INDERSCIENCE PUBLISHERS
Linking academia, business and industry through research

European J. of Industrial Engineering

ISSN online: 1751-5262 - ISSN print: 1751-5254
https://www.inderscience.com/ejie

Data-driven imitation learning-based approach for order size determination in supply chains

Dony S. Kurian, V. Madhusudanan Pillai, J. Gautham, Akash Raut

DOI: 10.1504/EJIE.2023.10046611

Article History:

Received:	19 August 2021
Accepted:	16 February 2022
Published online:	01 May 2023

Data-driven imitation learning-based approach for order size determination in supply chains

Dony S. Kurian, V. Madhusudanan Pillai* and J. Gautham

Department of Mechanical Engineering,
National Institute of Technology Calicut,
NIT Campus, Calicut – 673601, Kerala, India
Email: donuskurian@gmail.com
Email: vmp@nitc.ac.in
Email: gauthamastro@gmail.com
*Corresponding author

Akash Raut

Department of Electrical and Electronics Engineering,
National Institute of Technology Calicut,
NIT Campus, Calicut – 673601, Kerala, India
Email: mr.akashraut@gmail.com

Abstract: Past studies have attempted to formulate the order decision-making behaviour of humans for inventory replenishment in dynamic stock management environments. This paper investigates whether a data-driven approach like machine learning can imitate the order size decisions of humans and consequently enhance supply chain performances. Accordingly, this paper proposes a supervised machine learning-based order size determination approach. The proposed approach is initially executed using the order decision data collected from a simulated stock management environment similar to the 'beer game'. Subsequent comparative analysis shows that the proposed approach successfully enhances all supply chain performance measures compared to other well-known ordering methods. Additionally, the proposed approach is validated on a retail case study to investigate its efficacy. This paper thus focuses on extending the past works reported in the literature by modelling human order decision-making as data-driven imitation learning and contributing to machine learning applications for order management. [Submitted: 19 August 2021; Accepted: 16 February 2022]

Keywords: supply chain; order size determination; machine learning; behavioural experiments; LightGBM; imitation learning; beer game.

Reference to this paper should be made as follows: Kurian, D.S., Pillai, V.M., Gautham, J. and Raut, A. (2023) 'Data-driven imitation learning-based approach for order size determination in supply chains', *European J. Industrial Engineering*, Vol. 17, No. 3, pp.379–407.

Biographical notes: Dony S. Kurian is a research scholar in the Department of Mechanical Engineering, National Institute of Technology Calicut, Kerala, India. He has his post-graduation in Industrial Engineering from the College of

Engineering Trivandrum, Kerala, India. He is also working as an analytics consultant at the EXL Services Pvt. Ltd. His current research interests include supply chain ordering management, inventory analytics and machine learning.

V. Madhusudanan Pillai is a Professor, Department of Mechanical Engineering, National Institute of Technology Calicut, Kerala, India. With more than 30 years of teaching experience and over 150 publications in international journals and conferences, and editorship of a book, he has developed several laboratory exercises and software packages in the area of manufacturing management and supply chain management. His research interest includes modelling of problems in supply chain operation simulation, cellular manufacturing systems, material requirements planning, scheduling, facility layout planning, inventory control, lean manufacturing, manpower planning – annualised hours, ergonomics and machine learning and blockchain applications in operations management.

J. Gautham is a BTech undergraduate in the Department of Mechanical Engineering, National Institute of Technology, Calicut, Kerala, India. His interests span artificial intelligence, deep learning, and scalable ledger technology. He is also working as the CEO and acting CTO of Polkadex, a blockchain company registered in Estonia.

Akash Raut is working as a senior technical member at the ADP. He has his undergraduate in the Department of Electrical and Electronics Engineering, National Institute of Technology, Calicut, Kerala, India. His areas of interests include research in the field of deep reinforcement learning, deep learning, applied artificial intelligence, computer vision and robotics.

1 Introduction

The decision of 'how much to order' for inventory replenishment is one of the routine decisions made by the supply chain managers. The replenishment decisions are critical since they directly impact the supply chain performance. In fact, if the order size is large, then inventory piles up, resulting in an increased holding cost, and if the order size is small, then stockout risk surges. Moreover, a poor order size decision will lead to an inefficient outcome called the bullwhip effect (BWE).

Despite its importance, the order size decision task for stock management, in reality, is complicated because managers are subjected to lagged feedbacks and the involvement of multiple interacting decision-makers in the supply chain system (Croson et al., 2014). Besides, the complexity of production and business process in organisations is increasing daily because of the global markets and uncertain environments. The decision process becomes more challenging when the supply chain managers are subjected to uncertain system parameters like customer demand and lead time (Ramaekers and Janssens, 2008).

In this paper, we investigate whether a data-driven approach like machine learning (ML) can imitate the replenishment ordering decisions of the supply chain decision-makers and subsequently enhance the supply chain performance in a dynamic stock management environment. This paper thus focuses on extending the past works reported in the literature by modelling order decision-making as data-driven imitation learning and contributing to ML applications for order management.

This paper continues in Section 2 with a brief survey of the related literature. Section 3 describes an overview of the methodology. Section 4 discusses the description of data collection for training the ML models. Section 5 presents the data cleaning phase of the proposed approach. Section 6 discusses the implementation and assessment of the ML methods. Section 7 is dedicated to the performance analysis of supply chains operated using the proposed approach. Section 8 validates the proposed approach using practical retail case data. Finally, discussions and conclusions are presented in Sections 9 and 10, respectively.

2 Related literature

Several studies have been carried out to investigate the ordering behaviour of humans operating in dynamic stock management environments. Almost all the studies related to stock management experiments were performed using the beer distribution game (BDG) or its variants (Yang et al., 2021). BDG is a role-play of four human players that simulates an industrial production and distribution system. At first, Sterman (1989) proposed an ordering rule based on the 'anchoring and adjustment' heuristic (Tversky and Kahneman, 1974), which imitates the decisions of the subjects exercising BDG. The proposed model (Sterman model, which we call from now onwards) follows a simple linear formula based on incoming order, expected demand, incoming and outgoing shipments, on-hand inventory, and current backorder to determine the order size. The principle behind the Sterman model is to:

1 replenish inventory when the expected level decreases

2 lessen the gap between target and on-hand inventory level

3 uphold a sufficient supply line of unfilled orders.

Following the footsteps of Sterman, several other researchers have revised the Sterman model for supply chain performance improvement. In one such research, Barlas and Özevin (2004) modified the linear Sterman model as a nonlinear formulation and showed that in many cases, the latter effectively accommodated the ordering dynamics of humans. Strozzi et al. (2007) and Liu et al. (2009) optimised the coefficients of the Sterman model using a genetic algorithm and particle swarm intelligence, respectively. They introduced minor variations to the classical demand pattern of BDG in their study and found that minimum supply chain cost is obtained when the Sterman model has optimised coefficients. Additionally, Rong et al. (2008) modified the Sterman model by incorporating a function that addressed a supply disruption and investigated the ordering behaviour of the players during the disruption. Similarly, Li and Yan (2015) have examined BWE variation and service level by including two more behaviour adjustments (to deal with uncertain demand and unsatisfied demand) in the Sterman model. On the other hand, Wright and Yuan (2008) investigated the effect of forecasting techniques in the Sterman model for alleviating the BWE.

Several research studies have also been conducted to study the effect of different parameter combinations of the Sterman model (Mosekilde and Laugesen, 2007; Macdonald et al., 2013; Edali and Yasarcan 2016; Gonçalves and Moshtari, 2021). The change in order size when on-hand and supply line inventory levels deviate from the desired targets are characterised by the adjustment (or decision) parameters of the

Sterman model. The different parameter combinations portray a given player's ordering behaviour in BDG.

The Sterman model has also been effectively used to investigate the behavioural causes of the BWE. The findings from previous studies point out that the players of BDG tend to underweight their supply line (Sterman, 1989; Croson and Donohue, 2006; Croson et al., 2014; Sarkar and Kumar, 2015, Perera et al., 2020). That is, players do not consider the orders placed in the previous periods which are not yet arrived when placing the orders for the next period.

A well-known industrial ordering rule commonly practised in multi-echelon supply chain systems is the order-up-to (OUT) policy (Costantino et al., 2015). Besides, the principle behind OUT policy, as per the Sterman model, is that OUT policy eases the discrepancy between target and on-hand inventory level by considering the supply line inventory (Tokar et al., 2012; Udenio et al., 2015).

Collectively, the above literature highlights the critical role in examining Sterman's ordering model in dynamic stock management environments. However, no research has been found in the literature that models human ordering behaviour by utilising the actual order decision data and investigating the subsequent supply chain performances. In the light of the stated research gap, this paper introduces data-driven imitation learning that aims to mimic human behaviour in a given order decision task. Imitation learning offers an implicit means of training an agent by sufficient demonstrations to map between observations and actions (Hussein et al., 2017). Recently, imitation learning principles have successfully been employed to a wide range of problems, including robotics (Tanwani et al., 2021), medical imaging (Kläser et al., 2021), power scheduling (Gao et al., 2021), logistics (Jothimurugan et al., 2021). Accordingly, this paper proposes a supervised ML-based order size determination (MLOD) approach that extends the Sterman model literature. The outcome of the proposed approach can perform as a data-driven decision support model for dynamic inventory replenishment systems.

Recently, reinforcement learning (RL) methods have been employed to determine optimal/near-optimal ordering strategies (Chaharsooghi et al., 2008; Oroojlooyjadid et al., 2021). RL algorithm primarily deals with artificial agents that interact with an environment, and the agent learns to make optimal/near-optimal decisions based on the rewards received (Sutton and Barto, 1998). Ultimately, the learning mechanism of RL involves explicit programming of the preferred environment and requires the design of the intricate reward functions specific to the task. Besides, the implementation of RL to practical business applications is a tedious effort and has to ignore complex states for simulating the same in a controlled environment (Dulac-Arnold et al., 2021). However, the present scope of the study follows a supervised ML methodology where agents are trained based on labelled data containing both predictor (independent) and dependent variables (Cunningham et al., 2008). As the data, including the order size decisions for different instances, are sufficiently available in the organisational databases, supervised learning can be directly applied for building order size determination models.

Like the Sterman model, the proposed order decision model does not optimise instead exercise control by determining the order size from a good order decision history. Since ML can learn and find hidden patterns from the data, an effectively trained ML model can determine the order size by imitating the best order size decisions of supply chain managers resulting in an improved supply chain performance. Therefore, this paper is different from previous studies as it does not follow a formula-based approach but instead follows a data-driven approach in determining the order size.

The contribution of this paper thus focuses on extending the Sterman model by modelling the human ordering behaviour as an ML approach. To the best of our knowledge, the application of supervised ML (using the order decision data) for order size determination has not been reported in the past literature. The MLOD will be a significant contribution for the managers of a stock management system as it can act as a decision support model in determining the order size. The present work also intends how a disruptive technology like ML can function in an application like order size determination. Organisations currently leverage the potential of data in various supply chain decision-making situations.

3 Overview of the methodology

This paper seeks to propose an MLOD approach for inventory management systems. Initially, the proposed approach is implemented on a traditional serial supply chain with four echelons similar to the experiment settings of the BDG. The data samples for training the ML models are collected from role-play experiments conducted during the past six years. The participants of the experiments are primarily students and scholars who have core knowledge in the particular domain. A detailed description of the experimentation and data collection are provided in Section 4. Once trained and tested, the supply chain operated based on the proposed approach is evaluated in terms of total supply chain cost (TSCC), BWE, and fill rate (FR). Furthermore, the proposed approach is applied to retail case data for checking its practical efficacy. A schematic diagram to illustrate the methodology is presented in Figure 1.

The first phase is a prerequisite for the ML model building and focuses on data cleaning of the order decision data. The decision-making in stock management systems is typically affected by feedback complexity and time delay. As a result, order size decisions of the managers in certain instances might be ambiguous (Bolton and Katok, 2008). Therefore, there is a need to eliminate those imprecise decision samples from the available data for delivering expert demonstrations for imitation learning. This paper adopts the inter quartile range (IQR) method, a univariate statistical technique to detect an outlier decision sample (Ilyas and Chu, 2019).

According to the well-known ‘no free lunch theorem’, a single ML method cannot give a precise solution to all problems (Wolpert and Macready, 1997). Each method has its mathematical fitting property, and, therefore, it must try out more than one method to find out the best one that effectively solves a problem. Accordingly, in the second phase, three ML methods are employed for order size determination and select the best method based on its predictive performance. The three ML methods considered are random forest (RF), light gradient boosting machine (LightGBM) and artificial neural network (ANN). RF and LightGBM belong to the decision tree ensemble methods utilised by many researchers in recent years to solve practical supply chain prediction problems (Weng et al., 2019; Vairagade et al., 2019). Likewise, ANN can capture the nonlinear behaviour of complex processes like order size determination and has been widely used for developing supervised prediction models (Jaipuria and Mahapatra, 2014; Bousqaoui et al., 2017). Since no prior work has been reported for order size determination using supervised ML, the abovementioned methods are applied to check their efficacy.

Figure 1 Schematic diagram of the methodology (see online version for colours)

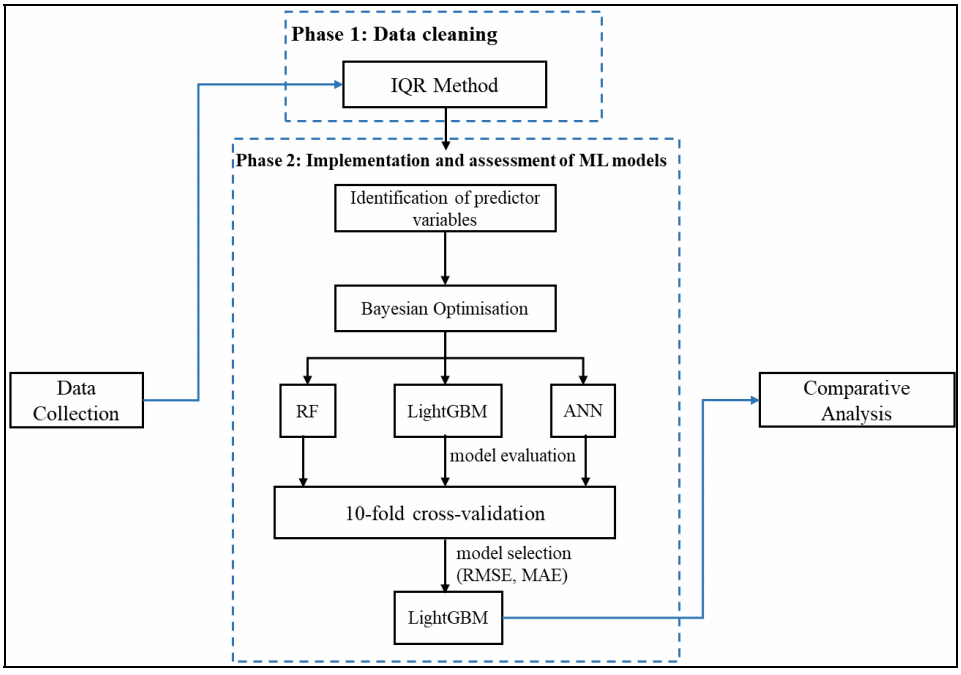
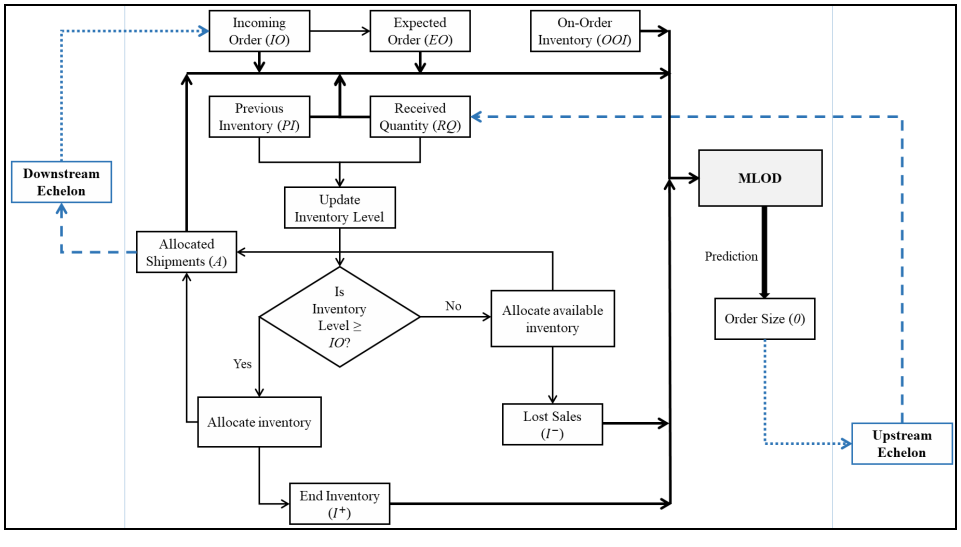


Figure 2 Inventory distribution logic at each echelon with the proposed method (see online version for colours)



Moreover, a proper hyper-parameter setting has considerable effects on the performance of any ML method. Consequently, in this paper, a Bayesian optimisation (BO) technique is employed to tune the hyper-parameters of RF, LightGBM and ANN. BO can effectively trade-off exploration and exploitation of the hyper-parameter space to

establish a configuration that best optimises a loss function (Snoek et al., 2012). Previous studies show that BO-based hyper-parameter tuning of ML methods is better than random search, grid search, and a manual search in terms of performance and speed (Hutter et al., 2015; Xia et al., 2017).

The best ML model identified in phase two is then embedded in the decision support systems of the supply chain echelons for order size determination. Figure 2 illustrates the inventory distribution logic at each supply chain echelon incorporating the proposed MLOD model. At any point in time, the ML model determines the replenishment order size based on the real-time inputs, and the determined order is subsequently placed to the upstream supplier. In addition, a comparative analysis of the supply chain performances is investigated. The performance analysis involves comparing supply chain operated by the MLOD model with supply chains operated by human decision-makers, supply chain operated by the Sterman model, and supply chain operated by the OUT policy.

4 Experimentation and data collection

The role-play experiments performed in this paper follow the standard procedure of conducting BDG similar to previous studies (Sterman, 1989; Croson and Donohue, 2006; Sarkar and Kumar, 2015) with few minor variations. The experiments are carried out on the Supply Chain Role-Play Game® (SCRPG) platform, an adaptation of BDG. The readers are recommended to refer to Pamulety and Pillai (2012), Pillai et al. (2014) and Pamulety and Pillai (2016) to know more about SCRPG and its background.

4.1 Experiment design

Consistent with BDG, we assume a serial supply chain with four echelons (i = 1 to 4) managed by human players. Each player in an echelon is assigned with the role of a retailer (i = 1), wholesaler (i = 2), distributor (i = 3) or a factory (i = 4) thereby forming a supply chain team. The players have to manage their assigned echelon independently by placing the orders upstream and meeting the demand from downstream over some specified game duration (t = 1, 2, …, T). The objective of each player is to minimise the accumulated inventory cost (IC) over the periods by maintaining inventory as low as possible while avoiding stockout situations.

In SCRPG, all players experience the following sequence of activities during each period (week):

1 delivery of allocated quantity from the upstream echelon (for factory, delivery from production plant)

2 incoming order from the downstream echelon (for retailer, the order is customer demand)

3 allocation to orders to fulfil the downstream demand with the available on-hand inventory (any shortage occurred is considered as lost sales)

4 placement of new replenishment order to the upstream echelon.

Before starting the actual experiments, players were briefed about the role-play environment. Additionally, a ‘role-specific’ trial game was performed to have a learning

experience for the players (Wu and Katok, 2006). The retailer echelon of the supply chain faces an unknown stationary customer demand, and it is not shared with other echelons. The customer demand for experiments is assumed from a normal distribution with a mean of 80 and a standard deviation of 10. The factory has an unlimited production capacity, and whatever quantity is demanded by the factory is replenished after production. Each player i places an order ($O_i(t)$) at the end of period t that will reach the upstream echelon $i + 1$ on $(t + l + 1)^{th}$ period (i.e., after an order lead time l). In contrast, each player i allocate $A_i(t)$ units to an order at the beginning of period t that will reach the downstream echelon $i - 1$ on $(t + k)^{th}$ period (i.e., after a delivery lead time k). If the available inventory meets the incoming order, the allocated units will be the same as the incoming order. Otherwise, available inventory on-hand will be allocated.

Furthermore, no communication between the players is allowed during the game, and each game is conducted for 25 weeks. An initial inventory of 190 units has been set for every echelon at the start of the game. A deterministic short lead time ($l = 0$ and $k = 1$ week) was considered and kept the same across all echelons to maintain moderate supply uncertainty. Furthermore, an asymmetric nature of unit cost consistent with prior studies (Daniel and Rajendran, 2005; Pillai et al., 2014) is considered for IC calculation. The unit cost per week for holding and lost sales for each echelon is shown in Table 1. In addition, we have entirely displayed the information like on-order inventory, on-hand inventory, orders placed to upstream, and end period inventory.

Table 1 Unit cost of holding and lost sales

Cost (in $ per unit per week)	*Retailer*	*Wholesaler*	*Distributor*	*Factory*
Holding cost (C_i^h)	5	4	3	1
Lost sales cost (C_i^s)	10	8	6	2

4.2 Supply chain performance measures

A supply chain performance depends on the individual decisions of its members, and the quality of a supply chain is evaluated by analysing its performance measures. The present study uses the BWE, FR, and total supply chain IC as the performance measures to evaluate the supply chains and assess echelon performance.

The BWE is the increase in order variability from downstream to upstream echelons in a supply chain. BWE_i of echelon i is statistically quantified by taking the ratio between order variance of an echelon (σ_i^2) and variance of customer demand at retailer echelon (σ_D^2) and can be presented as in equation (1) (Chen et al., 2000). The order variance of echelon (σ_i^2) can be calculated as per equation (2), where $\overline{O_i}$ represents the mean of the order placed by the echelon i over the game duration T. BWE of the echelons should be as low as possible, and BWE_4 (at factory) is considered as the quantified value of BWE for the whole supply chain.

$$BWE_i = \frac{\sigma_i^2}{\sigma_D^2} \tag{1}$$

$$\sigma_i^2 = \frac{\sum_{t=1}^{T}\left(O_i(t) - \overline{O_i}\right)^2}{T-1} \tag{2}$$

FR is an indication of the service level of supply chains. FR_i of echelon i is computed as the ratio of the demand met on time to the demand that arose (Chopra et al., 2013) and can be presented as in equation (3). FR_1 (at retailer) is considered as the supply chain FR (Zhao et al., 2001), and it should be as high as possible.

$$FR_i = \frac{\sum_{t=1}^{T} A_i(t)}{\sum_{t=1}^{T} IO_i(t)} \tag{3}$$

$IO_i(t) = O_{i-1}(t - l - 1)$ is the incoming order to an echelon i from the downstream $i - 1$, received after the order lead time l. For the retailer, incoming order is the customer demand.

IC of an echelon i is the sum of the costs incurred for holding the inventory ($I_i + (t)$) and lost sales ($I_i - (t)$) suffered by the echelon over the game duration T [see equation (4)]. Every player aims to minimise the accumulated IC at each echelon. The sum of ICs of all the echelons is the TSCC presented in equation (5).

$$IC_i = \sum_{t=1}^{T}\left[C_i^h \times I_i^+(t) + C_i^s \times I_i^-(t)\right] \tag{4}$$

$$TSCC = \sum_{i=1}^{4} IC_i \tag{5}$$

$$I_i^+(t) = \max\left\{0,\, I_i^+(t-1) + A_{i+1}(t-k) - IO_i(t)\right\} \tag{6}$$

$$I_i^-(t) = \max\left\{0,\, IO_i(t) - \left(I_i^+(t-1) + A_{i+1}(t-k)\right)\right\} \tag{7}$$

4.3 Data collection

The order decision details represent a player's order size decisions and related inventory information throughout the period of operation. Table 2 illustrates a typical order size decision detail of a retailer echelon. The order decision details for different players at each echelon are drawn from the role-play experiments conducted for the past six years (2014–2020). During this period, there was a total participation of 448 players forming 112 supply chain teams. The participated players were primarily students from undergraduate, post-graduate, management studies, and PhD. Before participating in the game, they had undergone a course on 'inventory and supply chain management'. According to the review carried out by Yang et al. (2021), it is noted that the performance of the 'professional' participants and 'student' participants in a BDG does not differ significantly, which support the authenticity of the data used in the current research.

Therefore, order decision details representing 112 retailer, 112 wholesaler, 112 distributor, and 112 factory are available for the study.

Though experiments have been carried out for 25 weeks, information from 1–22 weeks is considered as order decision details for a particular player. The end periods are excluded from the evaluation to remove the end game effects (Sterman, 1989). In fact, lagged feedbacks are not reflected in the end periods in contrast to the beginning periods. These order decision details are then aggregated for four supply chain echelons producing 2,464 (112 × 22) data points each. The performance measures described in the previous section, i.e., BWE (BWE_1), FR (FR_1), and IC (IC_1) of the sample retailer echelon ($i = 1$) as mentioned above, are 26.63, 0.96, and 2,575, respectively.

Table 2 Order decision detail of a retailer

Week	*Received shipments*	*Previous inventory*	*Incoming order*	*Expected demand*	*Allocated quantity*	*End inventory*	*Lost sales*	*On-order inventory*	*Order placed*
1	0	190	81	81	81	109	0	0	2
2	0	109	79	80	79	30	0	2	75
3	2	30	72	78	32	0	40	75	80
4	75	0	83	79	75	0	8	80	190
5	80	0	65	75	65	15	0	190	55
6	150	15	81	76	81	84	0	55	70
7	55	84	76	76	76	63	0	70	60
8	70	63	74	75	74	59	0	60	0
9	60	59	82	76	82	37	0	0	90
10	0	37	79	76	37	0	42	90	80
11	90	0	80	77	80	10	0	80	80
12	80	10	81	78	81	9	0	80	90
13	80	9	83	79	83	6	0	90	80
14	90	6	83	80	83	13	0	80	85
15	80	13	85	81	85	8	0	85	82
16	85	8	76	79	76	17	0	82	78
17	82	17	87	81	87	12	0	78	80
18	78	12	75	79	75	15	0	80	82
19	80	15	71	77	71	24	0	82	80
20	82	24	73	76	73	33	0	80	75
21	70	33	93	80	93	10	0	75	80
22	75	10	85	81	85	0	0	80	80

5 Phase 1: data cleaning

This section is a data cleaning/pre-processing step where imprecise order size decisions of the decision-makers are identified based on the IQR method. The IQR is defined as the difference between the third quartile (Q_3) and first quartile (Q_1) of the data. In the IQR method, an outlier is defined as those data points having values 1.5 of the IQR below the

first quartile (lower limit) or above the third quartile (upper limit) of the data, and constant factor 1.5 in the IQR method is determined statistically (Ilyas and Chu, 2019).

For instance, a sudden rise in the order size on week 4 (refer to Table 2) caused a hike in the ending inventory at week 6. Besides, this unexpected increase in order size triggers demand amplification and affects upstream members suffering from stockout. In addition, an immediate drop in order decision on week eight has led to a substantial stockout at week 10. These particular order sizes are ambiguous decisions of humans that seriously impact supply chain performance and have to be removed before imitation learning.

Figure 3 illustrates the box and whisker plot of the order size decisions placed by the respective supply chain echelons. Box and whisker plots graphically represent quartile ranges and spot potential outliers beyond the upper and lower limits as per the IQR method. The entire row samples of the imprecise order decisions are subsequently eliminated from the data set using the IQR method to lift the ML model's predictive performance. Table 3 shows the consolidated data points available for training the ML models. Each data point thus represents predictor variables available at the time of decision-making (order size determination) and the dependent variable (order size).

Figure 3 Box and whisker plot of the order placed data for different echelons (see online version for colours)

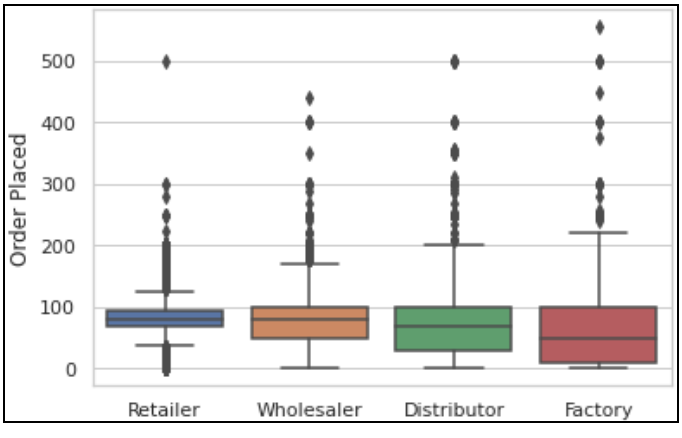


Table 3 Total number of data points available for training the ML model

	Retailer	*Wholesaler*	*Distributor*	*Factory*
Total number of data points collected	2,464	2,464	2,464	2,464
Number of decision samples identified as outliers	310	157	138	134
Total number of available data points after outlier removal	2,154	2,307	2,326	2,330

6 Phase 2: implementation and assessment of ML methods

This phase implements and evaluates three nonlinear ML methods to determine the order size for each echelon. Figure 1 provides the scheme followed for MLOD in phase 2.

Initially, predictor variables for determining the order size are identified. The best ML method for each echelon is then selected based on its predictive performance assessed from the outputs of a ten-fold cross-validation (CV) technique.

6.1 Predictor variables

The development of the MLOD model follows a supervised learning (regression) approach where dependent and predictor variables are well defined. Here, the dependent variable is order size ($O_i(t)$), determined based on a set of predictor variables. Predictor variables are variables used to predict another variable (dependent), and in this paper, they are identified from the previous literature that discusses on Sterman model and its variations. At any point of time, the participants of the stock management environment partially observe the available local inventory information for order placement (Chen, 1999), and it includes the following predictor variables,

1 Current period end inventory $(I_i^+(t))$.

2 Current period lost sales quantity $(I_i^-(t))$.

3 Previous period's end inventory ($PI_i(t)$) which is equal to $I_i^+(t-1)$.

4 Allocation of shipments to the downstream echelon ($A_i(t)$).

5 Shipments received from the upstream echelon ($RQ_i(t)$). For $l = 0$ and $k = 1$, $RQ_i(t) = A_{i+1}(t-1)$ and $RQ_4(t) = O_4(t-2)$.

6 Incoming orders received from the downstream ($IO_i(t)$).

7 On-order inventory ($OOI_i(t)$) is defined as an order placed in the previous periods but not yet received. For $l = 0$ and $k = 1$, $OOI_i(t) = O_i(t-1)$.

8 Expected incoming order for period $t + 1$ ($EO_i(t+1)$) which is calculated using simple exponential smoothing method [see equation (8)] with smoothing constant (θ) equal to 0.25.

$$EO_i(t+1) = \theta \times IO_i(t) + (1-\theta) \times EO_i(t) \tag{8}$$

A descriptive summary of predictor variables is presented in Table 4. From the descriptive analysis, it is evident that the summary values of each predictor variable, like minimum, maximum, mean and standard deviation, differ widely across each echelon. For example, the mean value of end period inventory $(I_i^+(t))$ varies from 28 at the retailer echelon to 133 at the factory echelon. Accordingly, order decision data of each echelon follows a distinct distribution, and ML models have to be built independently for each echelon.

The correlation coefficient values indicate the strength of the linear relationship between the dependent and different predictor variables and are illustrated in Figure 4. The correlation plot in Figure 4 is developed in a Python library called Seaborn, and the readers are requested to view the online version to distinguish the colours. The correlation coefficient ranges from –1 to +1, and these values indicate a strong negative (red) relationship to a strong positive (green) relationship. The figure shows that the correlation coefficient ranges mostly between [–0.5, 0.5], indicating a weak relationship

between order size and other predictor variables. Since there is no strong linear relationship between the dependent variable and predictor variables, nonlinear ML prediction models are used for determining the order size instead of generalised linear models.

Table 4 Descriptive summary of the predictor variables

Variables	*Retailer*				*Wholesaler*			
	Minimum	*Maximum*	*Mean*	*Standard deviation*	*Minimum*	*Maximum*	*Mean*	*Standard deviation*
RQ	0	175	64	30	0	200	59	38
PI	0	231	35	45	0	280	61	70
IO	55	99	80	8	0	250	76	32
EO	55	98	79	4	0	178	66	27
A	0	99	71	17	0	170	65	30
I^+	0	231	28	32	0	280	54	66
OOI	0	280	78	30	0	440	72	43
I^-	0	95	8	17	0	225	10	25
Variables	*Distributor*				*Factory*			
	Minimum	*Maximum*	*Mean*	*Standard deviation*	*Minimum*	*Maximum*	*Mean*	*Standard deviation*
RQ	0	300	57	50	0	300	58	62
PI	0	424	84	79	0	485	135	91
IO	0	400	73	43	0	356	67	53
EO	0	227	62	34	0	243	58	37
A	0	200	62	37	0	264	59	45
I^+	0	424	79	77	0	520	133	92
OOI	0	500	69	62	0	300	59	60
I^-	0	340	10	26	0	355	8	26

6.2 ML methods

RF is a tree-based parallel ensemble learning method for classification and regression (Breiman, 2001). RF method uses a collection of decision trees at the training stage and outputs the mean prediction of individual trees for regression. It utilises two key concepts rather than just averaging the prediction, viz.,

1 random sampling of training data points while building trees

2 random subsets of variables while splitting decision tree nodes.

This helps to produce a better predictive performance compared to constituent tree predictions.

LightGBM is a recent ML algorithm introduced by Microsoft in 2017 (Ke et al., 2017). LightGBM is used explicitly for classification and regression tasks of ML and operates based on the principle of gradient boosting decision tree (GBDT). Unlike RF, GBDT sequentially builds models until the minimisation of loss function becomes

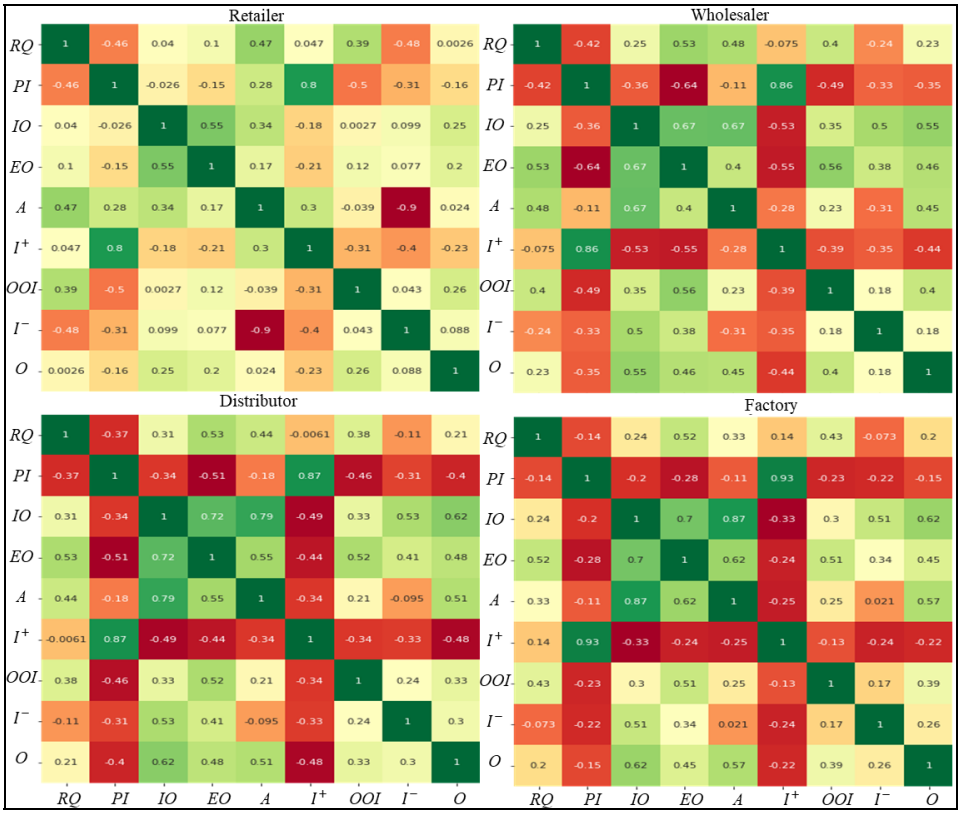
limited. LightGBM mainly utilises two novel techniques that make it more efficient and faster than conventional GBDT, and they are:

1 gradient-based one-side sampling (GOSS)

2 exclusive feature bundling (EFB).

GOSS reduces the number of data instances by focusing on under-trained instances (those with larger gradients) to estimate accurate information gain. Alternatively, as a result of EFB, exclusive features are bundled into a single feature, and the sparsity of the variable space is reduced. Consequently, LightGBM reduces the computational time for the training process and reduces memory consumption.

Figure 4 Correlation plot between dependent and predictor variables (see online version for colours)



ANN model is based on the structure and functions of a biological neural network and mimics the learning process performed by the brain (Bishop, 1995). ANN architecture typically contains one input layer, one or more hidden layers and an output layer. Each layer consists of a finite number of nodes/neurons, and adjacent layers are interconnected through nodes either partially or fully. Initially, data is fed into the input layer, which is then passed to the output layer through the nodes of different hidden layers (forward pass). The output of this pass is compared with the actual value, and a mean square error is calculated. The results are then passed back to the hidden layers to adjust the weight of

nodes through a process called back-propagation. The back-propagation algorithm optimises the weight of nodes by a series of forward and backward passes, thereby minimising the error between the forward pass output and the actual output.

6.3 ML implementation and assessment

A brief flow of the procedure explained in this section is illustrated in Figure 1 (phase 2). Initially, BO is performed to optimise the hyper-parameters of the ML methods. Out of the total available data points in each echelon dataset, 20% was chosen randomly for BO. For example, the retailer data set has 2,154 data points. Of which, 431 data points (20% of 2,154) are chosen randomly to find the optimal hyper-parameter combination based on BO. The hyper-parameter search space considered for three ML methods is shown in Table A1 of Appendix section. The final optimised hyper-parameter values after BO are shown in Table A2. Additionally, the hyper-parameter search space assumed in the study is intended to provide considerable model complexity to avoid high bias or underfitting phenomena.

Afterwards, ten-fold CV is employed using the entire data set to evaluate the predictive performance of the three ML methods. The execution of ten-fold CV involves splitting the entire data set randomly into ten partitions of equal size. For example, if the ten-fold CV was performed on the retailer data set, then 2,154 data points are, initially, divided into ten equal subsets (216 each). Then, one subset is held as a test set, and the remaining subsets (216 × 9 = 1,944) are combined to form the training set. ML methods are trained using the training set, and the predictive performance of the trained ML models is tested on the test set. The predictive performance of the ML models is assessed based on two generally applied error measures – root mean square error (RMSE) and mean absolute error (MAE).

$$RMSE_i = \sqrt{\frac{\sum_{j=1}^{N}\left(O_j - O_j^{'}\right)^2}{N}} \tag{9}$$

$$MAE_i = \frac{1}{N}\sum_{j=1}^{N}\left|O_j - O_j^{'}\right| \tag{10}$$

where O_j and $O_j^{'}$ are the actual and predicted order size values, respectively. This validation process is repeated ten times, and each test set is used exactly once, giving ML methods an option to train on multiple train-test splits. Furthermore, the entire data sets corresponding to each supply chain echelon are initially standardised during the training process. The performance assessment of the ML methods is also performed based on the standardised values and rescaled to their original units after performance assessment. The training of ML methods, ten-fold CV, etc. are carried out with Sklearn, LightGBM and Keras packages on Python 3.7 and run on a machine with 3.70 GHz Intel Xeon CPU and 32 GB memory.

The mean and standard deviation of RMSE and MAE after ten-fold CV are the assessment values for comparing the final predictive performance of the trained ML models. To show the level of predictive performance variability of the ML models to its

mean performance, we define the coefficient of variation (cv_i), which is the ratio of standard deviation and mean of predictive performance values:

$$cv_i = \frac{standard\ deviation}{mean} \times 100 \tag{11}$$

The final predictive performances of three ML methods after ten-fold CV for each echelon are reported in Table 5. The basis of comparison is the coefficient of variation value of RMSE and MAE, and it should be as low as possible. For every echelon, the coefficient of variation values is least for LightGBM compared to RF and ANN. Although in some cases, the mean values of RMSE and MAE for RF and ANN are less than LightGBM, which has higher predictive variability than LightGBM. This indicates that predictive performance of RF and ANN is inconsistent in determining the order size, and the primary objective of CV, i.e., model generalisation (James et al., 2013), is achieved through the results of the LightGBM model.

Table 5 Predictive performance of ML methods after ten-fold CV

ML methods	*Mean RMSE*	*Standard deviation*	*cv*	*Mean MAE*	*Standard deviation*	*cv*
Retailer						
RF	1.0252	0.0847	8.14	0.7516	0.0597	8.55
LightGBM	0.9059	0.0377	4.16	0.6716	0.0278	4.14
ANN	0.9008	0.0475	5.32	0.6647	0.0294	4.47
Wholesaler						
RF	0.993	0.0596	5.97	0.7647	0.0493	6.47
LightGBM	0.779	0.0449	5.76	0.5711	0.0323	5.66
ANN	0.7666	0.072	9.37	0.5642	0.0496	8.98
Distributor						
RF	0.7107	0.055	7.75	0.5181	0.0423	8.32
LightGBM	0.7148	0.0535	7.48	0.5218	0.031	5.94
ANN	0.7269	0.0616	8.47	0.5285	0.0344	6.56
Factory						
RF	0.7596	0.044	5.83	0.5687	0.0295	5.15
LightGBM	0.7083	0.0234	3.3	0.5139	0.0178	3.46
ANN	0.7308	0.0443	6.21	0.5533	0.0331	6.06

Furthermore, Figures 5–6 illustrate the learning curves that show the learning performance variation of the LightGBM model in terms of RMSE and MAE for the four echelons. It is observed that the learning curves also confirm the generalisability of the model's predictive performance because the LightGBM can fit better on the validation/ test set as the training set size increases. The figures show that as the training set size increases, the training error and validation error gap becomes narrower. Therefore, it appears that order size determined using the LightGBM model does not experience any overfitting or underfitting phenomenon, and the application of the model might lead to an improved supply chain performance, which is investigated in the next section.

Figure 5 Learning curve of LightGBM for RMSE (see online version for colours)

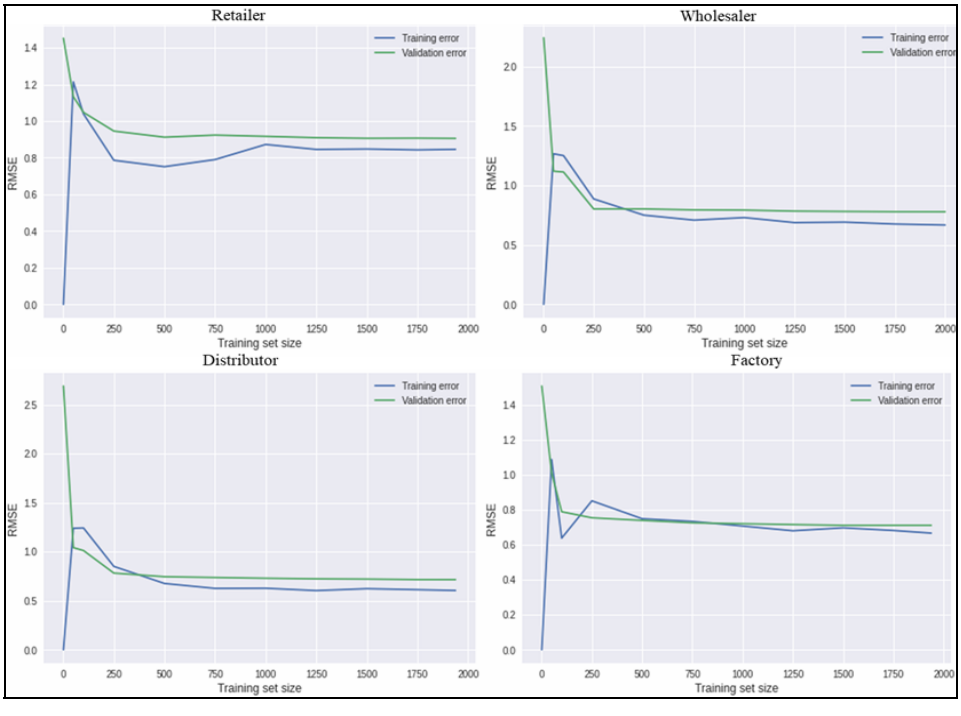
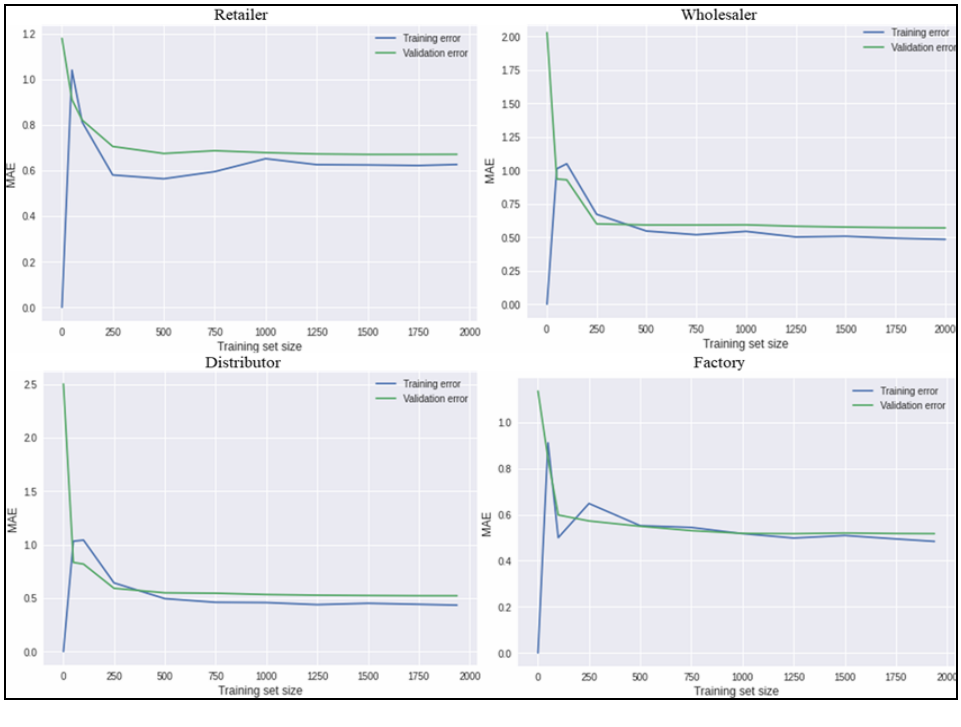


Figure 6 Learning curve of LightGBM for MAE (see online version for colours)



7 Comparative analysis

After identifying the best ML method for order size determination, the LightGBM model is trained using all the available data points. The trained ML model (LightGBM) is then embedded in the decision support systems of the supply chain echelons for order size determination. As in Figure 2, the ML model determines the order size based on the real-time values of the predictor variables, and the determined order is placed to the upstream supplier.

A comparative analysis of the supply chain performances is carried out to investigate the effectiveness of the MLOD model against the human decisions, Sterman model, and OUT policy. Accordingly, four-echelon serial supply chains are simulated for 25 weeks, with similar experimental assumptions and input conditions used for the role-play experiments. Supply chain performances and corresponding echelon performances are evaluated on four test problem instances with different customer demand samples (demand distribution remains the same) shown in Table 6.

Table 6 Test problem instances

Test problem 1	[82, 73, 92, 85, 90, 74, 83, 79, 85, 83, 91, 57, 72, 87, 77, 69, 74, 91, 84, 67, 101, 82, 72, 83, 97]
Test problem 2	[84, 92, 92, 77, 92, 73, 90, 90, 84, 84, 84, 72, 69, 84, 74, 84, 93, 81, 79, 55, 78, 70, 74, 95, 65]
Test problem 3	[61, 78, 62, 87, 81, 77, 78, 76, 74, 72, 69, 72, 56, 81, 77, 86, 58, 63, 88, 83, 68, 77, 71, 65, 75]
Test problem 4	[88, 86, 92, 70, 73, 74, 100, 82, 76, 89, 57, 79, 89, 67, 72, 94, 92, 82, 87, 70, 75, 77, 78, 77, 90]

The following four scenarios represent different order size determination approaches used for comparison:

1 Supply chains operated by human players: SCRPG is used to simulate this scenario. More than one supply chain team has participated in the different test problems. Due to human behaviour, different order sizes are possible at ordering instances of various replications of the test problems. Hence, more than one supply chain team is considered for simulation. For test problems 1, 2, 3, and 4, the supply chain teams participated is 6, 8, 6, and 8, respectively. The mean performance measures for each test problem are reported in Table 7. The best performing supply chain (human best) results are also reported in Table 7 and Figures 7–9.

2 Supply chain operated by the Sterman model: Microsoft Excel spreadsheet-based simulation is used to simulate this scenario. The order size calculated as per the Sterman model for echelon i is as follows:

$$O_i(t) = \max\left\{0,\, EO_i(t+1) + \alpha_I\left(I^* - I_i^+(t)\right) + \alpha_O\left(OOI^* - OOI_i(t)\right)\right\} \tag{12}$$

where α_I and α_O are the fractional adjustments for on-hand inventory and supply line, respectively. Also, I^* and OOI^* are the target or desired levels for on-hand and on-order inventory levels, respectively. The following are the parameter values used for the simulation of the Sterman model: $\alpha_I = 0.5$, $\alpha_O = 0.5$, $I^* = 80$ and $OOI^* = 80$ (mean demand), and these are set by assuming the stability of the steady-state system

(Macdonald et al., 2013). For instance, during simulation, the retailer echelon at week 9 for the test problem 1 has the following details: customer demand = 85, expected incoming order (for week 10) = 81, current inventory level = 27, and on-order inventory = 95. Accordingly, the order placed by the retailer as per equation (12) is equal to 100.

3 Supply chain operated by OUT policy: Microsoft Excel spreadsheet-based simulation is used to simulate this scenario. In OUT policy, an order is placed at each review period equal to the difference between the target (OUT level) inventory level and inventory position. Inventory position for lost sales order management environment is calculated as the sum of on-hand and on-order inventory (George and Pillai, 2021). Mathematically, the order size calculated as per the OUT policy for echelon *i* is as follows:

$$O_i(t) = \begin{cases} 0, & \text{if } IP_i(t) \geq OUT_i(t) \\ OUT_i(t) - IP_i(t), & \text{if } IP_i(t) < OUT_i(t) \end{cases} \tag{13}$$

where OUT level, $OUT_i(t) = 2 \times EO_i(t+1)$ and inventory position, $IP_i(t) = I_i^+(t) + OOI_i(t)$. For instance, during simulation, retailer echelon at week 9 for the test problem 1 has the following details: customer demand = 85, expected incoming order (for week 10) = 81, OUT level = 162, end inventory level = 7, on-order inventory = 64 and inventory position = 71. Accordingly, the order placed by the retailer as per equation (13) is equal to 91.

4 Supply chain operated by MLOD model (LightGBM): The Python program of the scenario is used for the order decision in the simulation. The input settings are the same as the ones used in other scenarios. Unlike other scenarios, the trained LightGBM model determines the order size based on the eight input variables. For instance, during simulation, retailer echelon at week 9 for the test problem 1 has the following details: received shipments = 73, previous inventory = 27, customer demand = 85, expected incoming order (for week 10) = 81, allocated quantity = 85, end inventory level = 15, lost sales = 0 and on-order inventory = 78. Accordingly, the order placed by the retailer at week 9 is equal to 88, which is the output from the trained LightGBM model using the above inputs.

Furthermore, supply chain performance measures are evaluated from the 4th week to the 22nd week, and the remaining periods are excluded to avoid the influences of warm-up and end-game effects. The supply chain performances obtained for the four test problems are reported in Table 7.

Upon investigating Table 7, it can be observed that the supply chain operated with the MLOD model has achieved the least TSCC compared to the other three scenarios. Similarly, the MLOD model has achieved to produce a low order rate variance ratio. Likewise, the supply chain FR obtained using the MLOD model is comparable with the highest FR of the Sterman model and the best human decision-makers.

Echelon performances of four scenarios in terms of IC, FR, and BWE are also investigated for all the test problems, and they are presented in Figures 7–9. Accumulated IC and order variance ratio (BWE) of each echelon in most cases are least for the supply chain operated by MLOD. Likewise, echelons operated by the MLOD model have satisfied the respective downstream demand well compared to the other three scenarios,

as illustrated in Figure 8. On the whole, comparative analysis indicates that the MLOD model successfully enhances supply chain performance compared to human decision-making or other analytical methods.

Table 7 Performance analysis of supply chains

	Supply chain performance measures	*Human*	*Human best*	*Sterman model*	*OUT policy*	*MLOD*
Test problem 1	Total supply chain cost	11,274	8,573	13,603	15,025	6,593
	Fill rate	0.98	0.99	1	0.63	0.99
	Bullwhip effect	19.62	6.64	8.46	23.1	6.6
Test problem 2	Total supply chain cost	16,579	10,475	14,085	15,870	6,875
	Fill rate	0.92	0.96	0.99	0.62	0.99
	Bullwhip effect	24.7	12.37	9.88	30.23	8.03
Test problem 3	Total supply chain cost	16,171	11,615	15,202	15,225	6,841
	Fill rate	0.91	0.99	1	0.61	0.99
	Bullwhip effect	23.2	2.42	11.16	24.2	7.88
Test problem 4	Total supply chain cost	16,724	8,780	13,482	16,028	6,708
	Fill rate	0.88	0.98	1	0.6	0.98
	Bullwhip effect	44.63	9.32	5.96	17.12	5.7

Figure 7 IC of each echelon under four scenarios for the four test problems (see online version for colours)

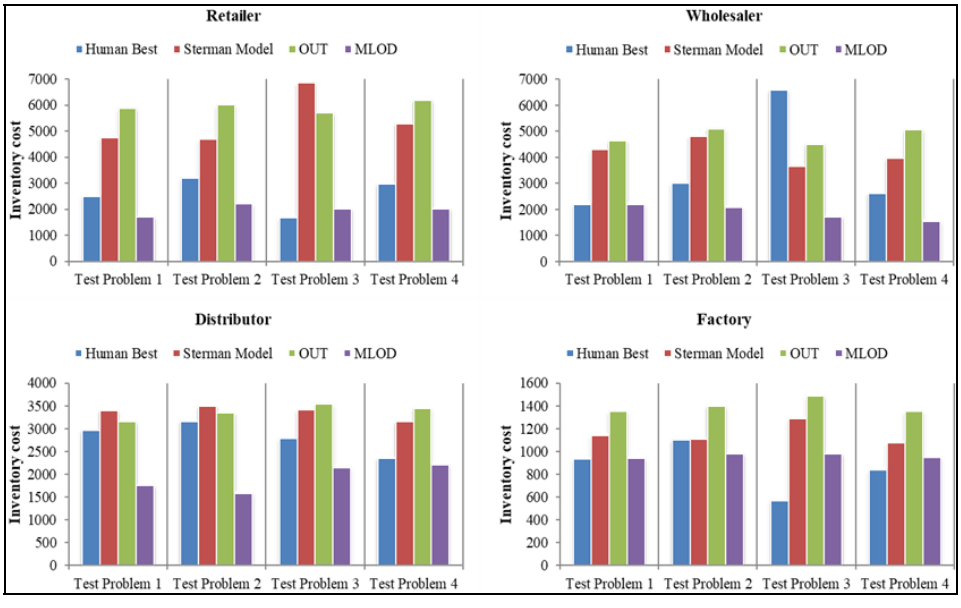


Figure 8 FR of each echelon under four scenarios for the four test problems (see online version for colours)

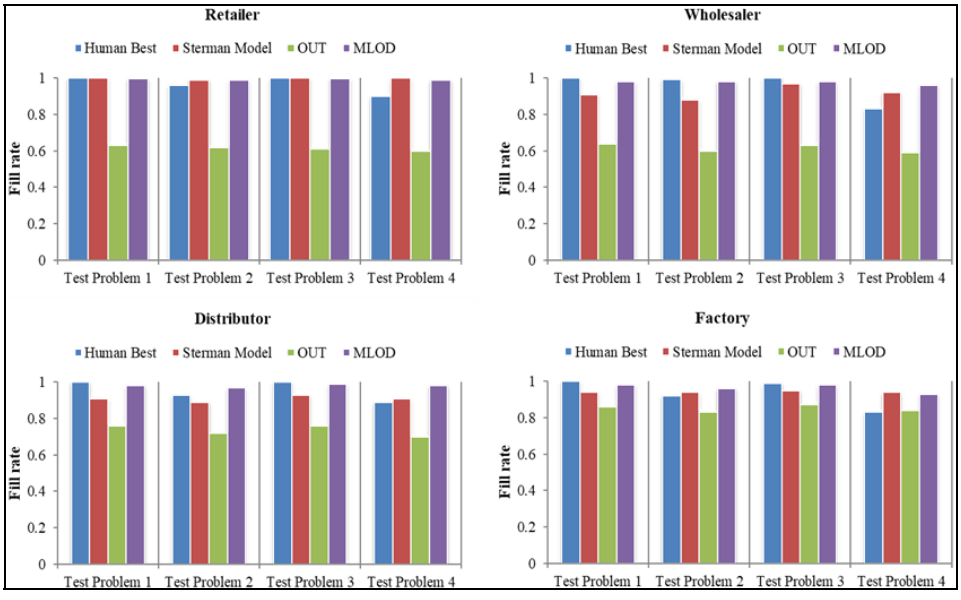
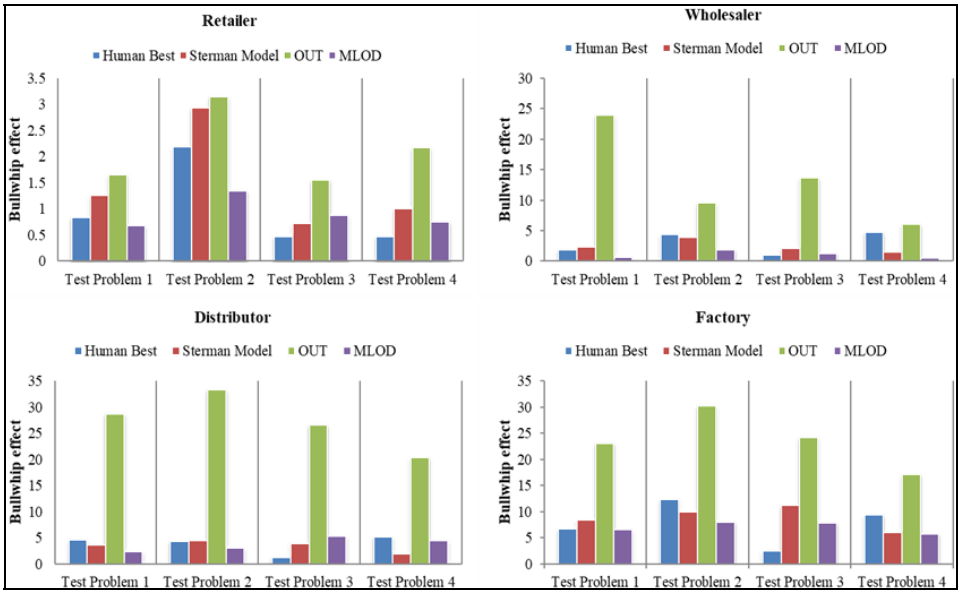


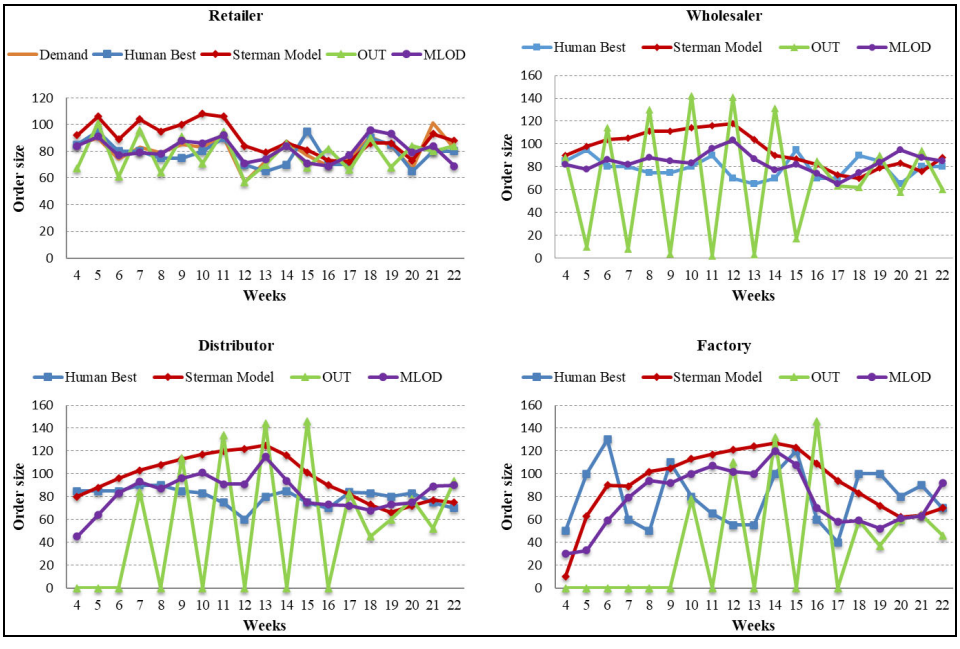
Figure 9 BWE of each echelon under four scenarios for the four test problems (see online version for colours)



Furthermore, Figure 10 illustrates the variation of orders placed by the different echelons under each scenario for test problem 1. In the case of the retailer echelon (see Figure 10 – retailer), orders determined by the MLOD model are almost closer to the incoming customer demand. Besides, compared to the other order management scenarios, the order

variability of the MLOD model is least across the echelons. Consequently, the application of imitation learning for order size determination produces lower and smoother oscillations, resulting in better performance of the MLOD method for order management. Similar observations could also be noted for other test problems.

Figure 10 Order placement of each echelon under four scenarios for the test problem 1 (see online version for colours)



8 Case study

This section tests the proposed approach on case data corresponding to an actual consumer durable available in a local retail hardware store. At present, order management at the retail store is entirely based on human decisions. As inferred in the previous section, our objective is to investigate any performance enhancement using the proposed approach. The weekly order decision details from 2015–2016 financial year (1st April of a year to 31st March of next year) till 2019–2020 are collected, and it constitutes a total record of 259 data samples. Data collected from 2015–2016 till 2018–2019 is considered as the training data set, and 2019–2020 financial year data is considered as the validation set. Furthermore, the data collected from the retail store have a mean weekly demand of 2,268 units of the item with a standard deviation of 258, which is very much different from the demand parameters assumed in the previous sections.

The retail store places an order at the end of every week, and it reaches the upstream Distributor without any delay (order lead time = 0). The upstream Distributor satisfies the store's demand within one week (delivery lead time = 1 week). Additionally, the unit cost per week for holding and lost sales for the retail store is estimated to be \$0.53 and \$5, respectively. This problem is a single echelon order management problem.

As a first step, the data cleaning phase of the proposed approach is introduced on the training set. The application of the IQR method discovered 16 ambiguous decision-making samples, and they are eliminated from the training data set. The three ML methods (RF, LightGBM, and ANN) are trained using the training data set in the second phase. Similar to experimental simulation, the predictive performance of the LightGBM model was observed to have the least RMSE and MAE values, which is then operated as the MLOD model.

Furthermore, a comparative performance analysis of the different decision-making scenarios is carried out on the 2019–2020 financial year data (validation set). Results of the performances are reported in Table 8. It can be observed that the implementation of the proposed MLOD model on real case data has also successfully reduced the IC and order variability as well as improved the service level. Consequently, it can be inferred that a significant reduction of 19.6% in IC could have been achieved in the 2019–2020 financial year at the retail store when the MLOD model is used instead of the present order management method.

Table 8 Performance analysis on case data

	Fill rate	*Order variance (customer demand variance = 18,183.71)*	*Annual holding cost*	*Annual lost sales cost*	*Annual inventory cost*
Human	0.9888	26,134.95	4,189.65	6,625	10,814
Sterman model	0.9884	5,400.04	2,284.5	6,850	9,134
OUT policy	0.9882	37,857.40	3,092.55	7,000	10,092
MLOD	0.9945	11,010.46	5,469.6	3,225	8,694.6

9 Discussion

This paper has focused on an imitation learning approach to model human decision-making in a dynamic stock management environment. More specifically, this paper proposes a supervised ML methodology for inventory order size determination. The outcome of the proposed approach is that it can perform as a data-driven decision support model that assists managers in dynamic inventory replenishment systems.

The investigation of our results proves that the trained MLOD model (LightGBM) produces or replicates orders that result in consistent echelon performance. This establishes the model generalisation characteristic of the ML algorithm (James et al., 2013). In addition, if the trained LightGBM model were experiencing any overfitting or underfitting, then the value of the performance measures would have deviated considerably across the test problems leading to inconsistent results. In brief, the MLOD model has learnt to place best order size decisions of the decision-makers in the dynamic stock management environments.

From a system point of view, supply chain performance measures are enhanced when expert decision-makers manage each echelon since supply chain performance depends on the ordering behaviour of the individual decision-makers (Bolton and Katok, 2008). Likewise, it can be realised that when the supply chain echelons are operated by the MLOD model, as in scenario 4, the replicated order size at each echelon is good enough to produce a better supply chain performance. Furthermore, results reported in the study

have practical implications so that the proposed approach can be used independently by each echelon where information is not transparent.

The supply chain performance measures found in Table 7 and echelon performance measures illustrated in Figures 7–9 demonstrate that the supply chain operated by MLOD has consistent performance over the other three approaches in all the test problems. For example, the TSCC of the proposed MLOD approach across the four test problems obtained in our simulation experiment is in the range [6,593, 6,875]. In fact, the range of this performance measure is minimal compared to the other three approaches. Meanwhile, performance measures of human ordering, the Sterman model, and OUT policy differ considerably across the test problems making those approaches less reliable than the MLOD model. For example, the mean TSCC of the human decision-makers across the four test problems is in the range [11,274, 16,724]. Furthermore, the application of the proposed approach in the retail case data also shows its practical efficacy.

Concerning BWE results, it can be noted that BWE persists in supply chains. The demand amplification is visible across the echelons in all the scenarios (see Figure 10), and its severity is more at the upstream factory echelon (see Figures 9 and 10). This is similar to the typical supply chain nature described in the literature (Croson and Donohue, 2006). However, BWE severity in supply chains operated using the proposed method is less than humans, Sterman model and OUT policy (see Table 7 for comparison). It can be inferred that the MLOD model tries to smooth the replenishment order dynamically, as seen in Figure 9, without over-ordering or under-ordering as it learns the best order size pattern of the decision-makers suitable for each time-step. Meanwhile, the Sterman model and OUT policy at each time-step follow a fixed ordering rule [see equations (12) and (13)] without learning any smoothing pattern that resulted in demand amplification at the upstream echelons.

Furthermore, the incoming customer demand at each time-step is satisfied more or less entirely in supply chains operated using the MLOD model (refer to Tables 7–8 and Figure 8). The supply chain FR values of the proposed scenario across the four test problems range 98%–99%, comparable to scenario two and best human decision-makers, and are better than scenario three. As mentioned previously, the trained ML model makes replenishment orders based on the order size pattern of the decision-makers who can satisfy the incoming demands completely. Additionally, the variation of the FR across the echelons (as in Figure 8) is minimal, confirming the ML model's consistency in different test problem instances. The supply chains operated by the humans and the Sterman model has a comparatively higher FR than the OUT policy since these scenarios tend to place orders more than required. This validates the typical ordering behaviour of humans to keep an increased inventory in stock to avoid the risk of stockout (Nienhaus et al., 2006).

10 Conclusions

This paper has reported the entire process of ML model building, including data cleaning, model training, performance assessment, and its application in order management. The proposed approach is a generic methodology that can be applied to any stock management environment with a good amount of order decision history. Nowadays, as the data containing the order size decisions for different instances are sufficiently

available in the organisational databases, the imitation learning principles can be directly applied to build order size determination models.

The data considered in this study represent human decision-making behaviour that has biases and judgments in order decision-making. The implementation of ML has successfully captured specific trends and patterns in these data, thereby arriving at a model for well-ordered behaviour. From the performance analysis, it can be concluded that rather than following a fixed ordering rule like the Sterman model, the proposed approach has learnt to imitate the best order size decisions of the decision-makers and subsequently resulted in an enhanced supply chain performance. This paper thus extends the Sterman model by modelling the human ordering behaviour as an imitation learning approach. The outcome of the proposed approach is a data-driven model that organisations prefer these days as they can leverage the potential of data. This is the key idea put forth through this paper.

The scope of this paper is limited by training the ML model only to a single demand distribution and the application of the proposed model to a single retail case. This limitation also suggests the directions for future research by experimenting and training the ML models for different demand distributions and seasonal changes since the present work proves how ML models can be deployed for order size determination based on the underlying data. The present work also assumes that the order decision data of humans have cognitive biases. That means the order size determined from the proposed approach is still not optimal since it has not been trained on correct input/output pairs. Additionally, this paper has used the order decision data of humans to train the ML models. The comparative analysis shows that the Sterman model and OUT policy provide a better performance, at least in some echelons, than the echelons operated by humans. Therefore, an implication for the future study is that imitation learning can be applied to order decision data obtained from the simulated samples of the Sterman model and OUT policy. The present work thus has thrown a prospect of using ML for order size determination in dynamic stock management environments. It provides contributions to the literature with opportunities for future research.

References

Barlas, Y. and Özevin, M.G. (2004) 'Analysis of stock management gaming experiments and alternative ordering formulations', *Systems Research and Behavioral Science: The Official Journal of the International Federation for Systems Research*, Vol. 21, No. 4, pp.439–470.

Bishop, C.M. (1995) *Neural Networks for Pattern Recognition*, Clarendon Press, Oxford.

Bolton, G.E. and Katok, E. (2008) 'Learning by doing in the newsvendor problem: a laboratory investigation of the role of experience and feedback', *Manufacturing & Service Operations Management*, Vol. 10, No. 3, pp.519–538.

Bousqaoui, H., Achchab, S. and Tikito, K. (2017) 'Machine learning applications in supply chains: an emphasis on neural network applications', in *IEEE 2017: 3rd International Conference of Cloud Computing Technologies and Applications*, Rabat, Morocco, pp.1–7.

Breiman, L. (2001) 'Random forests', *Machine Learning*, Vol. 45, No. 1, pp.5–32.

Chaharsooghi, S.K., Heydari, J. and Zegordi, S.H. (2008) 'A reinforcement learning model for supply chain ordering management: an application to the beer game', *Decision Support Systems*, Vol. 45, No. 4, pp.949–959.

Chen, F. (1999) 'Decentralized supply chains subject to information delays', *Management Science*, Vol. 45, No. 8, pp.1076–1090.

Chen, F., Drezner, Z., Ryan, J.K. and Simchi-Levi, D. (2000) 'Quantifying the bullwhip effect in a simple supply chain: the impact of forecasting, lead times, and information', *Management Science*, Vol. 46, No. 3, pp.436–443.

Chopra, S., Meindl, P. and Kalra, D.V. (2013) *Supply Chain Management: Strategy, Planning, and Operation*, Pearson, Boston, MA.

Costantino, F., Di Gravio, G., Shaban, A. and Tronci, M. (2015) 'The impact of information sharing on ordering policies to improve supply chain performances', *Computers & Industrial Engineering*, Vol. 82, pp.127–142.

Croson, R. and Donohue, K. (2006) 'Behavioral causes of the bullwhip effect and the observed value of inventory information', *Management Science*, Vol. 52, No. 3, pp.323–336.

Croson, R., Donohue, K., Katok, E. and Sterman, J. (2014) 'Order stability in supply chains: coordination risk and the role of coordination stock', *Production and Operations Management*, Vol. 23, No. 2, pp.176–196.

Cunningham, P., Cord, M. and Delany, S.J. (2008) 'Supervised learning', in *Machine Learning Techniques for Multimedia*, pp.21–49, Berlin, Heidelberg.

Daniel, J.S.R. and Rajendran, C. (2005) 'A simulation-based genetic algorithm for inventory optimization in a serial supply chain', *International Transactions in Operational Research*, Vol. 12, No. 1, pp.101–127.

Dulac-Arnold, G., Levine, N., Mankowitz, D.J., Li, J., Paduraru, C., Gowal, S. and Hester, T. (2021) 'Challenges of real-world reinforcement learning: definitions, benchmarks and analysis', *Machine Learning*, Vol. 110, pp.2419–2468.

Edali, M. and Yasarcan, H. (2016) 'Results of a beer game experiment: should a manager always behave according to the book?', *Complexity*, Vol. 21, No. S1, pp.190–199.

Gao, S., Xiang, C., Yu, M., Tan, K.T. and Lee, T.H. (2021) 'Online optimal power scheduling of a microgrid via imitation learning', *IEEE Transactions on Smart Grid*, Vol. 13, No. 2, pp.861–876.

George, J. and Pillai, V.M. (2021) 'Evaluation of inventory replenishment policies on supply chain performance with grey relational analysis', *International Journal of Integrated Supply Management*, Vol. 14, No. 2, pp.197–227.

Gonçalves, P. and Moshtari, M. (2021) 'The impact of information visibility on ordering dynamics in a supply chain: a behavioral perspective', *System Dynamics Review*, Vol. 37, Nos. 2–3, pp.126–154.

Hussein, A., Gaber, M.M., Elyan, E. and Jayne, C. (2017) 'Imitation learning: a survey of learning methods', *ACM Computing Surveys (CSUR)*, Vol. 50, No. 2, pp.1–35.

Hutter, F., Lücke, J. and Schmidt-Thieme, L. (2015) 'Beyond manual tuning of hyperparameters', *KI-Künstliche Intelligenz*, Vol. 29, No. 4, pp.329–337.

Ilyas, I.F. and Chu, X. (2019) *Data Cleaning*, ACM Books, New York.

Jaipuria, S. and Mahapatra, S.S. (2014) 'An improved demand forecasting method to reduce bullwhip effect in supply chains', *Expert Systems with Applications*, Vol. 41, No. 5, pp.2395–2408.

James, G., Witten, D., Hastie, T. and Tibshirani, R. (2013) *An Introduction to Statistical Learning*, Springer, New York.

Jothimurugan, K., Andrews, M., Lee, J. and Maggi, L. (2021) *Learning Algorithms for Regenerative Stopping Problems with Applications to Shipping Consolidation in Logistics*, arXiv preprint arXiv:2105.02318.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q. and Liu, T.Y. (2017) 'LightGBM: a highly efficient gradient boosting decision tree', *Proceedings of the 31st International Conference on Advances in Neural Information Processing Systems*, CA, USA, December, Vol. 30, pp.3146–3154.

Kläser, K., Varsavsky, T., Markiewicz, P., Vercauteren, T., Hammers, A., Atkinson, D. and Ourselin, S. (2021) 'Imitation learning for improved 3D PET/MR attenuation correction', *Medical Image Analysis*, Vol. 71, p.102079.

Li, Z. and Yan, G. (2015) 'Exploring different order decision behaviors with bullwhip effect and service level measures in supply chain system', *Discrete Dynamics in Nature and Society*, Vol. 2015, pp.1–8.

Liu, H., Howley, E. and Duggan, J. (2009) 'Optimisation of the beer distribution game with complex customer demand patterns', in *IEEE 2009: Proceedings of the Congress on Evolutionary Computation*, Trondheim, Norway, pp.2638–2645.

Macdonald, J.R., Frommer, I.D. and Karaesmen, I.Z. (2013) 'Decision making in the beer game and supply chain performance', *Operations Management Research*, Vol. 6, Nos. 3–4, pp.119–126.

Mosekilde, E. and Laugesen, J.L. (2007) 'Nonlinear dynamic phenomena in the beer model', *System Dynamics Review: The Journal of the System Dynamics Society*, Vol. 23, Nos. 2–3, pp.229–252.

Nienhaus, J., Ziegenbein, A. and Schönsleben, P. (2006) 'How human behaviour amplifies the bullwhip effect. A study based on the beer distribution game online', *Production Planning & Control*, Vol. 17, No. 6, pp.547–557.

Oroojlooyjadid, A., Nazari, M., Snyder, L.V. and Takáč, M. (2021) 'A deep q-network for the beer game: deep reinforcement learning for inventory optimization', *Manufacturing & Service Operations Management*, Vol. 24, No. 1, pp.285–304.

Pamulety, T. and Pillai, V. (2012) 'Performance analysis of supply chains under customer demand information sharing using role play game', *International Journal of Industrial Engineering Computations*, Vol. 3, No. 3, pp.337–346.

Pamulety, T. and Pillai, V.M. (2016) 'Effect of customer demand information sharing on a four-stage serial supply chain performance: an experimental study', *Uncertain Supply Chain Management*, Vol. 4, No. 1, pp.1–16.

Perera, H.N., Fahimnia, B. and Tokar, T. (2020) 'Inventory and ordering decisions: a systematic review on research driven through behavioral experiments', *International Journal of Operations & Production Management*, Vol. 40, Nos. 7/8, pp.997–1039.

Pillai, V.M., Lavanya, K. and Pamulety, T.C. (2014) 'Bullwhip effect analysis using supply chain role play game and ranking of supply chains', *International Journal of Procurement Management*, Vol. 7, No. 3, pp.299–315.

Ramaekers, K. and Janssens, G.K. (2008) 'On the choice of a demand distribution for inventory management models', *European Journal of Industrial Engineering*, Vol. 2, No. 4, pp.479–491.

Rong, Y., Shen, Z.J.M. and Snyder, L.V. (2008) 'The impact of ordering behavior on order-quantity variability: a study of forward and reverse bullwhip effects', *Flexible Services and Manufacturing Journal*, Vol. 20, No. 1, pp.95–124.

Sarkar, S. and Kumar, S. (2015) 'A behavioral experiment on inventory management with supply chain disruption', *International Journal of Production Economics*, Vol. 169, pp.169–178.

Snoek, J., Larochelle, H. and Adams, R.P. (2012) 'Practical Bayesian optimization of machine learning algorithms', *Proceedings of the 25th International Conference Advances in Neural Information Processing Systems*, NY, USA, December, Vol. 25, pp.2951–2959.

Sterman, J.D. (1989) 'Modeling managerial behavior: misperceptions of feedback in a dynamic decision making experiment', *Management Science*, Vol. 35, No. 3, pp.321–339.

Strozzi, F., Bosch, J. and Zaldívar, J.M. (2007) 'Beer game order policy optimization under changing customer demand', *Decision Support Systems*, Vol. 42, No. 4, pp.2153–2163.

Sutton, R.S. and Barto, A.G. (1998) *Introduction to Reinforcement Learning*, MIT Press, Cambridge.

Tanwani, A.K., Yan, A., Lee, J., Calinon, S. and Goldberg, K. (2021) 'Sequential robot imitation learning from observations', *The International Journal of Robotics Research*, Vol. 40, Nos. 10–11, pp.1306–1325.

Tokar, T., Aloysius, J.A. and Waller, M.A. (2012) 'Supply chain inventory replenishment: the debiasing effect of declarative knowledge', *Decision Sciences*, Vol. 43, No. 3, pp.525–546.

Tversky, A. and Kahneman, D. (1974) 'Judgment under uncertainty: heuristics and biases', *Science*, Vol. 185, No. 4157, pp.1124–1131.

Udenio, M., Fransoo, J.C. and Peels, R. (2015) 'Destocking, the bullwhip effect, and the credit crisis: empirical modeling of supply chain dynamics', *International Journal of Production Economics*, Vol. 160, pp.34–46.

Vairagade, N., Logofatu, D., Leon, F. and Muharemi, F. (2019) 'Demand forecasting using random forest and artificial neural network for supply chain management', in *International Conference on Computational Collective Intelligence, Lecture Notes in Computer Science*, Springer, Cham, Switzerland, pp.328–339.

Weng, T., Liu, W. and Xiao, J. (2019) 'Supply chain sales forecasting based on LightGBM and LSTM combination model', *Industrial Management & Data Systems*, Vol. 120, No. 2, pp.265–279.

Wolpert, D.H. and Macready, W.G. (1997) 'No free lunch theorems for optimization', *IEEE Transactions on Evolutionary Computation*, Vol. 1, No. 1, pp.67–82.

Wright, D. and Yuan, X. (2008) 'Mitigating the bullwhip effect by ordering policies and forecasting methods', *International Journal of Production Economics*, Vol. 113, No. 2, pp.587–597.

Wu, D.Y. and Katok, E. (2006) 'Learning, communication, and the bullwhip effect', *Journal of Operations Management*, Vol. 24, No. 6, pp.839–850.

Xia, Y., Liu, C., Li, Y. and Liu, N. (2017) 'A boosted decision tree approach using Bayesian hyper-parameter optimization for credit scoring', *Expert Systems with Applications*, Vol. 78, pp.225–241.

Yang, Y., Lin, J., Liu, G. and Zhou, L. (2021) 'The behavioural causes of bullwhip effect in supply chains: a systematic literature review', *International Journal of Production Economics*, Vol. 236, p.108120.

Zhao, X., Xie, J. and Lau, R.S.M. (2001) 'Improving the supply chain performance: use of forecasting models versus early order commitments', *International Journal of Production Research*, Vol. 39, No. 17, pp.3923–3939.

Appendix

Hyper-parameters of ML methods

The ML methods employed in this paper involve multiple hyper-parameters and have distinct search space. The hyper-parameters corresponding to the ML methods RF, LightGBM and ANN are defined according to the Python packages, Sklearn, LightGBM and Keras, respectively. Table A1 provides the search space of hyper-parameters used in the present study. BO is then employed to find the best hyper-parameters that result in maximum predictive performance. Table A2 provides the optimised values of hyper-parameters after BO, and these values are used for training the ML models.

Table A1 Search space for hyper-parameter optimisation

RF		*LightGBM*		*ANN*	
Hyper-parameters	*Range*	*Hyper-parameters*	*Range*	*Hyper-parameters*	*Range*
n_estimators	(50, 500)	num_leaves	(5, 500)	learning_rate	(0.001, 0.5)
max_depth	(1, 150)	max_depth	(2, 50)	no_of_hidden_layers	(1, 4)
min_samples_leaf	(2, 10)	learning_rate	(0.001, 0.5)	no_of_nodes_in_ hidden_layers	(1, 10)
max_features	(2, 7)	min_data_in_leaf	(10, 200)	dropout	(0, 0.8)
max_leaf_nodes	(2, 50)	bagging_fraction	(0.01, 1)	batch_size	(8, 128)
		feature_fraction	(0.01, 1)	epochs	(10, 250)

Table A2 Optimized values of hyper-parameters after BO

Hyperparameters	*Retailer*	*Wholesaler*	*Distributor*	*Factory*
Optimised hyper-parameters of RF after BO				
n_estimators	489	459	422	349
max_depth	40	35	38	42
min_samples_leaf	6	9	10	10
max_features	2	4	2	2
max_leaf_nodes	49	49	50	50
Optimised hyper-parameters of LightGBM after BO				
num_leaves	463	420	380	440
max_depth	47	50	50	25
learning_rate	0.48	0.45	0.5	0.38
min_data_in_leaf	19	10	12	20
bagging_fraction	0.58	0.5	0.45	0.42
feature_fraction	0.69	0.65	0.62	0.58
Optimised hyper-parameters of ANN after BO				
learning_rate	0.001	0.001	0.001	0.001
no_of_hidden_layers	1	1	1	1
no_of_nodes_in_hidden_layers	10	10	10	10
dropout	0	0	0	0
batch_size	24	24	24	24
epochs	250	250	250	250