acid within a column.

[2]LR = linear regression; BP3 = three-layer backpropagation; GRNN = general regression neural network.

[3]Proximate analysis (%) = CP (%), moisture (%), fat (%), fiber (%), ash (%).

TABLE 5. Linear regression and neural network R^2 values for meat and bone meal amino acid levels[1]

Training method[2]	Met	TSAA	Lys	Trp	Tyr	Thr	Arg
	(Proximate analysis, R^2 values[3])						
LR	0.49	0.68	0.55	0.56	0.80	0.83	0.71
BP3	0.80	0.70	0.75	0.61	0.78	0.86	0.65
GRNN	**0.97**	**0.79**	**0.93**	**0.72**	**0.88**	**0.96**	0.64
	(Protein, R^2 values)						
LR	0.43	0.34	0.49	0.31	0.55	0.64	0.63
BP3	0.44	0.39	0.49	0.28	0.54	0.70	0.58
GRNN	0.58	0.45	0.65	0.40	0.60	0.70	**0.74**

[1]Bold-faced values indicate the highest R^2 level (and associated regression or neural network architecture) obtained for each amino acid in a column.

[2]LR = linear regression; BP3 = three-layer backpropagation; GRNN = general regression neural network.

[3]Proximate analysis (%) = CP (%), moisture (%), fat (%), fiber (%), ash (%).

TABLE 6. Linear regression and neural network R^2 values for fish meal amino acid levels[1]

Training method[2]	Met	TSAA	Lys	Trp	Tyr	Thr	Arg
	(Proximate analysis, R^2 values[3])						
LR	0.48	0.46	0.55	0.46	0.57	0.70	**0.78**
BP3	0.47	0.36	0.44	0.47	0.55	0.68	0.77
GRNN	**0.66**	**0.64**	**0.76**	**0.67**	**0.74**	**0.77**	0.73
	(Protein, R^2 values)						
LR	0.37	0.43	0.47	0.40	0.55	0.69	0.67
BP3	0.36	0.34	0.48	0.41	0.54	0.67	0.69
GRNN	0.37	0.57	0.54	0.47	0.59	0.71	0.77

[1]Bold-faced values indicate the highest R^2 level (and associated regression or neural network architecture) obtained for each amino acid in a column.

[2]LR = linear regression; BP3 = three-layer backpropagation; GRNN = general regression neural network.

[3]Proximate analysis (%) = CP (%), moisture (%), fat (%), fiber (%), ash (%).

TABLE 7. Summary of maximum R^2 values and associated regression or neural network architecture found in Tables 2 to 6 (10% test file extraction)

Ingredient[1]	Factors[2]	Met	TSAA	Lys	Trp	Tyr	Thr	Arg
Corn	R^2	0.78	0.81	0.96	0.44	0.53	0.85	0.77
	Input data	PA	PA	PA	PA	PA	PA	PA
	Method	GRNN	GRNN	GRNN	GRNN	LR	GRNN	GRNN
Wheat	R^2	0.96	0.97	0.98	0.76	0.86	0.90	0.91
	Input data	PA	PA	PA	PA	PA	PA	PA
	Method	GRNN	GRNN	GRNN	GRNN	GRNN	LR/GRNN[3]	GRNN
SBM	R^2	0.90	0.78	0.62	0.77	0.58	0.70	0.53
	Input data	PA	PA	PA	PA	PA	PA	PA
	Method	GRNN	GRNN	LR	GRNN	GRNN	GRNN	GRNN
MBM	R^2	0.97	0.79	0.93	0.72	0.88	0.96	0.74
	Input data	PA	PA	PA	PA	PA	PA	CP
	Method	GRNN	GRNN	GRNN	GRNN	GRNN	GRNN	GRNN
Fish	R^2	0.66	0.64	0.76	0.67	0.74	0.77	0.78
	Input data	PA	PA	PA	PA	PA	PA	PA
	Method	GRNN	GRNN	GRNN	GRNN	GRNN	GRNN	LR

[1]SBM = soybean meal, MBM = meat and bone meal, Fish = fish meal.

[2]R^2 = maximum R^2 from Tables 2 to 6, Input = CP (crude protein) or PA (proximate analysis), Method = Linear Regression (LR), Three Layer Back Propagation (BP3), or General Regression Neural Network (GRNN).

[3]Both LR and GRNN resulted in the same R^2.

for each AA in each ingredient (10% calibration extraction). Also presented is a description of the prediction method and input (CP vs PA) used to obtain these values.

In general, GRNN (PA input) results outperformed either LR or BP3. The BP3 method did not predict as well as expected (Caudill, 1987, 1991a), considering the wide use of BP3 in current literature. Of the 35 individual ANN optimal performance descriptions listed in Table 7, 31 were optimal using GRNN architecture. Thirty-four of the ANN predictions were best when PA input was employed. The Arg level in meat and bone meal were predicted better using CP input. The Tyr level in corn, the Lys level in soybean meal, and the Arg level in fish meal were predicted better using LR.

Neural networks appear to be a promising method for modeling the relationship between the PA of ingredients and AA content. For this study, defaults for both BP3 and GRNN architectures were held constant for all individual ANN during training. However, ANN training is still an "art" (Caudill, 1991b; Lawrence, 1991, 1993; Ward Systems Group, 1993). There is a potential to improve the predictability of the individual ANN by changing the architectures and training parameters of each ANN. When using the BP3 method, defaults such as initial linkage weights, learning rate, momentum values, pattern selection, and number of neurons in the hidden layer, can be changed for each AA ANN. For both BP3 and GRNN, the calibration test intervals and the calibration file extraction percentage can be changed. In addition, the smoothing rate of a GRNN problem can be altered (Ward Systems Group, 1993).

Also, improvements of ANN predictability may be made by changing the way the data base is presented to the ANN. For example, data base preparation can include sorting, scaling, and normalizing. It may be found that ANN need to be customized for each individual AA in each feed ingredient in order to maximize predictive abilities.

ACKNOWLEDGMENTS

Appreciation is expressed to The Pennsylvania Soybean Promotion Board (c/o American Soybean Association) for partial funding of this research project, to Margaret Sherald (Vice President, Ward Systems Group, Inc.) for her technical advice, and to Novus International for providing the ingredient data bases.

REFERENCES

Baxt, W.A.G., 1995. Application of artificial neural networks to clinical medicine. Lancet 346:1135–1138.

Caudill, M., 1987. Neural networks primer, Part I. AI Expert. December 2 (12):46–51. Series continues in later issues.

Caudill, M., 1991a. Avoiding the great backpropagation trap. AI Expert. July 6(7):29–35.

Caudill, M., 1991b. Neural network training tips and techniques. AI Expert. January 6(1):56–61.

Caudill, M., 1993. GRNN and bear it. AI Expert. May 8(5): 28–33.

Caudill, M. and C. Butler, 1993. Naturally Intelligent Systems. MIT Press, Cambridge, MA.

Das, K., and M. D. Evans, 1992. Detecting fertility of hatching eggs using machine vision II: Neural Network classifiers. Trans ASAE 35(6):2035–2041.

Deck, S., C. T. Morrow, P. H. Heinemann, and H. J. Sommer, III, 1992. Neural networks for automated inspection of produce. American Society of Agricultural Engineers Paper No. 923594. American Society of Agricultural Engineers, St. Joseph, MI.

Degussa Corporation, 1990. The Amino Acid Composition of Feedstuffs, Degussa Corp., Allendale, NJ.

Forsström, J. J., and K. J. Dalton, 1995. Artificial ANN for decision support in clinical medicine. Annal Med. 27: 509–517.

Lawrence, J., 1991. Data preparation for a neural network. AI Expert. November 6(11):34–41.

Lawrence, J., 1993. Introduction to Neural Networks. California Scientific Software Press, Nevada City, CA.

Monsanto, 1986a. Amino acids in feed ingredients and their predictability. Monsanto Nutrition Update. Vol. 4:2. Monsanto, St. Louis, MO.

Monsanto, 1986b. Amino acids in feed ingredients and their predictability. Part II. Animal byproducts. Monsanto Nutrition Update. Vol. 4:3. Monsanto, St. Louis, MO.

Monsanto, 1986c. Amino acids in feed ingredients and their predictability. Part III. Grain based ingredients. Monsanto Nutrition Update. Vol. 4:4. Monsanto, St. Louis, MO.

National Research Council, 1994. Nutrient Requirements for Poultry. 9th rev. ed. National Academy Press, Washington, DC.

Roush, W. B., Y. K. Kirby, T. L. Cravener, and R. F. Wideman, Jr., 1996. Artificial Neural Network prediction of ascites. Poultry Sci. 75:1479–1487.

SAS Institute, 1990. SAS/STAT® User's Guide: Statistics. Version 6, 4th Edition. SAS Institute Inc., Cary, NC.

Seginer, I., T. Boulard and B. J. Bailey, 1994. Neural network models of greenhouse climate. J. Agric. Eng. Res. 59: 203–216.

Specht, D. F., 1991. A general regression neural network. IEEE Trans Neural Networks. 2(6): 568–576.

Uhrig, J. W., B. A. Engel, and W. L. Baker, 1992. Pages 348–353 *in*: An Application of Neural Networks: Predicting Corn Yields. American Society of Agricultural Engineers, Orlando, FL.

Vallejo-Cordoba, B., G. E. Artega, and S. Nakai, 1995. Predicting milk shelf-life based on artificial neural networks and headspace gas chromatographic data. J. Food Sci. 60(5):885–888.

Ward, N. E., 1989. Various factors affect crude protein, amino acid in grains. Regression estimates of amino acids in ingredients. Feedstuffs 60(43).

Ward Systems Group, 1993. NeuroShell 2 User's Manual. Ward Systems Group, Inc., Frederick, MD.

Whittaker, A. D., B. S. Park, J. D. McCauley, and Y. Huang, 1991. Ultrasonic signal classification for beef quality grading through neural networks. Pages 116–125 *in*: American Society of Agricultural Engineers Publication. St. Joseph, MI.

Widman, L. E., K. A. Loparo, and N. R. Nielsen, 1989. Artificial Intelligence, Simulation and Modeling. John Wiley and Sons, New York, NY.

APPENDIX

Neural networks are radically different from conventional computing or statistical systems. The networks were inspired by the structure and operation of biological neurons. Knowledge is stored in the topology of the network itself and in the connections of the matrix, rather than in explicitly coded data structures. The problem structure is similar to that of regression analysis in that there are known inputs and outputs in a data base. The process of data manipulation is entirely different. Neural networks are composed of many simple processing units (or artificial neurons) joined through numerous interconnections. These neurons are usually organized into groups called layers. An example network would consist of a sequence of layers with connections between successive layers. The input layer is connected to the output layer through junctions with a hidden layer. The network "learns" by a process involving the modification of the connection weights between neurons and layers.

The artificial neuron receives inputs (either CP or PA in the current study) and corresponding output values (AA level). When information is loaded into a ANN, it must be scaled from the current numeric range into a range that the ANN deals with efficiently. Linear scales of [0,1] and [–1,1] are most commonly used.

After variables are imported into the ANN software program and scaled, a calibration set is extracted for use during ANN training. The percentage of the data base extracted and the method of pattern extraction may be altered. A pattern is a single row of the data base or a single observation. A rotation method of extraction selects patterns in the order they appear in the data base. A random method randomly chooses the calibration patterns.

When a network with backpropagation architecture is presented with a training set, the neuron transforms sums of inputs into weights that are transferred to other neurons. The difference between the predicted output and the actual training output is computed. The error is propagated backwards through the hidden layer to the input layer. The connection weights between neurons and layers are adjusted until the output error is minimized. All of the training set data are presented until the network is able to duplicate the training set with success. This trained ANN can then be used to predict outputs when given inputs upon which it has not been trained.

The learning rate is essential in backpropagation network training. Each time that input is presented to the network, the weights leading to the output are modified to produce a smaller error between the network prediction of output and the actual output values present in the calibration set. The amount of connection weight modification is determined by the learning rate multiplied times the error. For example, if the learning rate is 0.75, the weight change will be 3/4 of the error. Higher learning rates result in larger weight changes and faster training.

There are three available types of weight changes or updates in a backpropagation architecture. Vanilla (Euclidean) updates apply a learning rate to the weight updates but not a momentum. The term momentum is

associated with the learning rate. Momentum weight updates not only include the change dictated by learning rate, but also include a portion of the last weight change. Large learning rates can lead to oscillation of the weight changes and learning is not completed or the solution is not optimal. Making the current weight change a function of the previous weight change produces a smoothing effect. The momentum defines the proportion of the last weight change that is added to the new weight change. With a TurboProp method, weights are only updated after all patterns in the data base have been evaluated. Learning rate and momentum values are not set (Ward Systems Group, 1993).

General regression neural networks (GRNN) function by measuring how far an output prediction is from the training calibration set output in N dimensional space. N is the number of inputs in the problem. When a new pattern is presented to the network, it is compared in N dimensional space to all of the patterns in the training set to determine how far in distance it is from those patterns. The output predicted by the network is a proportional amount of all of the outputs in the training set. The proportion is based on how far the new pattern is from the given patterns in the training set.

The success of GRNN networks is dependent on a smoothing factor instead of a learning rate and momentum. The smoothing factor must be greater than 0 and can usually range from 0.01 to 1 with good results. A default smoothing factor is calculated when Calibration is used in training. An alternative smoothing factor can also be entered. Higher smoothing factors cause more relaxed surface fits through the data.

As with a regression analysis equation, either type of ANN architecture can be overfit. Just as a number of variables in equations can be made to fit a nonlinear line exactly, so too can the ANN be overtrained to fit the training data perfectly. The problem is that an overtrained ANN (or regression analysis equation) loses the ability to generalize and predict on data upon which it has not been trained. If the ANN overtrains, it starts to memorize the training set and becomes poorer at predicting the calibration set. As training proceeds, the ANN becomes better and, eventually, poorer at predicting the output values of the calibration set. The mean squared error between the actual and predicted output values are calculated by calibration and the ANN is saved at the point the calibration set is optimally predicted.

For Backpropagation networks, the network is saved every time a new minimum average error is reached. A calibration set interval is specified (usually between 50 and 500). This is the number of training patterns the ANN processes before training is temporarily interrupted and the error computed for the calibration set. Backpropagation ANN stop training when the events since the minimum error for the calibration set reaches a selected number (usually 20,000 to 40,000 events). For GRNN networks, calibration optimizes the smoothing factor. Different smoothing factors are tried and the one which results in the least mean squared error between the predicted and actual outputs is chosen.

A Dynamic Link Library (DLL) can be generated for each trained ANN. Once an ANN is trained, a DLL server enables the network to be saved in a DEF extension file, from which it can be accessed by a cell formula in an Excel™ spreadsheet. The trained network is executed in the spreadsheet by presenting new inputs to the network and observing predicted outputs.