*Proceedings of the 2025 Winter Simulation Conference*
*E. Azar, A. Djanatliev, A. Harper, C. Kogler, V. Ramamohan, A. Anagnostou, and S. J. E. Taylor, eds.*

# SALES PLANNING USING DATA FARMING IN TRADING NETWORKS

Joachim Hunker[1], Alexander Wuttke[2], Markus Rabe[2], Hendrik van der Valk[1,3], and Mario Di Benedetto[3]

[1]Fraunhofer Institute for Software and Systems Engineering, Dortmund, GERMANY
[2]Dept. of IT in Produktion and Logistics, TU Dortmund University, Dortmund, GERMANY
[3]Chair of Industrial Information Management, TU Dortmund University, Dortmund, GERMANY

## ABSTRACT

Volatile customer demand poses a significant challenge for the logistics networks of trading companies. To mitigate the uncertainty in future customer demand, many products are produced to stock with the goal to be able to meet the customers' expectations. To adequately manage their product inventory, demand forecasting is a major concern in the companies' sales planning. A promising approach besides using observational data as an input for the forecasting methods is simulation-based data generation, called data farming. In this paper, purposeful data generation and large-scale experiments are applied to generate input data for predicting customer demand in sales planning of a trading company. An approach is presented for using data farming in combination with established forecasting methods such as random forests. The application is discussed on a real-world use case, highlighting benefits of the chosen approach, and providing useful and value-adding insights to motivate further research.

## 1 INTRODUCTION

A central challenge in the competitive design of a company's supply chain planning processes is forecasting future customer demand and corresponding sales, known as sales planning (Serdarasan 2013). This applies in particular to trading companies with a focus on consumer goods, which produce their often homogeneous products to stock independently of specific customer orders and, therefore, require a flexible, responsive logistics network, in particular regarding distribution (Daugherty et al. 2019). The main task is to harmonize customer demand with the necessary production and stocks of products in order to achieve a high level of service on the one hand and the lowest possible capital commitment costs on the other hand (Pfohl 2022). Even today, managers in trading companies often predict sales using their own intuition, experience, and spreadsheets (Mitra et al. 2022). However, a manual sales planning and demand forecasting is often not feasible due to the complexity of the task described above. To tackle this issue, companies are increasingly using forecasting methods to predict future customer demand and mitigate variance in company sales (Saldaña-Olivas and Huamán-Tuesta 2021). It is, therefore, a necessity that the results from these forecasting methods are as sound and accurate as possible (Box et al. 2015).

With increasing computational power, numerous quantitative methods for demand forecasting have been presented in the scientific literature. An established approximation is the distinction between quantitative and qualitative sales forecasting methods, further differentiating into judgmental methods, experimental methods, causal methods, common projective methods, and advanced projective methods (Rushton et al. 2014). However, a strict separation between methods is lacking in the established publications. Typical examples of methods, which are widely used in research and practice, include time series models, such as Auto-Regressive Integrated Moving Average (ARIMA), or machine learning methods such as a random forest (RF) and artificial neural networks (Mitra et al. 2022).

The availability of suitable historical observational data is necessary for the use of these methods. However, the collection and use of observational data to forecast customer demand in sales planning is

considered a challenge in research and practice alike (Chase 2016). For example, data quality or the lack of corresponding data can prevent a value-adding use of forecasting methods (García et al. 2015).

This paper addresses this challenge and develops a method for combining forecasting methods with a data farming approach. Data farming describes the use of simulation models and targeted experiment design to generate comprehensive inferential data (Sanchez 2018). The generated simulation result data can then be used as input for forecasting methods. Von Rueden et al. (2020) motivate the combination of machine learning and simulation in a hybrid modeling approach in industrial settings, while Taylor et al. (2023) highlight the benefits and motivate avenues of further research in the context of digital twins.

The outline of the paper is as follows. The background on sales and distribution in supply chains, forecasting with an emphasis on methods and data, and data farming is given in Section 2. The novel approach to sales forecasting in combination with data farming is presented in Section 3. In this section, the use of simulation result data as the input for forecasting methods both from times series analysis and machine learning. In Section 4, the proposed approach is tested and validated using a real-world use case. The paper closes in Section 5 with a conclusion and an outlook on further research.

## 2 RELATED WORK

This section presents the background and the related work on core topics for this paper. In Section 2.1, an overview is given on supply chains and logistics distribution networks with an emphasize on the logistics tasks of sales planning and demand forecasting. Section 2.2 introduces fundamentals on methods for forecasting using time series analysis and machine learning. The section closes with background information on simulation and data farming in Section 2.3.

### 2.1 Logistics Networks of Trading Companies

Logistics networks of trading companies, called trading networks, are complex socio-technical systems involving a multitude of different actors. The main goal of such a system is to fulfill customer orders by efficiently managing the logistics processes involving procurement, movement, and storage of goods, called stock keeping units (SKU), as cost-effective as possible (Christopher 2016). Figure 1 illustrates a typical structure of a logistics network.

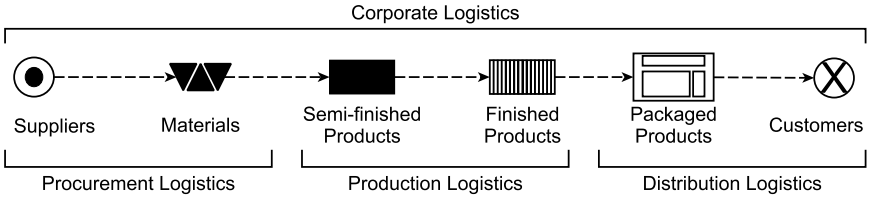


Figure 1: Common structure of a logistics network.

A logistics network consists of logistics areas, which can roughly be distinguished into the areas of procurement logistics, production logistics, and distribution logistics (Pfohl 2022), whereas procurement and distribution are core areas of a trading company (leaving aside possible value-added services).

Propagating upstream across the logistics network, customer demand is what drives a logistics network (Schulte 2016). Following Meffert et al. (2015), demand is defined as the quantity for a specific SKU requested by an economic entity. Demand for a SKU can be dependent on a multitude of different factors. Typical examples include seasonality and weather, marketing activities, or the results of independent product tests and reviews. The demand is met by market sales, defined as the SKUs sold by a company to a customer (Pfohl 2022). To tackle the problem of fluctuating customer demand, companies use an inventory management by producing products to stock and try to obtain more information about the customers' future demand (Feizabadi 2022). Stocks are used as a buffer to harmonize and balance the flow of goods and the

mismatch between supply and customer demand (Pfohl 2022). Stocks have a lot of positive characteristics, but are, however, not always desirable. Inventories can be used to prepare for uncertainty of events, as demand can not be predicted with full certainty. Yet, stocks lead to higher inventory costs for trading companies. The focus is on making to stock with capacity constraints, which implies a need for accurate sales planning performance. One of the relevant planning challenges is to gain an understanding of the future demand and to balance production and sales of inventory. Important performance metrics influenced by demand prediction are capacity utilization and return on investment (Pfohl 2022). To tackle this challenge, accurate forecasts are needed. Predicting the future demand of SKUs and planning possible sales is a complex challenge for decision makers in trading companies. Typical planning tasks include operations planning, inventory and supply planning, and sales planning (Feigin 2011). The goal of sales planning is to develop an unbiased, valid, and sound demand forecast. Since creating forecasts manually is unfeasible, various methods in research and practice have been proposed.

### 2.2 Demand Forecasting in Sales Planning

Due to the rise in available computational power, advancements in the use of observational data from the logistics network of a trading company for forecasting have been achieved in the last decade. Demand forecasting in sales planning has been extensively studied in scientific publications. Mitra et al. (2022) present a comprehensive literature review of demand forecasting models used in research and practice.

The goal in demand forecasting is to predict future outcomes based on observational data from history, called time series data (Hyndman and Athanasopoulos 2021). Time series data are "data that have been observed at different points in time" (Shumway and Stoffer 2017, p. 1), whereby a time series is defined as sequentially recorded data over time, with $x_n$ describing a random variable $x$ at discrete points in time $n$, $n \in \mathbb{N}$. Time series data might be univariate (e.g., a sensor measuring temperature) or multivariate (e.g., a sensor measuring speed, acceleration, and centrifugal force). To analyze such data, time series analysis can be used to find a mathematical model describing the time series data appropriately. Typically, forecasting methods for sales are differentiated into two groups: qualitative and quantitative methods. Following Rushton et al. (2014), qualitative methods can be distinguished into judgmental and experimental, while quantitative methods are further distinguished into causal, common projective, and advanced projective (e.g., ARIMA, machine learning). Common projective approaches to forecasting, called naïve approaches, are often used as a base reference for comparison of results. A typical example is exponential smoothing, which does not consider any trends or seasonality in the data, and is the basis for many advanced forecasting methods today (Hyndman and Athanasopoulos 2021). The idea behind simple exponential smoothing is to use the last value in the time series data, $x_n$, to forecast future values. Hence, the last value is the only important one providing information for the future.

With a focus on advanced projective methods, ARIMA is a well established method and is based on the Box–Jenkins method (Box et al. 2015). The idea is that a specific value in the time series data, $x_n$, can be described as a function of historical values (Shumway and Stoffer 2017). Following Buettner and Rabe (2021), ARIMA($p$,$d$,$q$) with $p,d,q \in \mathbb{N}$ can be described as a generic form of a time series model using an autoregressive part, number of observations $p$, to describe the regression of the time series values, an integrated part, $d$, to create a time series where the properties (e.g., the mean) of the series does not depend on the time (stationary), and a moving average part, $q$, to describe the relation between an observed value and the error of an observation.

Lately, a vast number of machine learning methods have been proposed to address time series sales forecasting. A typical machine learning technique in science and practice for demand forecasting is to use an ensemble algorithm called RF (Vairagade et al. 2019). An RF is an ensemble algorithm that inherently performs variable selection and is capable of capturing complex, non-linear interactions between features without the need for manual specification. This makes a RF suitable for cases where the relevance and structure of inputs are not fully known a priori. The basic idea behind a RF is to create large ensembles of decision trees in a first step and to merge them in a second step to create a prediction by taking the

mean from all tree predictions. In that way, a RF is a classifier of a collection of different tree-structured classifiers (Breiman 2001). An RF comes with hyperparameters that need to be tuned before training, testing, and application, such as the depth of a tree in the forest and the minimum number of splits to prevent overfitting. Many alternatives to using an RF for sales forecasting exist, with one of the most prominent ones being XGBoost. XGBoost is also a tree-based ensemble method, but the algorithm differs fundamentally in how the ensembles are build (parallel trees, called bagging versus sequential trees, called boosting). For a comparative study of different algorithms in this context, the reader is kindly referred to Mitra et al. (2022).

Most of the statistical and machine learning methods strongly depend on observational data from the logistics network of the trading company. However, flaws of observational data include, e.g., missing values or outliers, which lower the overall data quality (García et al. 2015). Another drawback is that observational data limit the types of insights that can be gained from applying forecasting algorithms on the (preprocessed) data basis (Sanchez et al. 2020). In this context, simulation-based data generation, called data farming, can be used efficiently explore vast input spaces, reveal key characteristics of complex simulation outcomes, and clearly identify causal relationships.

### 2.3 Data Farming

Simulation is an established method for the modeling and analysis of complex systems such as supply chains, where analytical methods are not applicable (Rabe et al. 2008). It is defined as the "representation of a system with its dynamic processes in an experimentable model to reach findings, which are transferable to reality; in particular, the processes are developed over time" (VDI-Guideline 3633 Part 1 2014, p. 3). Data farming can be used to address the flaws of observational data. The term data farming was coined by Brandstein and Horne (1998) in a project by the US Marine Corps. The term data farming is a metaphor for the iterative process of using a simulation model to generate vast amounts of inferential data (Sanchez 2021). To conduct a data farming study in a structured manner, procedure models are used. A procedural model structures the process of data farming into several phases, including model development, design of experiments, and a subsequent analysis of the simulation result data. Established processes are presented, for example, Feldkamp et al. (2015), Sanchez (2020), and Hunker et al. (2022).

Since complex simulation models usually contain a multitude of input variables, called factors, that supposedly have an influence on the response of the simulation model, well-designed experiments are inevitable to enable large scale experimentation (Sanchez et al. 2020). The goal of the design of experiments is to create a matrix, consisting of design points (rows) for the factors (columns). Each row represents a specific combination of factor settings, called levels (Kleijnen 2015). To create a design in a structured and value-adding way, different designs have been proposed. Commonly used, all-around suitable space-filling designs for large scale experimentation include designs based on Latin hypercubes, such as the nearly orthogonal Latin hypercubes (Cioppa and Lucas 2007). Latin hypercubes enable a rich and balanced model output, but need considerably less simulation runs in complex simulation models than, for example, a full factorial design. The generated data basis is used for a subsequent analysis to gain extensive insights on the model behavior. A typical method used for the analysis of the vast amounts of inferential data is knowledge discovery in databases. Furthermore, verification and validation (V&V) is a central part of data farming, integrating established and model-accompanying procedure models (Rabe et al. 2008).

Research in data farming has gained momentum in the last decade, and the credibility and suitability of data farming to tackle real world problems has been shown (Sanchez 2018). Originating in the domain of defense and manifold successful applications, e.g., in Horne and Seichter (2014), data farming has been applied in various other domains, such as manufacturing (Feldkamp et al. 2015), logistics (Hunker et al. 2021), and condition-based maintenance (Wuttke et al. 2023).

## 3 SALES PLANNING USING DATA FARMING

According to the authors' understanding, sales planning is the first step in supply chain planning tasks and lays the foundation for the operations planning and inventory planning in a trading company (see Section 2.1). Forecasting customer demand is a recurring and important challenge for managers to keep a trading company in a competitive state. Although manual forecasting by the managers is still a major factor in companies nowadays, forecasting methods are increasingly applied. One way to explore complex models' emergent behavior, where analytical methods fall short, is simulation (see Section 2.3). Therefore, a simulation-driven approach is promising to reap the benefits of data farming in sales planning.

This section presents a methodology to use data farming to generate input for forecasting methods in sales planning of trading companies. The procedure operates on the premise that if the simulation result data generated by the simulation resemble the necessary input needed for a specific forecasting algorithm, then the factors of the simulation model allow to identify cause-effect relationships accurately, providing broad insights.

For data-farming-based demand forecasting in general, it is crucial to have a high-quality simulation model of the companies trading network being analyzed. In this context, "high quality" means a model that adequately reflects the systems behavior, dynamics, and interdependencies under various system conditions. This includes capturing both normal system operation and deviations that could indicate a change in customer demand of a SKU. Here, expert knowledge from the trading network or observational data can yield insights on how to determine necessary assumptions for the simulation model. The model must be sufficiently detailed to include key components, interactions, and environmental factors that influence the customer demand. Additionally, it should be flexible and updatable to reflect changes in the systems configuration or operating conditions over time. It must also be robust enough to handle uncertainties and variations present in real-world operations.

This includes in particular the modeling of customer demand. When incorporating demand as an input for a simulation model of a trading network for data farming, it should vary within a specific range of values to facilitate the exploration of the models' behavior through targeted design of experiments. As detailed in Section 2.3, design of experiments guides the exploration of the simulation model by adjusting its input factors. For instance, when considering factors that influence an SKUs demand, one factor might serve as a multiplier to scale demand, while another could provide a time-based offset. However, the authors argue that relying solely on these simple operations may not adequately maintain the original demand behavior. Demand for a SKU in trading networks can be represented as a sequence of values over time, making it insufficient to describe them with a single factor. Thus, the factors generated by the design of experiments need to be converted into sequences of values that mimic demand. To achieve this transformation, the authors have developed a demand generator that is user-friendly and requires a minimal effort to set up. A fundamental discussion of this concept can be found in Wuttke et al. (2022). The objective is to create synthetic demand that closely maintains the original demand behavior for each SKU while allowing for effective exploration of the solution space. The demand generator preserves both the individual demand patterns and the overall meta behavior of a specific SKU. The factors resembling demand in the experiment matrix are varied by the chosen design of experiments in given ranges. Based on these inputs, the demand generator generates a synthetic demand for a given SKU for a simulation run. An example is illustrated in Figure 2.

Figure 2 shows the observational demand from the time series data of a given SKU over the course of a year in a trading company, which is resembled by the fundamental curve. A fitted curve shows the result of a parametrized distribution. Both extremal curves resemble minimal and maximal levels for the parameters of the function, giving a realistic range to vary the demand in. This is the basis to generate demand for the SKU.

The output of the simulation model is used as an input for the forecasting algorithms. The simulation result data resembles time series data as defined in Section 2.1 to fit the input requirements of a forecasting algorithm. Depending on the forecasting method used (see Section 2.2), training of the model using

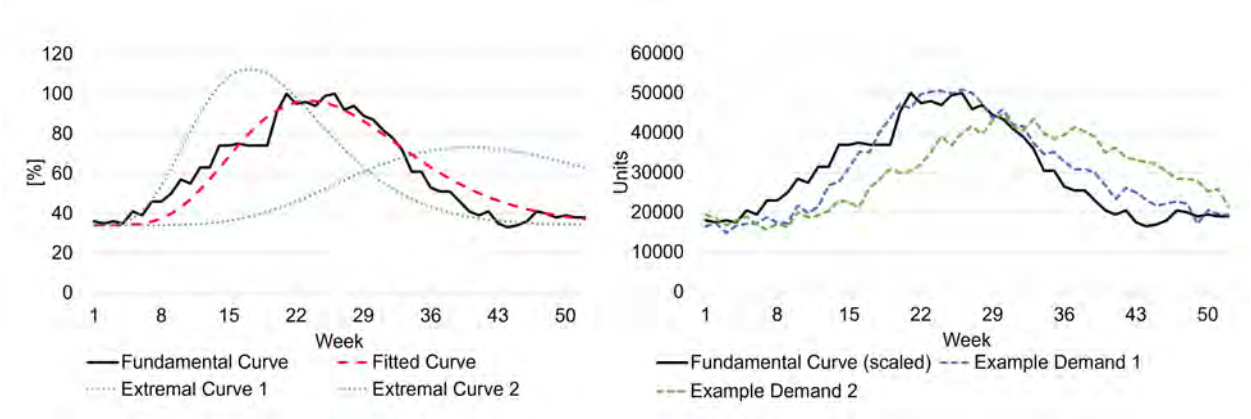


Figure 2: Exemplary results from the demand generator for a specific stock keeping unit.

training data is necessary. The current issue falls under the category of supervised learning methods, given that the data available for learning is labeled. More specifically, the objective is to develop a model for time series data. It is crucial to consider the data as sequences instead of treating them independently, as these sequences display temporal dependencies that offer valuable insights for demand forecasting. This method enables the model to utilize past observations and make educated predictions about future values by leveraging the historical context inherent in the time series.

The presented approach is based in particular on established procedure models that describe an explicit combination of data farming and, for example, knowledge discovery in databases (see Section 2.3). Three basic application scenarios of our approach are hypothesis testing, data enrichment by inferential data, and generating a complete inferential data basis for demand forecasting in sales planning. Our approach is structured into several consecutive steps. Overall, recommended is to use an established procedure model for conducting a forecasting study enhanced by data farming. The core phases of our approach are illustrated in Figure 3.

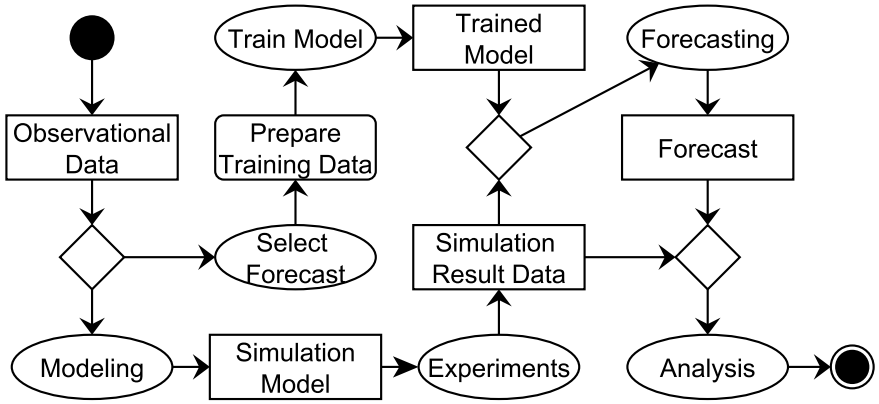


Figure 3: Procedure model to combine data farming and demand forecasting.

The method starts with observational data from a trading company. The previous steps for obtaining data from a trading network are deliberately excluded at this point. *Step 1* is concerned with building a simulation model and design of experiments. The simulation model should be able to output time series data for given points in time (univariate or multivariate, see Section 2.2). To vary the factors, using a suitable design of experiments is recommended. From the authors' experience, nearly orthogonal and balanced Latin hypercubes serve as a valid all-purpose design, as recommended by the established literature (see

Section 2.3). Step 1 results in an experimentable model. In *Step 2*, the experiments are run, resulting in simulation result data, which usually get stored in a database. In *Step 3*, it is necessary to select a forecasting algorithm for a given method. This is important, as forecasting methods may have different requirements for their set-up or the input data. For example, recurrent neural networks have requirements to select, e.g., the type and the number of layers, respectively. For *Step 4*, it is essential to prepare a training dataset for a given machine learning algorithm. This dataset should contain a substantial amount of observational data collected from the trading network, encompassing various operational scenarios and conditions. Such a dataset enables the machine learning models to generalize effectively and learn patterns. *Step 5* includes the training of the model selected in Step 3 using the time series data prepared for training in Step 4. In *Step 6*, the trained model is applied to the result of the data farming phase to generate forecasts. The final *Step 7* is concerned with a joint analysis of the forecasts and the simulation result data. This is accompanied by exploration of the input space, response surface techniques, and cause-effect-analysis.

A rigorous V&V of these steps is recommended. Typically, established procedure models contain a dedicated phase or multiple phases for conducting a structured V&V and recommend various techniques (see Section 2.3). However, the proposed procedure in Figure 3 does not depict specific steps for V&V and a reiteration of steps, e.g., if the V&V of the simulation model fail or if the training performance of the machine learning is unsatisfactory. The following section illustrates the newly introduced approach through an industrial use case.

## 4 USE CASE

The concept presented in Section 3 is demonstrated using a real-world use case and the gained results are then critically discussed. The observational data originates from a leading trading company in Germany that is active worldwide and sells a large number of different SKUs. The considered part of the trading network consists of three suppliers, five sites, and more than a hundred customers in the region Germany, Austria, and the Netherlands. Observational data were collected from the trading companies' enterprise resource planning system, such as site location, stocks of SKUs, customer orders, and deliveries to customers. In the use case considered here, the focus is on one of the company's main products, which is distributed to multiple customers in Europe (see Section 2.1). The time horizon for the data extraction was one year. Some hypotheses have been defined as a starting point. For example, an increase in sales activities in autumn should lead to an increase of at least 5% in the predicted customer demand. If that is the case, what impact will this have on the trading network and how much spread is expected in future demand?

The simulation model was developed using AnyLogic, which also supported the data farming experiments. Python, with Pandas and Scikit-learn, was used for analyzing simulation responses and applying demand forecasting. The experimental design employed nearly orthogonal and balanced Latin hypercubes, based on spreadsheets from the SEED Center for Data Farming (Sanchez 2011). An initial exploratory analysis was followed by preprocessing of trading company data, including harmonizing and aggregating customer order documents. Figure 4 shows sales data for a selected SKU, which exhibits a seasonal demand pattern typical for construction materials: higher demand during the warmer months from spring to autumn and a lower demand during the cold months in winter.

The next step is to select a forecasting method. In this use case, established methods such as ARIMA and RF have been applied (see Section 2.2). The observational data are split into training data and test data in the ratio of four to one while the forecasting error is averaged, which is a common approach in model training and validation. The RF has then been trained and validated based on the observational data. It is assumed that product-level sales do not conform to a neat parametric distribution and seasonal demand (as illustrated in the example from Figure 2), often exhibiting random-seeming, erratic movements in practice. Consequently, a bootstrapping approach is integrated to generate daily demand from historical data: each day's aggregated sales quantity is sampled (with replacement) from a set of real historical values. This preserves the empirical range and irregular fluctuations of the original data without imposing a traditional theoretical distribution. Next, we distribute the total bootstrapped distributed for each day

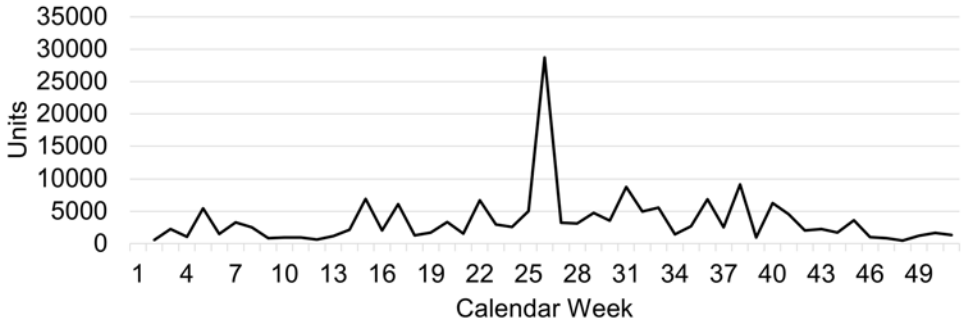


Figure 4: Sales data for a stock keeping unit.

among five hubs to ensure an accurate representation of the throughput variability observed in real-world scenarios. The allocation to each hub is based on predetermined percentage weights (for instance, 30%, 25%, and so forth). The model employs two distinct design strategies: the first utilizes fixed weights, realizing that two sites encounter higher traffic levels compared to the others. The second strategy adopts the largest remainder method for further refinement. In this approach, the integer portions are allocated initially, while the remaining units are assigned to the hubs with the largest fractional components. This method guarantees that the overall allocation across the hubs aligns with the daily aggregated total. Figure 5 illustrates exemplary demands for the selected SKU in this use case.

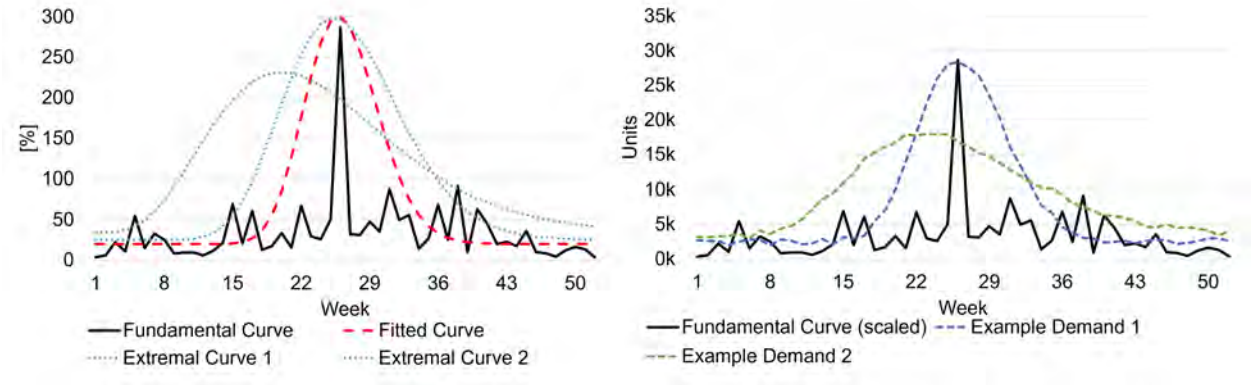


Figure 5: Examples of generated demand for a specific stock keeping unit.

Finally, this entire process is implemented in AnyLogic, with daily events tracking the bootstrapped demand and hub assignments. The resulting synthetic time series with integer hub distributions are logged to a CSV file format, forming a robust data basis for subsequent demand forecasting tests and model training. By refraining from a single parametric model, random-like variability realistically reflects the unpredictability inherent in real-world demand for SKUs. To investigate how hub-weight assumptions affect the final daily allocations, two distinct designs have been tested. Design A employs fixed proportions of 30%, 25%, 15%, 15%, and 15%, offering a balanced split with two moderately larger hubs and three smaller ones. Design B, on the other hand, reflects real measured utilization, with weightings of approximately 45.2%, 26.2%, 11.2%, 10.7%, and 6.7%. In each scenario, the simulation has been run for 365 days using the same bootstrapped daily demand, logging an integer-based breakdown per hub. Comparing the two output files allows for isolating the impact of more idealized compared to empirically derived hub distributions on overall demand patterns. The data farming model is used to generate time series data that can be used for an RF model. This leads to the creation of a comprehensive synthetical data basis that

allows different data sets to be compared with each other. In particular, a distinction is made between the observed data, the enriched database, and the complete synthetical data basis. The simulation model illustrated in Figure 6 consists of four core elements working in tandem to generate a realistic daily demand distribution. First, historicalSales serves as the reservoir of original data, holding an ArrayList of empirical sales figures that are sampled (with replacement) each day in a nonparametric bootstrapping fashion. This ensures that the synthetic time series reflects the actual variability observed in the real-world dataset. Second, the global variables, such as the day counter and the CSV writer, offer persistent references and logging capabilities throughout each day's process. Third, the dailyEvent triggers once per simulated day, getting a new demand value from the historical data and allocating the total across five hubs. During this allocation, either a straightforward percentage-based method or the Largest Remainder technique is applied to ensure that each hub's share remains an integer while preserving the exact sum of the total daily demand. Finally, HubSales (hubSales1 to hubSales5) captures the final allocation for each day, reflecting throughput differences between the hubs.

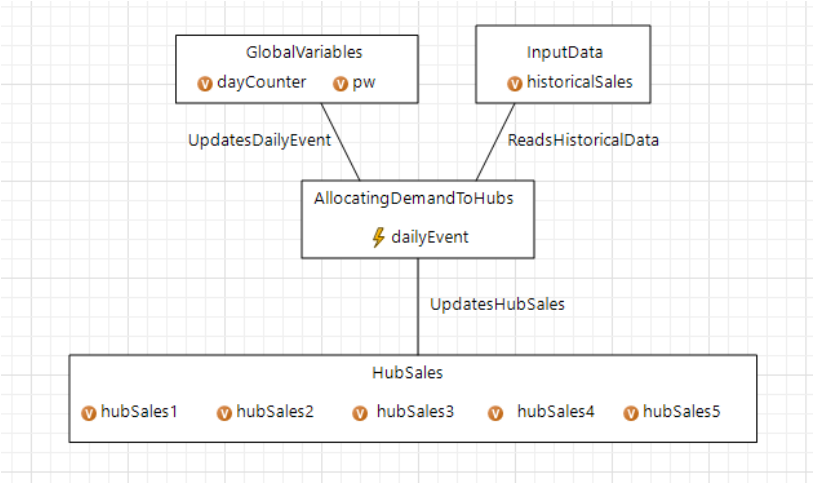


Figure 6: Core elements of the simulation model.

Extensive preprocessing is required when processing the observational data. This includes steps such as checking for missing values, renaming columns, and removing irrelevant columns. These measures are crucial to ensure the quality of the data and to ensure that the subsequent analyses are based on reliable information. In contrast, the evaluation based on the simulation results does not require preprocessing, since the response of the simulation model was configured to directly generate time series data. This increases the efficiency of data processing and minimizes the complexity of data preparation. Ultimately, the forecasting algorithms are applied to the simulation results. These algorithms take into account the specific characteristics of the time series data to produce accurate and meaningful forecasts. Figure 7 shows exemplary results of an RF for a specific replication and design point, compared to the real data for the predicted year. To check if the time series is stationary, an Augmented Dickey-Fuller test has been implemented. For hyperparameter tuning, the function GridSearchCV has been applied and the identified best parameters are then used to train the model based on the mean square error to approximate the model quality. The integration of these procedures into the data farming model ensures that the results are robust and provide significant added value for the analysis of sales planning in trading companies.

In this use case, the combination of data farming and forecasting methods has proven to be value-adding in the sales planning process of a trading company. For forecasting, we relied on commonly used techniques for forecasting, such as RF. More-complex architectures such as recurrent neural networks, large language models, agents, and retrieval-augmented generation have not been touched upon in this paper. The machine learning model has been trained and validated using observational data the materials trading company and

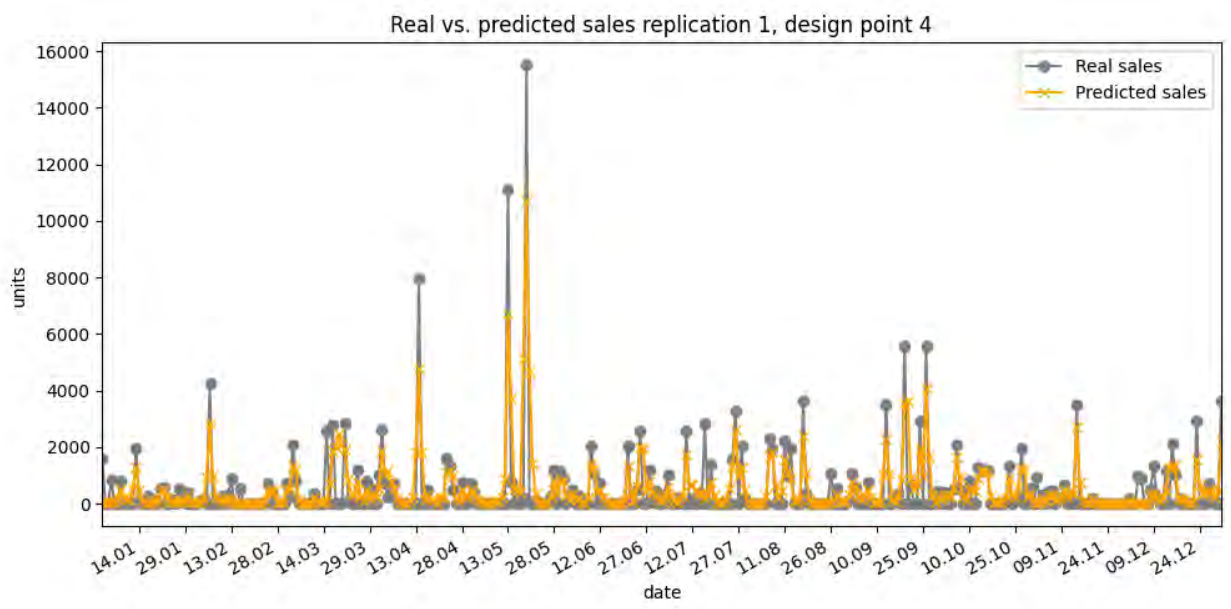


Figure 7: Predicted sales using a random forest based on simulation result data.

a situation, were observational data might not be available, has not been explored. However, the flexibility of this approach would technically allow to use simulation result data for training and validating a machine learning algorithm by adapting the method. The overall performance of our approach in comparison to a forecasting model trained on the observational data has not been investigated. An initial analysis based on the mean squared error showed a significant improvement in the forecast using our approach. Nevertheless, a sound investigation requires a more comprehensive research setting. Concluding, the use case showed that this combined method demonstrates a more broader and flexible approach to sales forecasting, leveraging advantages of both data farming and forecasting.

## 5 CONCLUSION AND OUTLOOK

In this paper, a novel approach has been showcased on how data farming can be used to add value to the sales planning process of a trading company. After introducing the related work and the theoretical background, the concept for combining those two approaches was introduced. The fundamental idea behind this approach is to use simulation based data generation to enhance the analysis of the models results. The presented concept is validated using a real-world use case from a trading company in Germany. Based on the company's observational data, the use of typical and well established approaches has been demonstrated by using ARIMA and an RF for demand forecasting. The results have been proven value-adding in the sales planning process and show that by combining data farming and demand forecasting a much better basis for decision-making can be provided.

Besides tuning and refining the method for more precise results, a promising research stream is called Robust Forecasting. This describes the idea to add this approach into a logistics assistance system and use demand forecasting for decision support. This could be done using an optimization setting to optimize from different data farming scenarios and find a possible decision for a predicted setting of the trading network which offers a competitive performance for a vast majority of possible scenarios. Here, concepts from retrieval augmented generation and agents could be a promising addition to the concept, in particular with regard to interaction with decision makers from a trading company. Furthermore, automatically integrating market trends and search patterns from customers (e.g., from Google Trends and market research) or weather forecasts could further enrich the demand forecasting and seem promising for future research.

## REFERENCES


Box, G. E. P., G. M. Jenkins, G. C. Reinsel, and G. M. Ljung. 2015. *Time Series Analysis: Forecasting and Control*. 5th ed. ed. Wiley Series in Probability and Statistics. Hoboken: Wiley.

Brandstein, A. G., and G. E. Horne. 1998. *Data Farming: A Meta-technique for Research in the 21st Century*. Quantico, Virginia: Marine Corps Combat Development Command Publication.

Breiman, L. 2001. "Random Forests". *Machine Learning* 45(1):5–32.

Buettner, D., and M. Rabe. 2021. "Sales Forecasting in the Electrical Industry - An Illustrative Comparison of Time Series and Machine Learning Approaches". In *Proceedings of the 9th International Conference on Traffic and Logistic Engineering (ICTLE)*, 69–78. Piscataway, NJ, USA: Institute of Electrical and Electronics Engineers, Inc.

Chase, C. W. 2016. *Next Generation Demand Management*. Hoboken, NJ, USA: Wiley.

Christopher, M. 2016. *Logistics & Supply Chain Management*. 5th ed. Harlow, England: Pearson Education.

Cioppa, T. M., and T. W. Lucas. 2007. "Efficient Nearly Orthogonal and Space-Filling Latin Hypercubes". *Technometrics* 49(1):45–55.

Daugherty, P. J., Y. Bolumole, and S. J. Grawe. 2019. "The New Age of Customer Impatience". *International Journal of Physical Distribution & Logistics Management* 49(1):4–32.

Feigin, G. 2011. *Supply Chain Planning and Analytics: The Right Product in the Right Place at the Right Time*. New York, NY: BusinessExpert Press.

Feizabadi, J. 2022. "Machine Learning Demand Forecasting and Supply Chain Performance". *International Journal of Logistics Research and Applications* 25(2):119–142.

Feldkamp, N., S. Bergmann, and S. Straßburger. 2015. "Visual Analytics of Manufacturing Simulation Data". In *2015 Winter Simulation Conference (WSC)*, 779–790 https://doi.org/10.1109/WSC.2015.7408215.

García, S., J. Luengo, and F. Herrera. 2015. *Data Preprocessing in Data Mining*. Cham: Springer International Publishing https://doi.org/10.1007/978-3-319-10247-4.

Horne, G., and S. Seichter. 2014. "Data Farming in Support of NATO Operations – Methodology and Proof-of-Concept". In *2014 Winter Simulation Conference (WSC)*, 2355–2363 https://doi.org/10.1109/WSC.2014.7020079.

Hunker, J., A. A. Scheidler, M. Rabe, and H. van der Valk. 2022. "A New Data Farming Procedure Model for a Farming for Mining Method in Logistics Networks". In *2022 Winter Simulation Conference (WSC)*, 1461–1472 https://doi.org/10.1109/WSC57314.2022.10015249.

Hunker, J., A. Wuttke, A. A. Scheidler, and M. Rabe. 2021. "A Farming-for-Mining-Framework to Gain Knowledge in Supply Chains". In *2021 Winter Simulation Conference (WSC)* https://doi.org/10.1109/WSC52266.2021.9715372.

Hyndman, R. J., and G. Athanasopoulos. 2021. *Forecasting: Principles and Practice*. 3rd ed. Melbourne, Australia: Otexts Online Open-Access Textbooks.

Kleijnen, J. P. 2015. *Design and Analysis of Simulation Experiments*. Cham: Springer International Publishing.

Meffert, H., C. Burmann, and M. Kirchgeorg. 2015. *Marketing*. Wiesbaden: Springer Fachmedien.

Mitra, A., A. Jain, A. Kishore, and P. Kumar. 2022. "A Comparative Study of Demand Forecasting Models for a Multi-Channel Retail Company: A Novel Hybrid Machine Learning Approach". *Operations Research Forum* 3(4).

Pfohl, H.-C. 2022. *Logistics Systems: Business Fundamentals*. Wiesbaden, Germany: Springer Gabler.

Rabe, M., S. Spieckermann, and S. Wenzel. 2008. "A New Procedure Model for Verification and Validation in Production and Logistics Simulation". In *2008 Winter Simulation Conference (WSC)*, 1717–1726 https://doi.org/10.1109/WSC.2008.4736258.

Rushton, A., P. Croucher, and P. Baker. 2014. *The Handbook of Logistics and Distribution Management: Understanding the Supply Chain*. 5th ed. London, Great Britain: Kogan Page.

Saldaña-Olivas, E., and J. R. Huamán-Tuesta. 2021. "Extreme Learning Machine for Business Sales Forecasts: A Systematic Review". In *Proceedings of the 5th Brazilian Technology Symposium*, edited by Y. Iano, R. Arthur, O. Saotome, G. Kemper, and R. Padilha França, Volume 201, 87–96. Cham: Springer International Publishing.

Sanchez, Susan M. 2011. "NOLHdesigns spreadsheet". http://harvest.nps.edu/, accessed: 29.03.2024.

Sanchez, S. M. 2018. "Data Farming: Better Data, Not Just Big Data". In *2018 Winter Simulation Conference (WSC)*, 425–439 https://doi.org/10.1109/WSC.2018.8632383.

Sanchez, S. M. 2020. "Data Farming: Methodes for the Present, Opportunities for the Future". *ACM Transactions on Modeling and Computer Simulation* 30(4):1–30.

Sanchez, S. M. 2021. "Data Farming: The Meanings and Methods Behind the Metaphor". In *Proceedings of the Operational Research Society Simulation Workshop 2021*, edited by M. Fakhimi, T. Boness, and D. Robertson, 10–17: Operational Research Society.

Sanchez, S. M., P. J. Sanchez, and H. Wan. 2020. "Work Smarter, not Harder: A Tutorial on Designing and Conducting Simulation Experiments". In *2020 Winter Simulation Conference (WSC)*, 1128–1142 https://doi.org/10.1109/WSC48552.2020.9384057.

Schulte, C. 2016. *Logistik: Wege zur Optimierung der Supply Chain*. 7th ed. Munich: Franz Vahlen.

Serdarasan, S. 2013. "A Review of Supply Chain Complexity Drivers". *Computers & Industrial Engineering* 66(3):533–540.

Shumway, R. H., and D. S. Stoffer. 2017. *Time Series Analysis and its Applications*. Cham: Springer International Publishing.

Taylor, S. J. E., C. M. Macal, A. Matta, M. Rabe, S. M. Sanchez, and G. Shao. 2023. "Enhancing Digital Twins with Advances in Simulation and Artificial Intelligence: Opportunities and Challenges". In *2023 Winter Simulation Conference (WSC)*, 3296–3310 https://doi.org/10.1109/WSC60868.2023.10408011.

Vairagade, N., D. Logofatu, F. Leon, and F. Muharemi. 2019. "Demand Forecasting Using Random Forest and Artificial Neural Network for Supply Chain Management". In *Computational Collective Intelligence*, edited by N. T. Nguyen, R. Chbeir, E. Exposito, P. Aniorté, and B. Trawiński, 328–339. Cham: Springer International Publishing.

VDI-Guideline 3633 Part 1 2014. *VDI 3633 – Simulation of Systems in Materials Handling, Logistics and Production: Fundamentals*. Berlin, Germany: Beuth.

von Rueden, L., S. Mayer, R. Sifa, C. Bauckhage, and J. Garcke. 2020. "Combining Machine Learning and Simulation to a Hybrid Modelling Approach: Current and Future Directions". In *Advances in Intelligent Data Analysis*, edited by M. R. Berthold, A. Feelders, and G. Krempl, Lecture Notes in Computer Science, 548–560. Cham: Springer International Publishing.

Wuttke, A., J. Hunker, M. Rabe, and J.-P. Diepenbrock. 2023. "Estimating Parameters with Data Farming for Condition-Based Maintenance in a Digital Twin". In *2023 Winter Simulation Conference (WSC)*, 1641–1652 https://doi.org/10.1109/WSC60868.2023.10408594.

Wuttke, A., J. Hunker, A. A. Scheidler, and M. Rabe. 2022. "Synthetic Demand Generation with Seasonality for Data Mining on a Data-Farmed Data Basis of a Two-Echelon Supply Chain". *Procedia Computer Science* 204:226–234 https://doi.org/10.1016/j.procs.2022.08.027.

## AUTHOR BIOGRAPHIES

**JOACHIM HUNKER** is deputy head of the department industrial manufacturing at the Fraunhofer Institute for Software and Systems Engineering and a PhD candidate at the department of IT in Production and Logistics at TU Dortmund University. He holds a Master of Science in Logistics, Infrastructure, and Mobility with a focus on IT in Logistics from the Technical University of Hamburg. He graduated with a master thesis on a hybrid scheduling approach of assembly lines of car manufacturers. His research focuses on simulation-based data generation and data analytics in logistics. His email address is joachim.hunker@tu-dortmund.de. ORCID: 0000-0002-2715-6430

**ALEXANDER WUTTKE** is a PhD Candidate and researcher at the department IT in Production and Logistics at TU Dortmund University. He holds a M.Sc. in Mechanical Engineering from TU Dortmund University. His research interests are digital twins, condition-based maintenance, and simulation. His email address is alexander2.wuttke@tu-dortmund.de. ORCID: 0000-0002-4435-2046

**MARKUS RABE** is a full professor for IT in Production and Logistics at the TU Dortmund University. Until 2010 he had been with Fraunhofer IPK in Berlin as head of the corporate logistics and processes department, head of the central IT department, and a member of the institute direction circle. His research focus is on information systems for supply chains, production planning, and simulation. Markus Rabe is vice chair of the "Simulation in Production and Logistics" group of the simulation society ASIM, member of the editorial board of the Journal of Simulation, member of several conference program committees, has chaired the ASIM SPL conference in 1998, 2000, 2004, 2008, and 2015, Local Chair of the WSC'2012 in Berlin and Proceedings Chair of the WSC'2018 and WSC'2019. More than 250 publications and editions report from his work. His email address is markus.rabe@tu-dortmund.de. ORCID: 0000-0002-7190-9321

**HENDRIK VAN DER VALK** is the chief engineer at the Chair for Industrial Information Management at TU Dortmund University, supervising the departments of Data-Driven Value Chains and Data Ecosystems, and is scientific coordinator at the Fraunhofer Institute for Software and Systems Engineering. His research focuses on the data-driven circular economy, digital twins, and data ecosystems. Also, his research interests lie in the field of reference architectures and procedure models. He holds a PhD in Mechanical Engineering from TU Dortmund University. He is currently a member of the Digital Twin Hub, the Institute for Operations Research and the Management Sciences INFORMS, the German Chapter of the Association for Information Systems, the German Association of Information Systems, and the Circular Economy Digital Hub Initiative. His e-mail address is hendrik.van-der-valk@tu-dortmund.de. ORCID: 0000-0001-6329-792X

**MARIO DI BENEDETTO** is a research assistant at the Chair for Industrial Information Management at TU Dortmund University. He holds a Bachelor's degree in Economics from TU Dortmund University. His research interests focus on digital platforms. His email address is mario.dibenedetto@tu-dortmund.de.