Employing Constrained Neural Networks for Forecasting new Product's Sales Increase

Ioannis E. Livieris[1], Niki Kiriakidou[2], Andreas Kanavos[1,3], Gerasimos Vonitsanos[3], and Vassilis Tampakas[1]

[1] Computer and Informatics Engineering Department
Technological Educational Institute of Western Greece, Antirrion, Greece.
`livieris@teiwest.gr`, `vtampakas@teimes.gr`
[2] Department of Statistics and Insurance Science
University of Pireaus, Pireaus, Greece.
`niki.c.kiriakidou@gmail.com`
[3] Computer Engineering and Informatics Department
University of Patras, Patras, Greece.
`{kanavos, mvonitsanos}@ceid.upatras.gr`

Abstract. An intelligent sales forecasting system is considered a rather significant objective in the food industry, since a reasonably accurate prediction has the possibility of gaining significant profits and better stock management. Many food companies and restaurants strongly rely on their previous data history for predicting future trends in their business operations and strategies. Undoubtedly, the area of retail food analysis has been dramatically changed from a rather qualitative science based on subjective or judgemental assessments to a more quantitative science which is also based on knowledge extraction from databases. In this work, we evaluate the performance of weight-constrained neural networks for forecasting new product's sales increase. These new prediction models are characterized by the application of conditions on the weights of the network in the form of box-constraints, during the training process. The preliminary numerical experiments demonstrate the classification efficiency of weight-constrained neural networks in terms of accuracy, compared to state-of-the-art machine learning prediction models.

Keywords: Artificial neural networks, machine learning, sales forecasting, food retail industry, constrained optimization, limited memory BFGS

1 Introduction

From the beginning of the 21st century, there has been an increasing interest on the application of modern information technology and artificial intelligence to food retail analysis. Food retail market has grown continuously rapidly and steadily during the last decades. Nevertheless, the number of retail outlets forced out of business has also increased. In this competitive environment, it is necessary and vital for marketing managers in the food retail industry to utilize intelligent

and efficient tools for better decision-making. Sales forecasting and product promotion activity constitute two of the most interesting and challenging problems in the field which significantly continue to grow in scale. As a result, a variety of promotional techniques are being used by companies in their marketing communication campaigns. A conventional measure of sales promotion effectiveness is to analyze sales patterns before and after promotion [14].

Nowadays, many food companies and restaurants try to build their business strategy by exploiting huge information stored in a data warehouse. For example, the largest retailer in the United States, Walmart, has a customer database which contains more than 460 terabytes of transactional data stored on Teradata mainframes, made by NCR [13]. The vigorous development of the Internet as well as the significant storage capabilities of electronic media, enabled the accumulation and storage of large repositories of data. On the one hand, many businesses and organizations realized that the knowledge in these data constitutes the key to various strategic decisions; but much of this knowledge is hidden and untapped. On the other hand, significant pressure on marketing decision-makers has been created by the intense competition. Therefore, there has emerged an attractive need to encourage customers to engage in a long-term relationship. This new trend has lead to the realization that food businesses must tailor their services and products according to customers' preferences by leveraging a large volume of transactional data.

During the last decades, several methodologies have been applied in the field of food retail marketing for extracting useful knowledge from transactional data, while most sophisticated analysis employs machine learning algorithms in order to provide a more meticulous view. Artificial Neural Networks (ANNs) constitute one of the most dominant machine learning algorithms for extracting useful knowledge; thus they have been extensively applied and evaluated across an impressive spectrum of applications (see [8–10, 19] and the references therein). Due to their excellent capability of self-learning and self-adapting, they are very appealing to address challenging real-world problems with poorly defined system models, noisy data and strong presence of non-linear effects.

In this work, we examine and evaluate the performance of weight-constrained neural networks for forecasting new product's sales increase. To this end, we conducted a series of experiments and compared the performance of these newly proposed prediction models against state-of-the-art machine learning prediction models. The reported numerical experiments demonstrate the classification accuracy of these models, providing evidence that the application of the bounds on the weights of the network, provides more stable and reliable learning.

The remainder of this paper is organized as follows: Section 2 presents a survey of recent studies concerning the application of machine learning in food retail analysis. Section 3 analyzes the weight-constrained neural networks, while Section 4 presents the dataset utilized in our study. Section 5 presents the numerical experiments. Finally, Section 6 presents the conclusions and our proposals for future research.

2 Related Work

During the last decades, a number of decision-making models have been proposed in the literature to assist marketing managers in order to decide a business strategy based on the model's outcome. Several rewarding studies have been carried out in recent years and some useful outcomes are briefly presented below.

Hasin et al. [5] presented a fuzzy ANN approach to predict the sales of a number of selected products (including perishable foods) in a retail chain, in Bangladesh. Based on their numerical experiments, the authors concluded that the prediction performance of fuzzy ANN is better than Holt-Winter's model. Additionally, they claimed that as the forecasting period becomes smaller, the proposed model provides more accurate predictions.

Žliobaitė et al. [25] introduced an intelligent two-level sales prediction approach which utilizes a mechanism for model switching, depending on the sales behavior of a product. For recognizing the behavior, they formulated three types of categorical features: behavioral, shape-related, and relational, which allow the categorization of the products sufficiently and accurately to beat the baseline in the final prediction. In their study, they utilized real data from the food wholesales company Sligro Food Group N.V. and illustrated the superiority of their approach compared to the baseline predictor as well as an ensemble of predictors. Additionally, they presented in detail the trade-offs between the risk and benefit to the final accuracy, while moving the categorization threshold.

Lee et al. [6] evaluated the performance of three prediction models, namely Logistic Regression, Moving Average and ANNs, for forecasting fresh food sales in a point of sales database for convenience stores. The data utilized in their study were collected from 35 days of fresh food sales from the database of the Hi-Life convenience stores, including the number of sales and the amount of fresh food discarded. Their extensive experimental analysis revealed that Logistic Regression presented the highest overall classification performance.

Shukla and Jharkharia [20] investigated the applicability of ARIMA models in wholesale market models to forecast the demand for vegetables on a daily basis. In their study, they utilized sales data over a period of twenty five months from Ahmedabad wholesales market in India. Their numerical experiments showed that the proposed models are highly efficient in forecasting the demand for vegetables on a day-to-day basis. Therefore, the authors claimed that this model may be used to facilitate farmers and wholesalers in effective decision making.

In more recent works, Tsoumakas [21] provided an excellent survey presenting the most elaborate machine learning algorithms, which have been applied for food sales prediction and the appropriate measures for evaluating their accuracy. Additionally, the significant design decisions of a data analyst working on forecasting food sales were discussed in detail, such as the temporal granularity of sales data, the input variables for predicting sales and the representation of the sales output variable. Finally, the main challenges and opportunities in the domain of food sales prediction were also presented.

3   Weight-constrained Neural Networks

Recently, Livieris [7] proposed a new approach for improving the generalization of ability of ANNs by applying conditions on the weights of the network in the form of box-constraints, during the training process. The motivation behind this approach focuses on defining the weights of the trained network more consistently, in order to explore and exploit all inputs and neurons of the network. More specifically, by placing constraints on the values of weights, the likelihood that some weights will "*blow up*" to unrealistic values is considerably reduced, improving, the generalization ability of the network.

Therefore, the mathematical problem of training an ANNs is re-formulated as the following constrained optimization problem:

$$\min\{E(w) \,:\, w \in \mathcal{B}\}, \tag{1}$$

with

$$\mathcal{B} = \{w \in \mathbb{R}^n \,:\, l \leq w \leq u\}, \tag{2}$$

where $E(w)$ is the error function which depends on the connection weights w of the network, $l \in \mathbb{R}^n$ is the vector of the lower bounds on the weights and $u \in \mathbb{R}^n$ is the vector of the upper bounds on the weights.

Furthermore, in order to evaluate the efficacy and the efficiency of this approach, Livieris proposed the Weight-Constrained Neural Network (WCNN) training algorithm which is based on the L-BFGS-B method [11].

At each iteration, WCNN algorithm approximates the error function $E(w)$ by a quadratic model $m_k(w)$ at a point w_k, namely

$$m_k(w) = E_k + g_k^T(w - w_k) + \frac{1}{2}(w - w_k)^T B_k(w - w_k), \tag{3}$$

where $E_k = E(w_k)$, g_k is the gradient of the error function at w_k and B_k is a positive-definite L-BFGS Hessian approximation [11].

Next, the algorithm performs a minimization procedure of the approximation model $m_k(w)$ to compute the new vector of weights, which consists of three distinct stages. In Stage I, the model (3) is approximately minimized subject to the feasible domain $\mathcal{B}$ utilizing the gradient projection method in order to compute the generalized Cauchy point w^C. Eventually, the active set $\mathcal{A}(w^C)$ is calculated which consists of the indices of weights whose values at w^C are at lower or upper bound. In Stage II, the quadratic model $m_k(w)$ is minimized with respect to the non-active variables utilizing a direct primal method [11]. It is worth noticing that the minimization is performed at a subspace of the feasibility domain $\mathcal{B}$ by considering as free variables, the variables which are not fixed on limits while the rest variables are fixed on their boundary value obtained during the previous stage. Finally, in Stage III, the algorithm calculates the new vector of weights w_{k+1} by performing a line search procedure.

Algorithm 1: WCNN

1. Initiate w_0, c_1, c_2 and vectors l and u.
2. Set $k = 0$.
3. **repeat**
4.   Calculate the error function value E_k and its gradient ∇E_k at w_k.
5.   Set the quadratic model (3) at w_k.
    /* STAGE I: CAUCHY POINT COMPUTATION */
6.   Calculate the generalized Cauchy point w^C.
7.   Define the active set $\mathcal{A}(w^C)$.
    /* STAGE II: SUBSPACE MINIMIZATION */
8.   Minimize the quadratic model $m_k(w)$

$$\overline{w}_{k+1} = \arg \min_{w \in D_S} m_k(w)$$

    where $D_S = \left\{ w \in \mathbb{R} \,|\, l_i \leq w_i \leq u_i, \, \forall i \notin \mathcal{A}(w^C) \right\}$.
    /* STAGE III: LINE SEARCH */
9.   Set $d_k = \overline{w}_{k+1} - w_k$.
10.   Compute the learning rate η_k satisfying the strong Wolfe line search conditions

$$E_{k+1} \leq E_k + c_1 \eta_k \nabla E_k^T d_k,$$
$$|\nabla E_{k+1}^T d_k| \leq c_2 |\nabla E_k^T d_k|.$$

    with $0 \leq c_1 \leq c_2 < 1$.
11.   Update the weights $w_{k+1} = w_k + \eta_k d_k$.
12.   Set $k = k + 1$.
13. **until** (stopping criterion).

---

4   Dataset

The data collected from the IBM Watson Analytics[1] concerning a self-service fast food restaurant, located in the Connaught Place area of New Delhi with several branches all over Delhi. The chain plans to add a new product to its menu, therefore it decided three possible marketing campaigns for promoting the new product. In order to determine which promotion has the greatest effect on sales, the new item is introduced at locations in several randomly selected markets. A different promotion is used at each location and the weekly sales of the new item are recorded for the first four weeks.

---

[1] https://www.ibm.com/communities/analytics/watson-analytics-blog/marketing-campaign-eff-usec_-fastf//

Table 1 presents a set of the six (6) specific attributes utilized in our study. The first two attributes concern a unique identifier for each market and the size of market area based on sales. The following two attributes concern a unique identifier for store location and the age of store in years. The last attribute concerns one of three tested promotions. Finally, the sales for each promotion and each store were classified utilizing a three-level classification scheme:

- "*Low*": the total amount of sales is less than 170 thousands (30 instances).
- "*Average*": the total amount of sales is between 170 and 200 thousands (44 instances).
- "*High*": the total amount of sales is more than 200 thousands (63 instances).

Attribute	Type	Values
Market ID	integer	1-10
Market size	nominal	{Small, Medium, Large}
Location ID	integer	1-10
Age of store	integer	1-28
Promotion	nominal	1, 2, 3
Sales (in thousand)	nominal	{Low, Average, High}

Table 1. List of attributes

Additionally, the class distribution is presented in Figure 1.
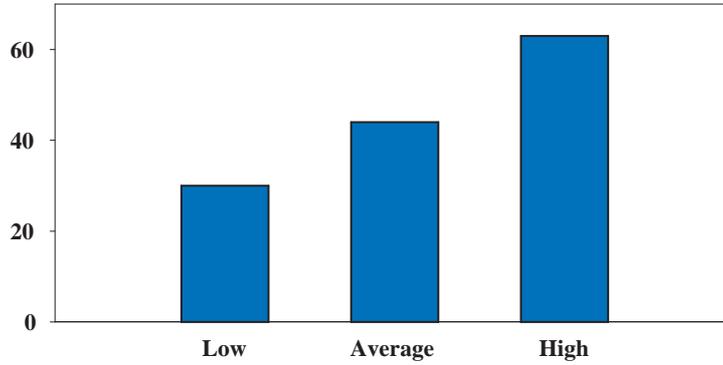

Fig. 1. Class distribution

5   Experimental Results

In this section, we present an experimental analysis in order to evaluate the performance of weight-constrained neural networks for forecasting new product's sales increase against the most popular and efficient prediction models.

The classification accuracy was evaluated using the standard procedure called *stratified* 10-*fold cross-validation* [18] i.e., this approach involves randomly dividing the set of instances into ten groups (folds), of approximately equal size, so that each fold has the same distribution of classes as the entire data set. Each fold is treated as a testing set, and the classification algorithm is fit on the remaining nine folds. The results of the cross-validation process are summarized with the mean of the prediction model skill scores.

The performance of all classification models was evaluated utilizing the following two performance metrics: F_1-*score* and *accuracy*. It is worth noticing that F_1-score consists of a harmonic mean of precision and recall while accuracy is the ratio of correct predictions of a classification model [7].

Our experimental results were obtained by conducting a two-phase procedure: In the first phase, the classification performance of the WCNN algorithm was evaluated against the most popular state-of-the-art neural network training algorithms; while in the second phase, we compare the performance of the weight-constrained neural networks trained with WCNN algorithm against the most popular and frequently utilized machine learning classification algorithms.

5.1 Performance Evaluation of Weight-constrained Neural Networks against Classical Neural Networks

In the sequel, we compare the classification accuracy of WCNN algorithm against state-of-the-art ANN training algorithms, i.e. Resilient backpropagation [17] and Levenberg-Marquardt training algorithm [4]. In other words, we evaluate the performance of weight-constrained neural networks against classical neural networks.

The implementation code was written in Matlab 7.6 and the simulations have been carried out on a PC (2.66GHz Quad-Core processor, 4Gbyte RAM) running Linux operating system, while the results have been averaged over 100 simulations. All networks had logistic activation functions and received the same sequence of input patterns and the weights were initiated using the Nguyen-Widrow method [12]. It is worth noticing that all training algorithms were utilized with their default parameter settings.

Furthermore, since a small number of simulations tend to dominate these results, the cumulative total for a performance metric over all simulations does not seem to be too informative. For this reason, we utilize the performance profiles of Dolan and Morè [2] to present perhaps the most complete information in terms of robustness, efficiency and solution quality. The use of performance profiles eliminates the influence of a small number of simulations on the benchmarking process and the sensitivity of results associated with the ranking of solvers [2]. The performance profile plots the fraction P of simulations for which any given method is within a factor τ of the best training method.

More analytically, let us assume that there exist n_s solvers and n_p problems for each solver s and problem p, requiring a baseline for comparisons, Dolan and Morè compared the performance $\alpha_{p,s}$ (based on a metric) by solver s on problem

p with the best performance by any solver on this problem; namely, using the performance ratio

$$r_{p,s} = \frac{a_{p,s}}{\min\{a_{p,s} \,:\, s \in S\}}.$$

The performance of solver s on any given problem might be of interest, but we would like to obtain an overall assessment of the performance of the solver. Next, the function ρ_s was the (cumulative) distribution function for the performance ratio is defined by

$$\rho_s(\alpha) = \frac{1}{n_p}\text{size}\left\{p \in \mathcal{P} \,:\, r_{p,s} \leq a\right\}.$$

where $\mathcal{P}$ is the set of all problems. Notice that the performance profile $\rho_s : \mathbb{R} \to [0,1]$ for a solver was a non-decreasing, piecewise constant function, continuous from the right at each breakpoint [2].

In other words, the performance profile plots the fraction P of problems for which any given algorithm is within a factor α of the training algorithm. The horizontal axis of each figure gives the percentage of the simulations for which a training algorithm achieved the best performance (efficiency). Regarding the above rules and discussion, we can conclude that one solver whose performance profile plot lies on top right, outperforms the rest of the solvers.

- "Rprop" stands for Resilient back propagation.
- "LM" stands for Levenberg-Marquardt training algorithm.
- "WCNN$_1$" stands for Algorithm 1 and bounds on the weights $-1 \leq w_i \leq 1$.
- "WCNN$_2$" stands for Algorithm 1 and bounds on the weights $-2 \leq w_i \leq 2$.
- "WCNN$_3$" stands for Algorithm 1 and bounds on the weights $-5 \leq w_i \leq 5$.

Figures 2 and 3 present the performance profiles based on F_1-score and accuracy, respectively, investigating the efficiency and robustness of each training method. Firstly, it is worth noticing that all versions of the WCNN outperformed the classical training algorithms, presenting the highest probabilities of being the optimal solvers, relative to both performance metrics. More specifically, WCNN$_1$, WCNN$_2$ and WCNN$_3$ report 28%, 52% and 42% of simulations with the best F_1-score, respectively; while Rprop and LM report 26% and 15%, respectively. As regards the classification accuracy, WCNN$_1$, WCNN$_2$ and WCNN$_3$ exhibit 55%, 63% and 59% of simulations with the highest accuracy, respectively; while Rprop and LM report 35% and 20%, respectively. Thus, we conclude that the application of the bounds on the weights of the neural network, increased the overall classification accuracy.

Furthermore, regarding the performance of the proposed algorithm, WCNN$_2$ illustrates the best performance, followed by WCNN$_1$ and WCNN$_3$. Therefore, from the interpretation of Figures 2 and 3, we are able to conclude that the application of the bounds on the weights of a neural network significantly increased the overall classification accuracy. Nevertheless, in case the bounds were too tight, this did not substantially benefit much the classification performance of the networks.
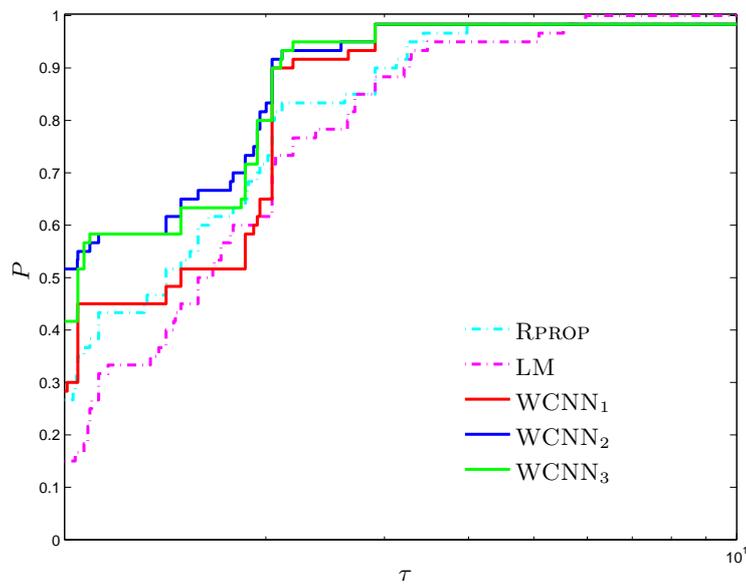
Fig. 2. Log_{10} scaled performance profiles for RPROP, LM, $WCNN_1$, $WCNN_2$ and $WCNN_3$ based on F_1-score
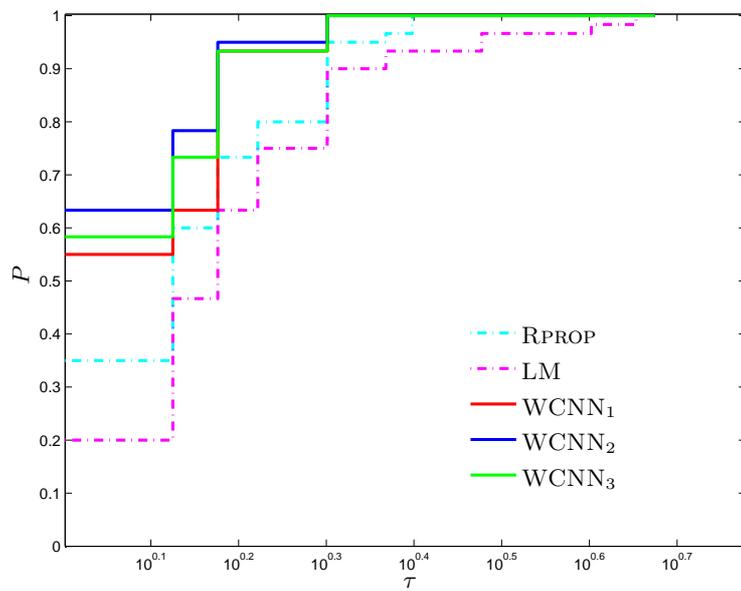

Fig. 3. Log_{10} scaled performance profiles for RPROP, LM, $WCNN_1$, $WCNN_2$ and $WCNN_3$ based on accuracy

5.2   Performance Evaluation of Weight-constrained Neural Networks against state-of-the-art Machine Learning Algorithms

Next, we evaluate the performance of the weight-constrained neural networks trained with WCNN against state-of-the-art machine learning algorithms. The Naive Bayes algorithm [3] was the representative of the probabilistic classifiers while kNN algorithm [1] was selected as instance-based learner with Euclidean distance as distance metric. From the support vector machines, we have selected the Sequential Minimal Optimization (SMO) algorithm [15] using Pearson VII function-based universal kernel [22]. From the decision trees, C4.5 algorithm [16] was chosen in our study. These classification algorithms constitute some of the most effective and widely utilized machine learning algorithms for classification [24] and are implemented in Java using WEKA Machine Learning Toolkit [23].

Algorithm	F_1-score	Accuracy
WCNN$_1$	77.18%	79.56%
WCNN$_2$	**79.04%**	**80.22%**
WCNN$_3$	79.17%	79.78%
Naive Bayes	63.82%	62.77%
kNN	71.33%	71.53%
SMO	74.67%	74.45%
C4.5	77.55%	77.37%

Table 2. Performance comparison of the weight-constrained neural network trained with WCNN$_1$, WCNN$_2$ and WCNN$_3$ against state-of-the-art classification algorithms, regarding F_1-score and accuracy

Table 2 presents the performance comparison of the weight-constrained neural networks, which were trained with all versions of WCNN against state-of-the-art classification algorithms, relative to F_1-score and accuracy. Notice that the best performance is highlighted in bold for each performance metric. Firstly, it is worth mentioning that the neural networks trained with WCNN$_2$ exhibited the highest average F_1-score. As regards the classification accuracy, the ANNs trained with all version of WCNN presented the best performance, outperforming all state-of-the-art classification algorithms.

Based on the above discussion, we are able to conclude out that the weight-constrained neural networks perform significantly better than any presented classification algorithm, reporting the best overall performance.

6   Conclusions

In this work, we evaluated the classification accuracy of weight-constrained neural networks for forecasting new product's sales increase. By placing constraints

on the values of weights, the likelihood that some weights will *"blow up"* to unrealistic values is considerably reduced; thus all inputs and neurons of the network are efficiently explored. These modified prediction models are efficiently trained with a new training algorithm, called WCNN, which exploits a gradient-projection strategy for handling the bounds on the weights together with the numerical and theoretical advantages of the limited memory BFGS matrices. Our preliminary numerical experiments illustrate the classification efficiency of weight-constrained neural networks in terms of accuracy, compared to state-of-the-art machine learning prediction models.

Nevertheless, the a-priori determination of the optimal bounds on the weights of the networks is a rather challenging open problem; thus more research and experiments are needed. To this end, the question of what should be the values of the bounds for each benchmark or which additional constraints should be applied is still under consideration. Probably, the required research to answer these questions may reveal additional and crucial information and questions.

Our future work is concentrated on incorporating the proposed methodology to more advanced and complex neural network architectures. Additionally, since our experimental results are quite encouraging, a next step could be the evaluation of the proposed framework in challenging real-world regression problems.

References

1. Aha, D.: Lazy learning. Springer Science & Business Media (2013)
2. Dolan, E., Moré, J.: Benchmarking optimization software with performance profiles. Mathematical Programming 91, 201–213 (2002)
3. Domingos, P., Pazzani, M.: On the optimality of the simple Bayesian classifier under zero-one loss. Machine Learning 29, 103–130 (1997)
4. Hagan, M., Menhaj, M.: Training feed-forward networks with the marquardt algorithm. IEEE transactions on Neural Networks 5(6), 989–993 (1994)
5. Hasin, M., Ghosh, S., Shareef, M.: An ANN approach to demand forecasting in retail trade in bangladesh. International Journal of Trade, Economics and Finance 2(2), 154 (2011)
6. Lee, W., Chen, C., Chen, K., Chen, T., Liu, C.: Comparative study on the forecast of fresh food sales using logistic regression, moving average and bpnn methods. Journal of Marine Science and Technology 20(2), 142–152 (2012)
7. Livieris, I.: Improving the classification efficiency of an ANN utilizing a new training methodology. Informatics 6 (2018)
8. Livieris, I., Drakopoulou, K., Pintelas, P.: Predicting students' performance using artificial neural networks. In: Information and Communication Technologies in Education. pp. 321–328 (2012)
9. Livieris, I., Pintelas, P.: An improved spectral conjugate gradient neural network training algorithm. International Journal on Artificial Intelligence Tools 21(01), 1250009 (2012)
10. Maren, A., Harston, C., Pap, R.: Handbook of neural computing applications. Academic Press (2014)
11. Morales, J., Nocedal, J.: Remark on "Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound constrained optimization". ACM Transactions on Mathematical Software (TOMS) 38(1), 7 (2011)

12. Nguyen, D., Widrow, B.: Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights. In: International Joint Conference on Neural Networks. pp. 21–26. IEEE (1990)
13. Pathak, S.: Social implications of data mining techniques. Journal of Basic and Applied Engineering Research (JBAER) p. 860 (2014)
14. Peattie, S.: Promotional competitions as a marketing tool in food retailing. British Food Journal 100(6), 286–294 (1998)
15. Platt, J.: Using sparseness and analytic QP to speed training of support vector machines. In: Kearns, M., Solla, S., Cohn, D. (eds.) Advances in neural information processing systems. pp. 557–563. MA: MIT Press (1999)
16. Quinlan, J.: C4.5: Programs for machine learning. Morgan Kaufmann, San Francisco (1993)
17. Riedmiller, M., Braun, H.: A direct adaptive method for faster backpropagation learning: The RPROP algorithm. In: IEEE International Conference on Neural Networks. pp. 586–591. IEEE (1993)
18. Russell, S., Norvig, P.: Artificial intelligence: a modern approach. Malaysia; Pearson Education Limited, (2016)
19. Shanmuganathan, S., Samarasinghe, S.: Artificial neural network modelling, vol. 628. Springer (2016)
20. Shukla, M., Jharkharia, S.: Applicability of ARIMA models in wholesale vegetable market: an investigation. International Journal of Information Systems and Supply Chain Management (IJISSCM) 6(3), 105–119 (2013)
21. Tsoumakas, G.: A survey of machine learning techniques for food sales prediction. Artificial Intelligence Review pp. 1–7 (2018)
22. Üstün, B., Melssen, W., Buydens, L.: Facilitating the application of support vector regression by using a universal pearson vii function based kernel. Chemometrics and Intelligent Laboratory Systems 81(1), 29–40 (2006)
23. Witten, I., Frank, E., Hall, M., Pal, C.: Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann (2016)
24. Wu, X., Kumar, V., Quinlan, J., Ghosh, J., Yang, Q., Motoda, H., McLachlan, G., Ng, A., Liu, B., Yu, P., Zhou, Z., Steinbach, M., Hand, D., Steinberg, D.: Top 10 algorithms in data mining. Knowledge and information systems 14(1), 1–37 (2008)
25. Žliobaitė, I., Bakker, J., Pechenizkiy, M.: Beating the baseline prediction in food sales: How intelligent an intelligent predictor is? Expert Systems with Applications 39(1), 806–815 (2012)