Modeling and optimizing the complex distribution supply chain to improve customer satisfaction using artificial intelligence

RAOILINIAINA William – Dr ROBINSON Matio – Pr ANDRIAMANOHISOA Hery Zo

Ecole Supérieure Polytechnique Antananarivo (ESPA) - Université d'Antananarivo
BP 1500, Ankatso – Antananarivo 101 – Madagascar

[1] *expwillo@gmail.com,* [2] *mat_robinson2000@yahoo*.fr, *aherizo@mail.fr*

Keywords: Data science, statistics, Machine Learning, Multilayer Neural Network, ARMA, Time Series Forecasting, Deep Learning

ABSTRACT

Managing distribution supply chain activities is of crucial importance to ensure the efficient and satisfactory delivery of products to end customers. Aligning system structure, replenishment levels, and accurate demand forecasting optimizes distribution, enhances customer satisfaction, and reduces associated costs.

1. INTRODUCTION

To cope with a competitive and uncertain economic environment, such as the one in which companies find themselves today, it will be challenging to meet the increasingly diverse and demanding customer expectations.

Various factors, such as road conditions, geographical distance, product availability in stock, available workforce, as well as natural disasters, can influence supply chain operations. These elements can disrupt logistics operations or cause damage to infrastructure, warehouses, and inventory. Furthermore, customer orders exhibit high variability, frequent disruptions occur in the supply chain, and delivery times are not met.

Given that business responsiveness is no longer sufficient, companies must proactively seek competitive advantages.

The distribution of finished products at the right time, in the right place, in the right quantity, while meeting the requirements of end customers, and at the lowest cost," as emphasized by Dominguez and Lashkari (2004). One of the benefits of the supply chain is sales optimization. This optimization involves strategically positioning products in optimal quantities, at the right locations, and at the ideal timing, all while minimizing costs.
The effective management of the supply chain is a strategic challenge for companies seeking to maintain their competitiveness in a complex and ever-evolving business environment. Customer satisfaction has thus become the ultimate measure of success.

How can companies model and optimize their complex supply chain using artificial intelligence, reduce overall logistic costs, and optimize inventory levels, all while ensuring optimal service to their customers?

What approaches and methodologies are most suitable for addressing these complex challenges?
The aim of this study is to achieve the following objectives:

- Determine, based on a supply chain model, for each period and depending on forecasted customer orders, the minimum quantity to be supplied from central warehouses to fulfill all customer orders while optimizing multi-level supply chain costs.
- Deduce the products to replenish in each intermediate warehouse and distribution depot.

The remainder of this article is organized as follows: Section 2 presents various previous works related to this issue. Section 3 elaborates on the application models we used to find solutions. Finally, we will conclude this article and outline our future work.

2. RELATED WORK

These past two decades have been marked by abundant literature on the integrated optimization of two or more links of the supply chain. Thomas and Griffin (1996) [1] categorize these works into three categories: supplier-customer integration, production-distribution integration, and distribution-inventory management integration. This last category, known as IRP, is defined by Cambell et al. (1998) as the optimization of the repeated distribution of a product from a central warehouse to a set of retailers or customers over a finite or infinite planning horizon.

Golden et al. (1984) addressed the IRP problem by adopting a different heuristic, which involves determining the locations to visit each day and generating the corresponding routes. They considered a finite-capacity plant supplying a set of customers primarily characterized by a stochastic demand rate. The objective of their model was to ensure the distribution of products from the warehouse to customers while minimizing transportation and storage costs and reducing stockouts at the end of periods [2]. They formulated their problem as a nonlinear integer program and proposed an approximation method to solve it.

The Stochastic Inventory Routing Problem (SIRP) has been modeled using a Markov decision process in various works, such as those by Minkoff (1993), Kleywegy et al. (2002), and Kleywegt et al. (2004). Some research works model multi-level supply chains based on classic inventory management policies, including:

Policies with continuous review: these include policies (s, S), (s, Q), and (S-1, S), where the stock level is continuously monitored.

Policies with periodic review: these include policies (R, s, S), (R, S), and (R, Q), where the stock level is periodically reviewed. The parameters s, Q, S, R, and T represent the reorder point, order quantity, replenishment level, and review interval, respectively. Hence, the concept of emergency transshipment is found as a form of cooperation among retailers to cope with stockout situations [3].

Research has shown that inventory management strategies are effective in enhancing the efficiency of the supply chain, reducing overall supply chain costs (Achabal et al., 2000; Jung, Chang, & Park, 2005) [1], and that communication among supply chain members is beneficial for constructing better forecasts and improving competitiveness (Vachon, Halley, Beaulieu, 2009).

An optimization model is a decision support tool with the objective of finding the optimal and feasible solution to a given problem. In this context, the objective function is either maximized or minimized by adjusting decision variables subject to constraints (Ding et al., 2020). The objective function of the model provides insights into the economic aspect of the network, including sales, profit, or cost. From a logistics perspective, it's the decisions the model makes to achieve such an objective value that we are interested in

[7]. Optimization, taking into account final customer demand forecasts, is being developed in planning to optimize inventory, enhance production stability, resource planning, and improve customer service rates (Bhaskaran, 1998; Grave et al., 1998; Towill, 1991).

In the study conducted by Rabenasolo et al. (2000), it was observed that, for a nonzero transportation lead time, changes in demand alone generate a significant variation in stock at the interface between the two links.

In a supply chain, many works have focused on forecasting customer demand. Most of them use time series processing algorithms. Traditional demand forecasting methods, based on elementary and/or advanced statistical methods, have proven useful in numerous cases (Kuo, 2001) [1]. In some cases, more recent statistical methods, such as genetic algorithms based on fuzzy networks, have improved the results provided by traditional methods, and research has shown that simple statistical methods tend to amplify the bullwhip effect (Carbonneau et al., 2008). Given the previous constraints, ARIMA is widely used for forecasting customer demand.

The ARIMA model considers data as primarily a function of time and is preferred when seeking the general trend of variations without considering factors influencing demand [8]. Researchers have developed more sophisticated methods than traditional tools. For example, FerBar et al. (2009) achieved better results with exponential smoothing by incorporating a wavelet denoising step.

In the 1990s, Leung (1995) identified artificial neural networks (ANN) as potentially suitable for demand forecasting in the supply chain. The architecture known as the multilayer perceptron with backpropagation is commonly used (Beccali, Cellura, Lo Brano, & Marvuglia, 2004) [9]. Another modification frequently mentioned in the literature to improve decision-making processes is to integrate a fuzzy component (Chang et al., 2011; Efendigil et al., 2009) to avoid binary yes/no decisions (Barajas & Agard, 2004) [6]. After being properly configured with historical data, artificial neural networks (ANN) can be used to accurately approximate any measurable function.

Su and Wong (2008) studied a dynamic and stochastic sizing problem under the bullwhip effect. The authors proposed an optimization solution using the ant colony method while discussing the solution's quality and the relationship between the bullwhip effect and the replenishment cycle [10]. To reduce the bullwhip effect and net inventory amplification, Devika et al. (2016) applied a new multi-objective hybrid evolutionary optimization approach (MOHES).

Pai and Lin (2005) provided outstanding results even in the presence of noise or missing information. They combined support vector machines (SVM) with ARIMA and found that the hybrid model performed better than SVM or ARIMA [1].

Other research enriches the concept of supply chain performance by demonstrating that the operational performance of decentralized supply chains can be significantly enhanced through a collaborative forecasting improvement source (ARIMA with genetic algorithm) [10]. Brahim and Bensaadie (2023) propose two new hybrid deep learning models applied to multi-step ahead time series forecasting. This includes a combination of Convolutional Neural Network, Gated Recurrent Unit network (GRU), and Deep Temporal Convolutional Network (TCN) to improve time series forecasting accuracy.

Time series analysis and forecasting have not yet reached their zenith and remain a domain dominated by statistical models today. Recent results from the M4 forecasting competition indicate that both worlds are converging, giving rise to hybrid approaches that combine statistical models and machine/deep learning models. It is worth noting that in the case of building global models when hundreds of interconnected time series are available, DeepAR enables the creation of highly performing models for both point forecasts and probabilistic forecasts [12].

Our work focuses on the modeling, dynamic optimization, and prediction of the complex supply chain using historical order data and artificial intelligence methods, including:

- The combination of ARIMA and DeepAR for demand forecasting.
- Multilayer neural networks for supply chain optimization

3 BASIC MODELING PRINCIPLES

3.1 Time Series with ARIMA

The ARIMA model is defined by three components: AR(p), I(d), and MA(q). The notation ARIMA(p, d, q) is used to describe the complete ARIMA model, which is given by the following equation:

$$y_t' = u + \sum_{i=1}^{P} \Phi_i y_{t-i}' + \epsilon_t + \sum_{i=1}^{P} \theta_i \epsilon_{t-i}$$

Where :

- $y_t{}'$ is the stationary time series after differencing of order d.
- μ is the mean of the time series.
- ϕ_i are the autoregressive coefficients for lags i=1,2,...,p
- θ_i are the moving average coefficients for lags i=1,2,...,q
- ϵ_t is the white noise term

The methodology of Box and Jenkins allows determining the suitable ARIMA model for modeling a time series, so it is about building a model that best captures the behavior of a time series. This methodology suggests four steps:

- Identification
- Estimation
- Validation
- Model Forecasting

Identification involves specifying the three parameters p, d, q of the ARIMA(p, d, q) model. Model stationarity is first tested through a graphical study, autocorrelation, and an augmented Dickey-Fuller test. If the series is not stationary, it should be transformed into stationarity. The order of integration "d" is the number of times the original series has been differenced to achieve stationarity. Autocorrelations and partial autocorrelations are used to estimate the orders p and q for the AR and MA models:

- Partial autocorrelations are zero beyond order p.
- Autocorrelations are zero beyond order q.

Simple Dickey Fuller Test: Dickey and Fuller were the first to provide a set of formal statistical tools for detecting the presence of a unit root in a first-order autoregressive process. This test is used to test the hypothesis:

$$\begin{cases} H_0: The\ model\ has\ a\ unit\ root \\ H_1: The\ model\ does\ not\ have\ a\ unit\ root \end{cases}$$

In 1981, Dickey and Fuller extended this testing procedure to autoregressive processes of order p, and it became the Augmented Dickey-Fuller test (ADF). ACF and PACF plots (Figure 1) can be analyzed to specify the values of the seasonal model by examining the correlation of seasonal lags.

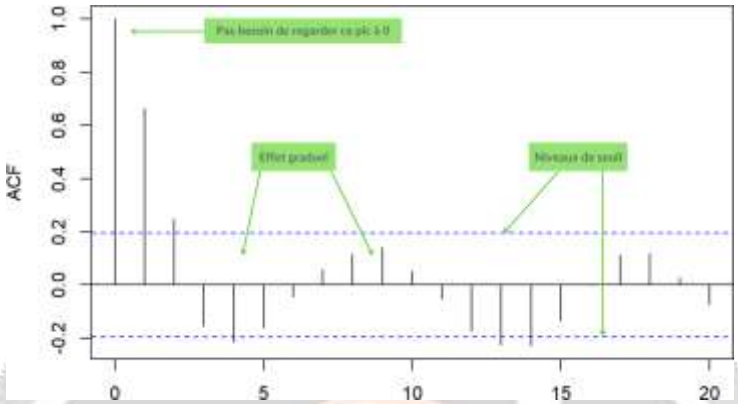

Figure 1. ACF and PACF plots

Often it is not easy to determine a single model. After estimating the different ARIMA models, it is now necessary to validate these models, using, on the one hand, tests of parameter significance for the coefficients, and, on the other hand, an analysis of the estimated residuals. The model coefficients must be significantly different from zero, which is achieved by using the classic Student's t-test.

The null hypothesis is rejected : $H_0: \theta_j = 0, if\ |tc| > \left|\tau^{\alpha}_{T-q}\right|, where\ |tc| = |\frac{\hat{\theta}}{\hat{\sigma}_{\hat{\phi}}}|$

To ensure that the obtained models are valid, it is necessary to check that the estimated residuals follow a white noise pattern. The information criterion used is:

1. Akaike (1969) : $AIC(p,q) = \log(\hat{\sigma}_{\epsilon}^2) + 2\frac{p+q}{T}$
2. Schwarz (1977) : $BIC(p,q) = \log(\hat{\sigma}_{\epsilon}^2) + (p+q)\frac{logT}{T}$
3. Hannan_Quinn(1979) : $\varphi(p,q) = \log(\hat{\sigma}_{\epsilon}^2) + (p+q)c\left(\frac{\log(\log(T))}{T}\right), avec\ \epsilon > 2$

The selection of an ARIMA model (p, d, q) results from the following four main steps:

Step 1: Identifying the initial values of the orders p, d, and q is based on the study of simple and partial correlograms.

Step 2: Estimating the parameters θi and φi is based on maximizing likelihood functions using iterative procedures.

Step 3: Once the parameters are estimated, the estimation results should be examined with reference to tests for the significance of the parameters and the quality of the residuals (absence of autocorrelation).

Step 4: The choice of the most appropriate model among all estimated models is made based on two criteria: Akaike (AIC) and Schwartz (SC), which measure the quality of the model's approximation to reality.

Step 5 (Forecasts): Once the ARIMA model is estimated and validated, it can be used to make forecasts for future time series data.

3.2. Time Series by DeepAR

The first model capable of working natively on multiple time series is DeepAR [13], a recurrent autoregressive network developed by Amazon. DeepAR uses LSTM networks to create probabilistic outputs. DeepAR leverages LSTMs to parameterize a Gaussian likelihood function, that is, to estimate the parameters ϑ = (μ, σ) (mean and standard deviation) of the Gaussian function.

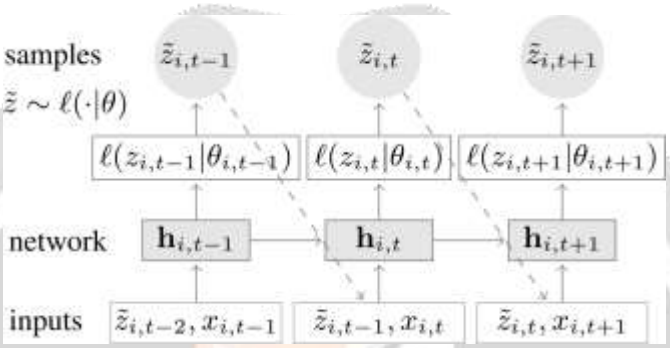

Figure 2. Inference during the training phase

a) First, the LSTM cell takes the input covariates xi,t from the current time step t and the target variable zi,t-1 from the previous time step t-1. The LSTM also receives the hidden state hi,t-1 from the previous time step.
b) Next, the LSTM cell outputs its hidden state hi,t, which is passed to the next step.
c) The values μ and σ are indirectly calculated from hi,t and become the parameters of a Gaussian likelihood function, denoted as ϑ = (μ, c_t).
d) This concludes training step t. The current target value zi and the hidden state hi,t are passed to the next time step, and the training process continues.

In statistics, the parameters μ and σ are typically estimated using Maximum Likelihood Estimation (MLE) formulas, which are derived by differentiating the likelihood function. Instead, LSTM and the two dense layers derive these parameters based on the model input. The process of estimating μ and σ is straightforward:

a) First, the LSTM computes its hidden state hi,t.
b) Then, hi,t passes through a dense layer W to calculate the mean μ.
c) Similarly, the same hi,t passes through a second dense layer W to calculate the standard deviation σ.
d) Now we have μ and σ. The model creates a Gaussian distribution with these parameters and takes a sample. Then, the model checks how close this sample is to the actual observation zi,t.
e) The LSTM weights and the two dense layers W and W are trained during backpropagation

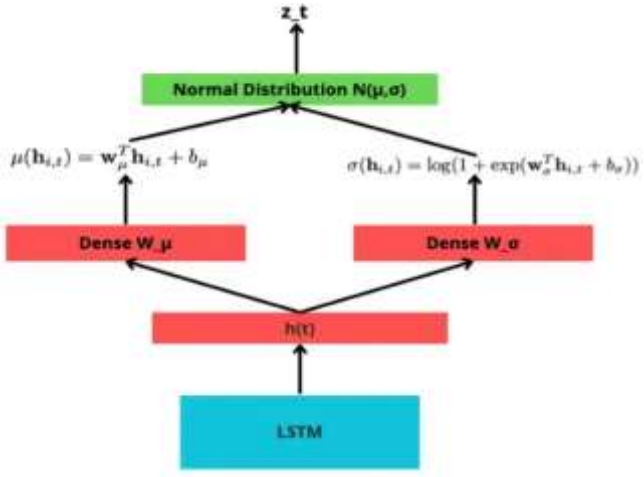

Figure 3. Process of estimating μ, σ

3.3. Le perceptron multicouche

La structure du perceptron multicouche utilisé est présentée par la figure 4 et est composée de neurones interconnectés en trois couches successives.

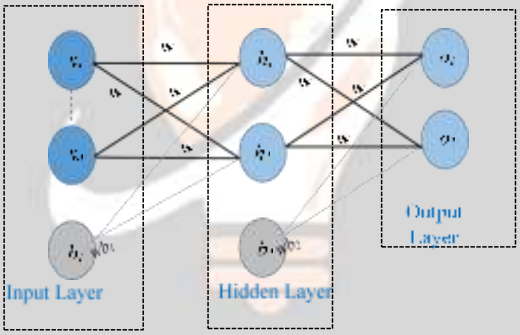

Figure 4. The structure of the multilayer perceptron

The first layer is composed of 'pass-through' neurons that perform no computation but simply distribute their inputs to all neurons in the next layer, called the hidden layer. The neurons in the hidden layer receive the n_0 inputs $\{x_1^0, 000, x_n^0\}$ from the input layer with associated weights $\{w_{i_1}^0, 000, w_{i_{n0}}^0\}$. This neuron starts by computing the weighted sum of its n_0 inputs:

$$Z_i^1 = \sum_{h=1}^{n_0} w_{ih}^1 * x_h^0 + b_i^l \qquad (1)$$

Where b_i^1 is a bias (or threshold g(l))

The output of the hidden neuron is obtained by transforming the sum (1) through the activation function g:

$$x_i^1 = g(z_i^1). \qquad (2)$$

Although many activation functions have been proposed, the function g(.) is typically the hyperbolic tangent:

$$g(x) = \frac{2}{1+e^{-2x}} - 1 = \frac{1-e^{-2x}}{1+e^{-2x}} \qquad (3)$$

The neuron in the last layer (or output layer) uses a linear activation function and therefore performs a simple weighting of its inputs:

$$Z = \sum_{i=i}^{n_l} w_i^2 * x_i^1 + b \quad (4)$$

Where w_i^2 are the weights connecting the outputs of hidden neurons to the output neuron, and b is the bias of the output neuron.

3.4 Model Performance Evaluation

This step involves evaluating the models by comparing the difference between estimated values and actual values. The model ultimately chosen is the one that minimizes one of the criteria using T observations

- Mean Absolute Error : $MAE = \frac{1}{T}\sum_{t=1}^{T}|\epsilon_t|$.
- Mean Squared Error : $MSE = \frac{1}{T}\sum_{t=1}^{T}\epsilon_t^2$
- Root Mean Square Error : $RMSE = \sqrt{\frac{1}{T}\sum_{t=1}^{T}\epsilon_t^2}$
- Mean Absolute Percent Error : $MAPE = \frac{100}{T}\sum_{t=1}^{T}\left|\frac{\epsilon_t}{X_t}\right|$
- Nash-Sutcliffe efficiency : $NSE = 1 - \frac{\sum_i^N (Y_i^{obs} - (Y_i^{sim})^2}{\sum_i^N (Y_i^{obs} - (Y_i^{mean\ obs})^2}$
- Determination coefficient: $R^2 = \frac{Cov^2(Y_i^{sim}, Y_i^{obs})}{V(Y^{sim}) * V(Y^{obs})}$

The lower the value of these criteria, the closer the estimated model is to the observations

4. CONTRIBUTION

4.1. Representation of the Studied Supply Chain

The optimization model is inspired by those presented by Cordeau (2014)[14] in the "VEGESUPPLY" project. The model leverages collaborative logistics, a powerful tool, to reduce costs and enhance competitiveness

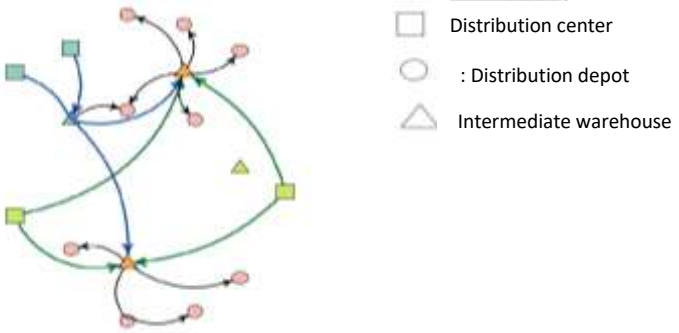

Figure IV. 1 Distribution network (Olivier Péton and Cordeau 2014)

The figure illustrates the role of ICCs (intermediate warehouses). Central depots ship their logistics units to one of the nearest ICCs. Subsequently, customer orders are consolidated in the ICCs and distributed via the nearest distribution routes. The mileage gain is made possible by the presence of multiple ICCs, which avoids unwanted detours. Furthermore, to ensure local services, the ICCs also manage orders for customers residing nearby.

The capacities of the warehouses are limited, as are the number of vehicles used:

- Each customer can be delivered by at most one vehicle in a period.
- A vehicle can make at most one tour per period.
- The demand of each customer in a period is deterministic and can be satisfied either from stock or by delivery in the same period.
- Costs to be considered include warehouse storage fees, fixed costs associated with vehicle use, and fixed costs associated with each distribution warehouse.

The objective is to determine, for each period, the quantities to be delivered to each warehouse, whether it is an intermediate warehouse or a distribution warehouse, as well as the transportation services provided, in order to minimize the total cost.

4.2 Modeling the Supply Chain Process

The current instant demand at warehouse i is modeled by

$$D_{i,t} = u_i + \epsilon_{i,t}$$

Where $D_{i,t}$ is the demand at instant t for the item with $1 \leq i \leq N$, and a represents the trend. For the entire distribution supply chain, the equation becomes:

$$D_t = \sum_{j=0}^{k} u_j t^j + \epsilon_t$$

Where: k is the number of distribution warehouses, u_i, σ^2 are the mean and standard deviation of the demand for product $i, \epsilon_{i,t}$ is a series of independently and identically distributed random values.

Selling is modeled by $V_{i,t} = \min(D_{i,t}, S_{i,t})$ where $V_{i,t}$ represents the sales made at time t, and $S_{i,t}$ is the available stock of product i at warehouse j.

First, we present the initial solutions determined in the studied system. Stock models for distribution depots and the distribution center are then established to determine the best stock parameters for all sites and minimize the total cost at the system level.

To address the total cost minimization for the distribution depots, we adopt the heuristic proposed by Ehrhardt and Mosier (1984)[99], which involves determining s_i^0 *and* S_i^0 Model (R,s,S) based on a backorder cost.

Let G_k and z_k be such that:

$$G_k = 1.3(u_k)^{0.494}\left(\frac{a_k}{h_k}\right)^{0.506}\left(1+\frac{\sigma_{k,R_k}^2}{u_k^2}\right)^{0.116}$$

$$\text{Et } z_k = \sqrt{G_k \frac{h_k}{\sigma_{k,R_k} b_k}}$$

So, for any vertex k, k ϵ $\{1, \dots K\}$:

$$s_k = 0.973\, u_{k,R_k} + \sigma_{k,R_k}\left(\frac{0.183}{z_k} + 1.063 - 2.192 z_k\right)$$

If $\frac{G_k}{u_k} > 1.5$ alors =

$$s_k^0 = G_k \;\; et \; S_k^0 = \; s_k^0 + G_k$$

Else

$$S_0 = u_{k,L+1} + k\sigma_{k,L+1}$$

$$s_k^0 = min\{G_k, S_0\}$$

$$S_k^0 = min\{s_k + G_k, S_0\}$$

With $u_{k,R_k} = (R_k + 1)h_k \;\; et \; \sigma_{k,R_k}^2 = (R_k + 1)\sigma_k^2$

For the distribution center, we adopt the concept of the echelon stock proposed by Clark and Scarf (1960). Therefore, the initial replenishment level in the distribution center is equal to the echelon stock of the entire system. To ensure that k is satisfactory, the initial stock position of the distribution center, which is limited to customer demand only during $L_{CD} + R$, is :

$$S^0 = Nu_i(L_{CD} + R) + \sum_{i=1}^{N} S_i^0$$

During each period t, we have the following sequence of events:

a) Arrival of deliveries from suppliers:

$$S_t^b = S_{t-1}^b + Q_{t-L_{CD}-1}$$

b) Receipt of orders from distribution centers:

$$D_t \; = \; \sum_{i=1}^{N} Y_{i,t}$$

c) Stock reservation and order placement with the supplier:

$$P_t^a = P_{t-1}^b - \sum_{i=1}^{N} QL_{i,t} \; et \;\; S_t^a = S_t^b - \sum_{i=1}^{N} QL_{i,t}$$

d) Stock situation is updated as follows:

$$S_{t+1}^b = S_t^a \;\; , \; t=1\dots T$$
$$P_{t+1}^b = P_t^a \;\; , t = 1..T$$

e) Redeployment of maximum stock quantities:

The available quantity for redeployment, Q_{dep} to avoid overstock in a distribution depot is determined in a way that ensures the remaining stock guarantees a minimum service level θ_k^{min} min over the same duration L_k:

$$Q_{red} = V_k L(Z_{\alpha_{max}} - Z_{\alpha_{min}})$$

The total management cost per period is given by the formula below:

$$\Gamma = \ minimiser \sum_{i=1}^{T}\left(\sum_{k=1}^{K} R.v_{kt} + \sum_{j=1}^{J}\sum_{k=1}^{K} c_j^C x_{jkt} + \sum_{j=1}^{J} h_j^C I_{jt}^C\right)$$

And the total stock of the product in the supply chain in this case becomes :

$$\mathrm{E^P} = \ Minimize \ \sum_{i=1}^{T}\left(h^P E_t^P + \sum_{k=1}^{K} R.v_{kt} + \sum_{j=1}^{J}\sum_{k=1}^{K} c_j^C x_{jkt} + \sum_{j=1}^{J}(h_j^C - h^P) I_{jt}^C\right)$$

Under the following constraints:

$$\sum_{j=1}^{J} q_{jkt}^C \leq W.v_{kt} \quad \forall k \in K, \forall t \in T$$

$$q_{jkt}^C \leq W.x_{jkt} \quad \forall j \in K \quad \forall t \in\in T$$

$$\sum_{k=1}^{K} x_{jkt} \leq 1 \quad \forall j \in J, \ \forall t \in\in T$$

4.2 Resolution Methodogy

We will present a methodology to address the challenge of modeling and optimizing the complex supply chain using artificial intelligence, with the aim of reducing overall logistics costs and optimizing inventory levels while ensuring optimal service to our customers.

This methodology involves implementing a multi-model representation for forecasting customer orders, as well as a model for the supply chain. We will detail each model structure.

For forecasting customer orders, we will use two different model structures:

- The ARIMA model, which is a statistically-based technique evaluated in various applications, demonstrating its proven performance.
- Amazon's DeepAR model, which is a deep learning technique that enables the development of a global forecasting model based on a recurrent autoregressive neural network.

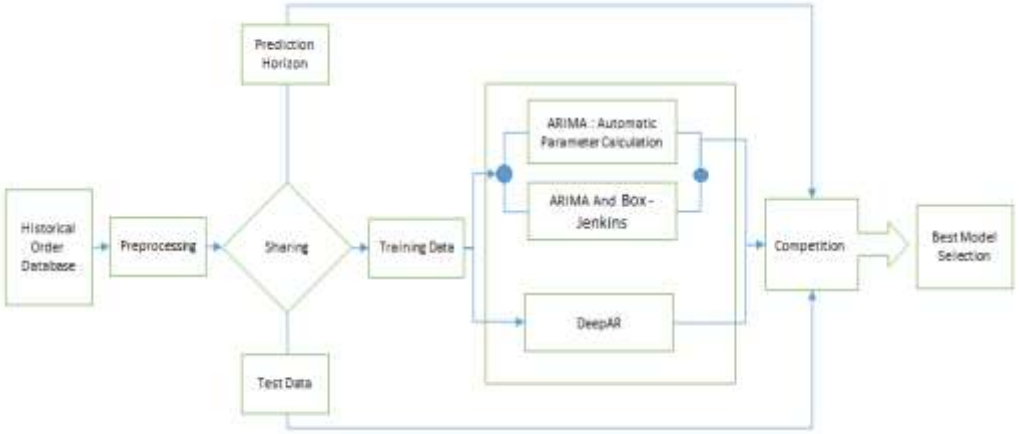

Figure 5. Methodology for Selecting the Best Customer Order Forecasting Model

From the historical order database, the first phase involves data preprocessing, followed by the separation of available data into a training set and a test set. We automatically calculate the ARIMA model parameters p, d, and q, then compare the results with those displayed in the ACF and PACF diagrams.

For the ARIMA model, the "p" and "q" parameters and the differencing term "d," we train the ARIMA structure by varying the parameters within their ranges using the Step-Wise algorithm presented in (Hyndman & Khandakar, 2008) and then identify the model with the lowest AIC (Akaike Information Criterion).

Based on these input data, the DeepAR algorithm forms a model that learns an approximation of these processes and uses it to predict how customer orders evolve. First, DeepAR trains the model by random sampling. Each training example consists of two adjacent windows: a context window and a prediction window. The size of each of these windows is fixed and imposed by the context_length parameter, which controls how far back the network can look into the past, and the prediction_length parameter, which controls how far into the future predictions can extend.

Then, for the second step, the best structures of each model type are evaluated to choose the one that performs the best. The testing step involves competing these trained models on the test data to select the best structure (Winmodel) by minimizing the test error calculated for each series (eteste) according to the following equation.

$$Win\ model = argmin k(\varepsilon test\ k)$$

Where k is the number of tests conducted, and eteste is the error measure used for each model. The competition criterion used is the root mean square error (RMSE).

Once the best model between ARIMA and DeepAR is selected, for each product, we retrain this model on all the data (including the data used in the test set during the model competition) to refine its hyperparameters for better product modeling. Finally, forecasts are calculated and sent to the optimization model according to the requested horizon (a 12-month period). To make predictions at different horizons, we used a recursive prediction strategy.

Figure 5 illustrates our approach to model selection for each product. This model selection step is the most crucial in the prediction process.

4.2 Optimization Structure and Algorithm

For the supply chain optimization model, we use a Multilayer Perceptron (MLP) neural network with the Newton SQP method. Here is the algorithm:

SQP Algorithm with Equality and Inequality Constraints

Data : $f : R^n \to R, g: R^n \to R^q, h : R^n \to R^p$ $Differentiables, x_0$ $initial\ point,$
$\lambda_0 \in R_+^q$ $et\ u_0 \in R^p$ $initial\ multipliers, \varepsilon > 0$ $requested\ accuracy.$
Output: an approximation x of the solution.*
1. $k := 0;$
2. $while \ \| \nabla L(x^k; \lambda^k \| > \varepsilon,$
$(a) Résoudre\ le\ sous - problème\ quadratique$:

$$(QP_k) \begin{cases} \min\limits_{d \in R^n} \nabla f(x_k)^T d + \frac{1}{2} d^T H_k d \\ s.t \ g_j(x^k) + \nabla g_j(x^k)^T d = 0 \,, i = 1, \dots, q, \\ h_i(x^k) + \nabla h_i(x^k)^T d = 0 \,, \ i = 1, \dots, p. \end{cases}$$

$and\ obtain\ the\ primal\ solution\ d_k and\ the\ multipliers\ \lambda' and\ u'$
$associated\ with\ the\ inequality\ constraints\ and\ equality\ constraints, respectively.$
$(b)\ x^{k+1} = x_k + d_k \ ; \ \lambda^{k+1} = \lambda' \ ; u^{k+1} = u' \ ; k = k + 1$
3. $Return\ x_k$

One of the strategies that can be used to avoid overfitting is regularization, which employs the weight decay technique. This involves replacing the error with:

$$Erreur\big(y, f(x)\big) + \frac{\lambda}{2} \sum\nolimits_j w_j^2$$

Using the forecasted data of customer orders and instance parameters, we will create a training and validation database to determine the optimal value. Our initial database will be divided into three parts. The first part of the data will be used for model training, by product. The second part will be reserved for training the combination method with the Newton model. Finally, the last part will be used to evaluate the prediction results in order to determine the optimal value

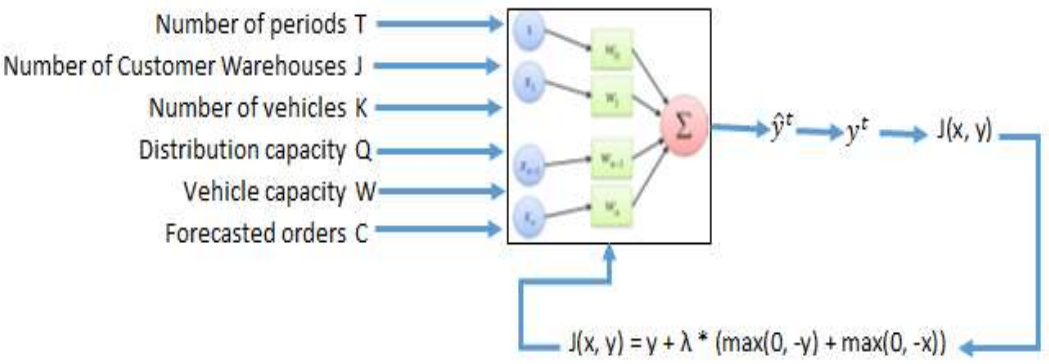

Figure 6. Structure and optimization algorithm

5. CONCLUSION

In this article, we studied a multi-site, multi-echelon supply chain consisting of a distribution center, intermediate warehouses, and distribution depots. The primary contribution of this study lies in selecting a high-performing model between ARIMA and DeepAR for predicting customer orders, thus creating a hybrid optimization model that combines the multilayer perceptron neural network and the Newton optimization model.

The use of the multilayer perceptron (MLP) neural network allowed us to consider the complexity of interactions between cost variables, transportation, stocks, and decision variables. This led to more precise and effective recommendations for managing our supply chain.

Incorporating predictive order data introduced an element of predictability, offering the opportunity to better anticipate demand fluctuations and adjust our decisions accordingly. At the same time, other cost-related variables were integrated to minimize expenses while maintaining an optimal level of service.

This optimization approach not only reduced operational costs but also optimized inventory levels, reduced transportation times, and increased customer satisfaction through improved product availability. Furthermore, it facilitated a more efficient and cost-effective supply chain, thereby enhancing our competitiveness in the market.

REFERENCES

1. **Mohamed Sameh Belaid**. *Modélisation des processus de prévision de la chaîne logistique pour l'amélioration des performances industrielles. Automatique.* Ecole nationale supérieure Mines- Télécom Lille Douai, 2022. Français. NNT : 2022MTLD0004. tel-03860429

2. **Imane Bouhaddou.** Vers une optimisation de la chaine logistique : proposition de modèles conceptuels basés sur le PLM (Product Lifecycle Management). Autre [cs.OH]. Université du Havre; Université Moulay Ismaïl (Meknès, Maroc), 2015. Français.

3. **A. Federgruen and M. Tzur**. The joint replenishment problem with the time-varying costs and demands : E_cient, asymptotic and epsilon-optimal solutions. Operations Research, 42 :1067 1086, 1994.

4. **Benbouteldja.M**, *Series temporelles par les réseaux de neurones et architecture optimale, mémoire de Magistere, Université des Sciences et de la Technologie Houari Boumediene, N◦ d'ordre : 13 / 2010*

5. **Mohamed Zied Babai**. *Politiques de pilotage de flux dans les chaînes logistiques : impact de l'utilisation des prévisions sur la gestion de stocks. Sciences de l'ingénieur [physics]. Ecole Centrale Paris, 2005. Français*.

6**. Houssam Moumouh**. *Synthèse d'un contrôleur prédictif auto adaptatif réglé par réseau de neurones artificiels. Systèmes et contrôle [cs.SY]. Normandie Université, 2021. Français.*

7. **Marwa Turki.** *Synthèse de contrôleurs prédictifs auto-adaptatifs pour l'optimisation des performances des systèmes. Automatique / Robotique. Normandie Université, Université de Rouen, 2018. Français.*

10. **M. Tlili, M. Moalla , Z. Bahroun , J.-P.** Campagne. *"Gestion de stocks avec transbordement dans un réseau de distribution multi-sites et multiéchelons". [In Proceedings of the 8e International Conference on Modeling and simulating (MOSIM'10)], Hammamet, Tunis, Tunisia.*

11. **Salinas, D., Flunkert, V., & Gasthaus, J. (2017**). *DeepAR: Probabilistic Forecasting with Autoregressive Recurrent Networks. Retrieved from http://arxiv.org/abs/1704.04110*

12**. Z.Jemai et R Kalai,** Analye d'un problème de tournée de vehicules avec Gestion de stock dans un contexte de stock consignation, 8é conference Internationale de Modélisation et Simulation « MOSIM » , Hammamet Tunisie 2010

13. **Paul W MURRAY**, Segmentation de données de livraison pour la prévision de la demande des clients

ECOLE POLYTECHNIQUE DE MONTREAL & CIRRELT 2015

14. **Fabien Lehuédé**. *Problèmes de Tournées de Véhicules avec Synchronisation et Optimisation Multicritère avec l'Intégrale de Choquet. Recherche opérationnelle [cs.RO]. Université de Nantes, 2015*