Demand Forecasting using Artificial Neuronal Networks and Time Series: Application to a French Furniture Manufacturer Case Study

Julie Bibaud-Alves[1,2,3], Philippe Thomas[1,2,a] and Hind Bril El Haouzi[1,2]
[1]*Université de Lorraine, CRAN, UMR 7039, Campus Sciences, BP 70239, 54506 Vandœuvre-lès-Nancy Cedex, France*
[2]*CNRS, CRAN, UMR7039, France*
[3]*Parisot Meubles, 15 Avenue Jacques Parisot, 70800 Saint-Loup-sur-Semouse, France*

Keywords: Sales Forecasting, Neural Networks, Multilayer Perceptron, Seasonality, Overfitting, Time Series.

Abstract: The demand forecasting remains a big issue for the supply chain management. At the dawn of Industry 4.0, and with the first encouraging results concerning the application of deep learning methods in the management of the supply chain, we have chosen to study the use of neural networks in the elaboration of sales forecasts for a French furniture manufacturing company. Two main problems have been studied for this article: the seasonality of the data and the small amount of valuable data. After the determination of the best structure for the neuronal network, we compare our results with the results form AZAP, the forecasting software using in the company. Using cross-validation, early stopping, robust learning algorithm, optimal structure determination and taking the mean of the month turns out to be in this case study a good way to get enough close to the current forecasting system.

1 INTRODUCTION

One of the most important decision problem of supply chain management is the demand forecasting in order to balance inventory and service levels (Brown 1959), (Chapman et al., 2017). (Boone et al., 2019) note that the role and use of artificial intelligence and machine learning methods in supply chain forecasting remains underexplored. Besides, the important issues to the development of supply chain forecasting are:

- The processes and systems through which the disaggregated forecasts are produced;
- Methods and selection algorithms that are suitable for supply chain data;
- The impact of new data sources from both consumer and supply chain partners;
- The effects of uncertainty and forecasts errors on the supply chain;
- The effects of linking forecasting to supply chain decisions, at both aggregate and disaggregate levels.

Several quantitative approaches exist. For this paper, we chose to focus our work on neural approaches, which are more and more used in supply chain management (Carbonneau et al., 2008). The purpose of this paper is to evaluate the ability of a neural model of time series prediction to predict sales. As this work is being conducted as part of a collaboration project between CRAN laboratory and Parisot company, we apply this approach to Parisot, which is currently using a commercial software to establish its sales forecasts. Two main problems have been studied for this article:

- The seasonality, which will be taken into account through two different ways as input data in the experiments: by assigning a number to the month in question, and by using the average, over the period, of the recorded orders relating to the month in question
- The small amount of data available, constituting a risk of overfitting, which will be taken into account through the algorithm.

The paper begins by presenting a brief state of the art regarding sales forecasts, then the neural network methods used in this work is presented. Then we will present the results before concluding and putting into perspective this work.

[a] https://orcid.org/0000-0001-9426-3570

Bibaud-Alves, J., Thomas, P. and El Haouzi, H.
Demand Forecasting using Artificial Neuronal Networks and Time Series: Application to a French Furniture Manufacturer Case Study.
DOI: 10.5220/0008356705020507
In *Proceedings of the 11th International Joint Conference on Computational Intelligence (IJCCI 2019)*, pages 502-507
ISBN: 978-989-758-384-1

2 SALES FORECASTING

In classical forecasting techniques, it is first of all a question of straightening the series in order to eliminate the “accidental” variations whose origin is known. Next, we must determine the typology of the time series. Only after, comes the stage of the selection of forecast techniques adapted to the problems.

Complex time series can be broken down into three components (Chatfield 1996), (Nelson 1973):

- A trend component
- A cyclical component (seasonal) which can itself know a temporal evolution
- A random component (disturbance)

Concerning forecasting techniques, (Giard 2003) proposes the following classification, Figure 1:

- Explanatory models (called exogenous): the forecast is based on values taken by variables other than those we are trying to predict.
- Autoprojective models (called endogenous): the future is simply deduced from the past. We can refer to (Kendall 1976), (Kendall and Stuart 1976).

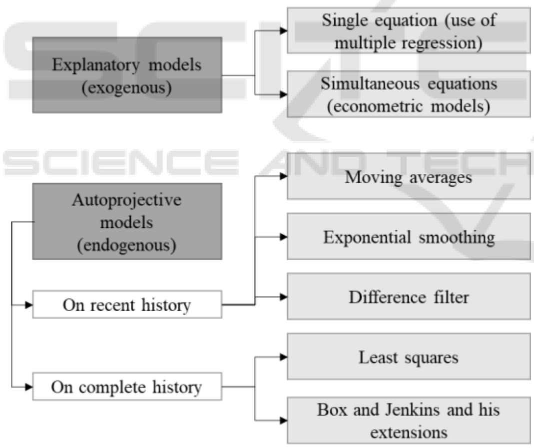

Figure 1: Forecasting techniques classification.

The choice of the technique must minimize the cost of forecasting for a given level of precision, taking into account the type of time series, and the purpose, depending on whether the forecast will be used in the short or long term.

As part of Industry 4.0, the 4th Industrial Revolution centred on the data valorisation, researchers are turning to approaches based on artificial intelligence and new technologies increasingly accessible to businesses (Liu et al., 2013). These new approaches seem to offer overall better results than traditional methods (Jurczyk et al., 2016).

3 MULTILAYER PERCEPTRON

3.1 Structure

The traditional multilayer neural network consists of only one hidden layer (using a sigmoidal activation function) and one output layer. It is commonly referred to as a multilayer perceptron (MLP) and has been proven to be a universal approximator (Cybenko 1989), (Funahashi 1989). Its structure is given by:

$$\hat{y} = g_o\left(\sum_{h=1}^{n_o} w_h^2 . g_h\left(\sum_{i=1}^{n_i} w_{hi}^1 . x_i + b_h^1\right) + b\right) \qquad (1)$$

where x_i are the n_i inputs, w_{hi}^1 are connecting weights between input and hidden layers, b_h^1 are the hidden neurons biases, $g_h(.)$ is the activation function of the hidden neurons (hyperbolic tangent), w_h^2 are connecting weights between hidden and output layers, b is the bias of the output neuron, $g_o(.)$ is the activation function of the output neuron, and $\hat{y}$ is the network output. The considered problem is a regression one. The $g_o(.)$ is a linear activation function.

Learning of a MLP is performed by using a local minimum search. Therefore, the model accuracy depends of the choice of the initial parameter set. The one used here is a modification of the Nguyen and Widrow (NW) algorithm (Nguyen and Widrow 1990). It allows a random initialization of weights and biases, combined with an optimal placement in the input space (Demuth and Beale 1994)

This type of MLP has been extensively used in a broad range of problems including classification, function approximation, regression, or times-series forecasting.

The considered problem here is a sales forecasting problem which can be viewed as a special case of times-series forecasting including consideration of seasonality.

Moreover, this problem must be solved by using small datasets which can lead to overfitting problem.

3.2 The Seasonality Problem

Seasonality is an aspect of the data that makes the prediction task difficult. (Zhang and Qi 2005) studie the effectiveness of seasonal series pre-treatments on the performance of multilayer perceptron. They conclude that even with pre-processing, their performance remains limited.

To take into account the phenomenon of seasonality, we propose to test and compare two approaches using the dataset of the company (here AZAP FORECAST and HISTORICAL ORDERS, used in the columns CDES 1 to 3 in the following paper):

- We replace the month with a number, Table 1, which is varied (as example, in a first approach, January is assigned the value 1, February 2…, in a second step, January is assigned the value 12, February 1…), Table 2 ;
- We add the average value of sales for the chosen month considering the whole dataset (for January is 34586, for February is 33154, etc.), Table 3.

Table 1: Construction of the input file considering the number of the month.

AZAP FORECASTS	HISTORICAL ORDERS	CDES-1	CDES-2	CDES-3	MONTH	MONTH NUMBER
55 805	46 592	0	0	0	January	1
45 573	29 820	46 592	0	0	February	2
41 987	37 487	29 820	46 592	0	March	3
46 950	35 861	37 487	29 820	46 592	April	4
39 270	18 772	35 861	37 487	29 820	Ami	5
51 663	43 712	18 772	35 861	37 487	June	6
47 088	33 106	43 712	18 772	35 861	July	7
33 124	24 074	33 106	43 712	18 772	August	8
27 542	25 832	24 074	33 106	43 712	September	9
30 520	31 578	25 832	24 074	33 106	October	10
23 529	22 093	31 578	25 832	24 074	November	11
47 602	41 048	22 093	31 578	25 832	December	12
45 186	31 821	41 048	22 093	31 578	January	1
45 572	38 660	31 821	41 048	22 093	February	2

Table 2: Principle of shift of the number of the month.

HISTORICAL ORDERS	CDES-1	CDES-2	CDES-3	MONTH NUMBER January1	MONTH NUMBER January2	MONTH NUMBER January3
46 592	0	0	0	1	2	3
29 820	46 592	0	0	2	3	4
37 487	29 820	46 592	0	3	4	5
35 861	37 487	29 820	46 592	4	5	6
18 772	35 861	37 487	29 820	5	6	7
43 712	18 772	35 861	37 487	6	7	8
33 106	43 712	18 772	35 861	7	8	9
24 074	33 106	43 712	18 772	8	9	10
25 832	24 074	33 106	43 712	9	10	11
31 578	25 832	24 074	33 106	10	11	12
22 093	31 578	25 832	24 074	11	12	1
41 048	22 093	31 578	25 832	12	1	2
31 821	41 048	22 093	31 578	1	2	3
38 660	31 821	41 048	22 093	2	3	4

3.3 The Overfitting Problem

One of the main risks encountered in the use of machine learning is the overfitting problem. This problem is related to the fact that the dataset used to learn is generally noisy and generally corrupted by outliers. If the model used includes too many parameters (degrees of freedom), the learning step can lead to learning noise at the expense of learning the underlying system. To avoid this, different approaches may be used individually or in combination.

A classical approach is to perform a cross validation. Different approaches may be used. The considered here is the holdout method consisting by dividing the dataset into learning and validation dataset. This approach allows to detect and avoid this phenomenon by conjunction with the second approach: the early stopping. Early stopping consists to stop the learning when the performance of the model begins to deteriorate on the validation dataset.

The early stopping may be automatized by monitoring some parameters of the learning algorithm used. As example, in the Levenberg-Marquardt algorithm, it is possible to monitor the evolution of the gradient value and/or of the evolution of the parameter λ chosen to ensure the inversion of the Hessian matrix (Levenberg 1944), (Marquardt 1963). The learning algorithm used here is derived from the one proposed by (Norgaard 1995) which includes such mechanisms.

Table 3: Construction of the input file considering the average of the month.

AZAP FORECASTS	HISTORICAL ORDERS	CDES-1	CDES-2	CDES-3	MONTH	MONTH MEAN
55 805	46 592	0	0	0	January	34586
45 573	29 820	46 592	0	0	February	33154
41 987	37 487	29 820	46 592	0	March	31819
46950	35861	37 487	29 820	46 592	April	31688
39270	18772	35 861	37 487	29 820	Ami	24800
51663	43712	18 772	35 861	37 487	June	33862
47088	33106	43 712	18 772	35 861	July	31975
33124	24074	33 106	43 712	18 772	August	28447
27542	25832	24 074	33 106	43 712	September	30370
30520	31578	25 832	24 074	33 106	October	39067
23529	22093	31 578	25 832	24 074	November	29261
47602	41048	22 093	31 578	25 832	December	37296
45186	31821	41 048	22 093	31 578	January	34586
45572	38660	31 821	41 048	22 093	February	33154

The overfitting is directly related to the inclusion of useless parameters in the model. Another approach to avoid this problem is to determine the optimal structure of the model. This may be done by using constructive approach (Kwok and Yeung 1997) or pruning procedure (Thomas and Suhner 2015). Another approach is to use trial and error approach to determine the optimal structure of this model. This is this last approach which is used here because it is the simplest.

The last approach to limit the overfitting problem is to include a regularization effect in the learning algorithm. This effect may be obtained by adding a weight decay term to the criterion to minimize in the learning algorithm (Norgaard 1995). Another approach is to use a robust criterion to minimize in the learning algorithm (Thomas et al., 1999). This is this second approach which is used here.

So, to conclude, to avoid the overfitting risk, the optimal structure is obtained by using trial and error approach. A cross-validation is performed, and an automatic early stopping is performed. Last, a robust criterion is included in the learning algorithm.

4 APPLICATION AND EXPERIMENTAL RESULTS

4.1 Industrial Context

This paper is a case study, based on the data from Parisot – Saint Loup's Unit. It's an octogenarian French furniture industry. It sells furniture kits made of particle board in France and around the world. The catalog consists of furniture for bedroom, living room, kitchen, bathroom, in single package or multi package.

This company has had to face in recent years many changes. Sudden adaptations have not necessarily been optimal and have led to bad long-term practices. These bad practices now prevent the company from remaining competitive.

Today, the forecasts are made using AZAP [1].

AZAP is a publisher of a software suite that assists with optimum Supply Chain management. It covers the management functions of sales forecasting and demand management, industrial planning and inventory level optimization of companies looking to significantly cut costs, optimise stock management and improve customer service promptly.

The need of the company is to improve the sales forecasting function without the need of encapsulated business logic. Thus, the ML (Machine Learning) could be a good candidate.

4.2 Dataset

We used the summary file of final forecasts and orders recorded by large groups of customers. The data are monthly recorded from 2012 to 2018, constituting 81 values (named HISTORICAL ORDERS in the next tables), which is a small dataset. We chose to separate the series into two datasets, one for learning and one for validation. The chosen method is that of 80/20 randomly. So, the learning dataset includes 62 patterns when the validation one includes 16 patterns. The dataset is normalized before to perform the learning:

$$x_{norm} = \frac{x - \min_x}{\max_x - \min_x} \tag{2}$$

where $\min_x$ is the minimum and $\max_x$ is the maximum of the variable x.

In order to compare the results between them, we use the root mean square error value (RMSE):

$$RMSE = \sqrt{\frac{1}{N}\sum_{n=1}^{N}\left(y(n) - \hat{y}(n)\right)^2} \tag{3}$$

Where N is the size of the dataset, $y(n)$ is the n^{th} target, and $\hat{y}(n)$ its estimation.

4.3 Structure Determination using a Trial / Error Approach

The first step in this study was to determine the structure of the neural network.

In times-series forecasting, the prediction of the output (here HISTORICAL ORDERS) is performed by using past data of orders. The first step is to determine the time window. To do that, we varied the number of inputs considering the data about the orders, as shown by Table 4. This amounts to varying the number of delayed data to be taken into account in relation to the value considered (here CDES 1, CDES 2 and CDES 3).

Table 4: Construction of delayed data to be taken into account as inputs.

AZAP FORECASTS	HISTORICAL ORDERS	CDES-1	CDES-2	CDES-3
55 805	46 592	0	0	0
45 573	29 820	46 592	0	0
41 987	37 487	29 820	46 592	0
46 950	35 861	37 487	29 820	46 592
39 270	18 772	35 861	37 487	29 820
51 663	43 712	18 772	35 861	37 487

[1] http://www.azap.net/en/

We varied the frame size from 3 to 6, and, to avoid the problem of local minimum search, this task is performed hundred times and the best result is kept. The number of hidden neurons is tuned to 4. Table 5 presents the best RMSE obtained on the validation data set in function of the number of delayed inputs. This RMSE is calculated with normalized dataset. These results show that using a time window of size 3 is enough to learn the problem. Including more delayed inputs tends to degrade the accuracy of the model on the validation dataset (overfitting). So, in the sequel, the time window is tuned to 3.

Table 5: Results of the determination of the time window.

The number of delayed data	3	4	5	6
RMSE	0.0365	0.0511	0.0412	0.0585

In a second step, the optimal number of hidden neurons is expected, we varied its number from 3 to 6. As for the determination of the number of delayed inputs to use, to avoid the local search minimum problem, ten learning has been performed on ten different initial sets of parameters and the best result is kept. Table 6 presents the RMSE obtained on the validation dataset (normalized) when the number of hidden nodes varies. These results show that the variation of this number has no significant impact on the accuracy of the model. So, the number of hidden neurons is set to 3.

Table 6: Results of the determination of the number of the hidden neurons.

The number of hidden neurons	3	4	5	6
RMSE	0.0365	0.0366	0.0377	0.0369

For all following experiments, the structure of the MLP includes:

- The small amount of data available, which constitutes a risk of over-learning
- 4 inputs: the time window of 3 historical values and an input related to the month: the number or the mean of orders to the month considered.
- 3 hidden neurons.

4.5 Comparison

In a first step, the seasonality is considered by associating each month to a real value (Table 2). This approach presents the drawback to introduce an important gap between two consecutive months. As example, by associating 1 to January, 2 to February…, the transition from December to January induces a jump from 12 to 1. To evaluate the impact of this jump on the accuracy of the model, a permutation of these association is performed (in a first experiment, January is associated to 1, in a second it is associated to 2…). As for the determination of the structure, 10 learning has been performed with 10 initial set of parameters for each experiment in order to avoid the local minimum search problem.
Table 7 presents the RMSE obtained on the validation dataset for these 12 experiments. In order to compare these results with those obtained with AZAP, this RMSE is calculated on non-normalized dataset.

These results show that the position of the jump in the year has an important impact on the results. As example, when it is situated between January and February (line NN FORECASTS January12) the accuracy is 27% worse than that it is situated between May and June (line NN FORECASTS January8).

We can see that NN forecasts with some sequences (January3, January 4 and January 8) are slightly better than the AZAP forecasts.

Table 7: Results considering the variation of the month number.

CONSIDERED EXPERIMENT	RMSE
AZAP FORECASTS	**5882**
NN FORECASTS January1	7071
NN FORECASTS January2	6189
NN FORECASTS January3	**5814**
NN FORECASTS January 4	**5882**
NN FORECASTS January5	6332
NN FORECASTS January6	6870
NN FORECASTS January7	6678
NN FORECASTS January8	**5753**
NN FORECASTS January9	6140
NN FORECASTS January10	6050
NN FORECASTS January11	6213
NN FORECASTS January12	7287

The second approach to consider the seasonality, is to associate to each month the average sales for the considered month (Table 3). Table 8 presents the RMSE obtained on the validation dataset with this strategy and compares the obtained results with those obtained with AZAP and with the best model using the preceding strategy. These results show that the using of the average sales for the considered month improves the accuracy of the model comparatively to the two others. The improvement is of 5.5% comparatively to the preceding approach and up to 8% comparatively to AZAP. This can be explained by the fact that this approach gives a richer information.

Table 8: Results considering the input of the month mean.

CONSIDERED EXPERIMENT	RMSE
AZAP FORECASTS	5882
NN FORECASTS January8	5753
NN FORECASTS mean of the month	**5450**

5 CONCLUSIONS

This paper presents an initial work of a Ph.D study in collaboration with a French furniture manufacturer. Its goal is to propose a machine learning approach to perform sales forecasting.

A classical neural network model (multilayer perceptron) is used. The main difficulty is related to the small size of the dataset which can lead to the overfitting problem. To avoid it, a combination of different strategies is used (cross-validation, early stopping, robust learning algorithm, optimal structure determination).

The second difficulty is related to the taking into account of the seasonality. Two approaches have been proposed and compared. This study has shown that taking the mean of the month into account as an input is significant to solve the problem of seasonality.

We have to take in account that our result consists of the basic forecasts in the AZAP process, and we compare our results with the final forecasts obtained after the forecaster job. In future works, we must add the information of the commercial and marketing forecasts taking into account the effect of publicity and events. We can also test the impact of agglomerating or disaggregating customer hierarchy data. Last, the cross-validation approach used here is the holdout method which is simple but maybe not the more efficient when the dataset is small. Other cross-validation approaches must be tested and compared in the future such as k-fold or leave one out as example.

REFERENCES

Boone, T., Boylan, J. E., Ganeshan, R., Sanders, N., 2019. Perspectives on Supply Chain Forecasting. *International Journal of Forecasting* 121–127.

Brown, R. G., 1959. *Statistical Forecasting for Inventory Control*. McGraw-Hill.

Carbonneau, R., Laframboise, K., Vahidov, R., 2008. Application of Machine Learning Techniques for Supply Chain Demand Forecasting. *European Journal of Operational Research* (184): 1140–1154.

Chatfield, C., 1996. *The Analysis of Time-Series: Theory and Practice*. 5th edition. Chapman and Hall.

Cybenko, G., 1989. Approximation by Superpositions of a Sigmoidal Function. Mathematics of Control, *Signals and Systems* 2(4): 303–314.

Demuth, H., Beale, P., 1994. *Neural Networks Toolbox User's Guide*. 2th edition. The MathWorks.

Funahashi, K.I., 1989. On the Approximate Realization of Continuous Mappings by Neural Networks. *Neural Networks*, 2(3): 183–192.

Giard, V., 2003. *Gestion de Production et Des Flux*. 3th edition. Economica.

Jurczyk, K., Gdowska, K., Mikulik, J., Wozniak, W., 2016. Demand Forecasting with the Usage of Artificial Neural Networks on the Example of Distribution Enterprise. *Int. Conf. on Industrial Logistics ICIL'2016*, September 28- october1, Zakopane, Poland.

Kendall, M. G., 1976. *Time Series*. 2th edition. Griffin.

Kendall, M.G., Stuart, A., 1976. *The Advanced Theory of Statistics, vol.III*. 3th edition. Griffin.

Kwok, T.Y., Yeung, D.Y., 1997. Constructive Algorithms for Structure Learning in Feedforward Neural Networks for Regression Problems. *IEEE Transactions On Neural Networks*, 8: 630–645.

Levenberg, K., 1944. A Method for the Solution of Certain Non-Linear Problems in Least Squares. *Quarterly of Applied Mathematics*, 2(2): 164–168.

Liu, N., Choi, T.M., Hui, C.L., Ng, S.F. 2013. Sales forecasting for fashion retailing service industry: a review. *Mathematical Problems in Engineering*, http://dx.doi.org/10.1155/2013/738675.

Marquardt, D.W., 1963. An Algorithm for Least-Squares Estimation of Nonlinear Parameters. *Journal of the Society for Industrial and Applied Mathematics* 11(2). JSTOR: 431–441.

Nelson, C. R., 1973. *Applied Time Series Analysis for Managerial Forecasting*. Holden-Day.

Nguyen, D., and Widrow B., 1990. Improving the Learning Speed of 2-Layer Neural Networks by Choosing Initial Values of the Adaptive Weights. *In IJCNN*.

Norgaard, M., 1995. *Neural Network Based System Identification Toolbox*. Tech Report 95-E-773. Institute of Automation, Technical University of Denmark.

Stephen N. Chapman, J.R., Arnold, T., Gatewood, A.K., Clive, L.M., 2017. *Introduction to Materials Management*. 8th edition. Pearson.

Thomas, P., Bloch, G., Sirou, F., Eustache, V., 1999. Neural Modeling of an Induction Furnace Using Robust Learning Criteria. *Integr. Comput.-Aided Eng*. 6(1): 15–26.

Thomas, P., Suhner, M.C., 2015. A New Multilayer Perceptron Pruning Algorithm for Classification and Regression Applications. *Neural Processing Letters* 42(2): 437–458.

Zhang, P., Qi, M., 2005. Neural Network Forecasting for Seasonal and Trend Time Series. *European Journal of Operational Research* 160: 501–514.