

Shelf Space Optimization in Retail: A Study Using Linear Programming, Genetic Algorithm, and Proximal Policy Optimization

Anitha Palakshappa*

Department of ISE, Ramaiah Institute of Technology, Bengaluru-560054, Karnataka, Affiliated to VTU, Belagavi-590018, Karnataka, INDIA

E-mail: anitha.palakshappa@gmail.com

ORCID iD: <https://orcid.org/0000-0002-9904-8762>

*Corresponding Author

Shruti J. R.

Department of ISE, Ramaiah Institute of Technology, Bengaluru-560054, Karnataka, Affiliated to VTU, Belagavi-590018, Karnataka, INDIA

E-mail: jrshruti@msrit.edu

ORCID iD: <https://orcid.org/0009-0006-9700-4949>

Sowmya Kyatanahalli Nanjappa

Department of ISE, JSS Academy of Technical Education, Bengaluru-560060, Karnataka, Affiliated to VTU, Belagavi-590018, Karnataka, India

E-mail: kn_sowmya@rediffmail.com

ORCID iD: <https://orcid.org/0000-0001-7212-8677>

Ashwitha Anni

School of Computer Engineering, Manipal Institute of Technology Bengaluru, Manipal Academy of Higher Education, Manipal, India

E-Mail: ashwitha.a@manipal.edu

ORCID iD: <https://orcid.org/0000-0003-4767-8161>

Aditya Gaonkar

Department of ISE, Ramaiah Institute of Technology, Bengaluru-560054, Karnataka, Affiliated to VTU, Belagavi-590018, Karnataka, INDIA

E-mail: 1ms21is010@msrit.edu

ORCID iD: <https://orcid.org/0009-0006-1482-036X>

Bhawna Botra

Department of ISE, Ramaiah Institute of Technology, Bengaluru-560054, Karnataka, Affiliated to VTU, Belagavi-590018, Karnataka, INDIA

E-mail: 1ms21is029@msrit.edu

ORCID iD: <https://orcid.org/0009-0001-0889-1368>

Received: April 30, 2026; Revised: May 18, 2026; Accepted: June 9, 2026; Published: August 8, 2026

Abstract: The aim is to design a comprehensive shelf space optimization framework that maximizes profitability, enhances sales forecasting, improves efficiency of inventory management, and supports effective decision-making in retail businesses. A robust and interactive analytical dashboard is developed that allows users to visualize critical sales metrics, analyze historical data trends, and accurately forecast product demand and supply requirements based on seasonal variations and sales performance. The work integrates three mathematical optimization paradigms like Linear Programming (LP), metaheuristic search via Genetic Algorithms (GA), and reinforcement learning using Proximal Policy Optimization (PPO) to support both static and adaptive allocation strategies. Experimental validation highlights the relative advantages of each method, with detailed evaluations based on forecast accuracy, inventory turnover efficiency, shelf utilization rate, and overall improvement in profitability. Unlike traditional static optimization models,

the PPO-based framework continuously adapts allocation decisions using environmental feedback, improving flexibility in dynamic retail scenarios. The paper uses multi-objective shelf optimization considering profitability, utilization, and customer demand simultaneously. The results demonstrate that the integration of predictive analytics and advanced optimization techniques significantly outperforms traditional shelf management approaches, offering retailers actionable insights and operational advantages.

Index Terms: Shelf Space optimization, linear programming, genetic algorithm, proximal policy optimization, sales forecasting, inventory management, interactive dashboard, seasonal demand analysis

1. Introduction

Efficient management of retail shelf space plays a crucial role in maximizing profitability, enhancing inventory turnover, and improving customer satisfaction. In the dynamic and competitive landscape of modern retail, effective product placement and strategic space allocation significantly influence consumer purchasing behavior and operational efficiency. With increasing product variety, constrained shelf capacity, and fluctuating customer demand, traditional shelf management methods fall short in responsiveness and in decision making.

The current work proposes integrated shelf space optimization framework that leverages advanced data analytics and optimization techniques to address these challenges. The system combines predictive modeling, sales analytics, and optimization algorithms to offer data-driven support for strategic shelf planning. A robust and interactive dashboard is developed to visualize critical sales metrics, track product performance, and filter data by seasonal and categorical parameters. Additionally, a forecasting module, powered by Prophet, anticipates product demand to guide timely shelf adjustments and inventory decisions. A hybrid LP–GA–PPO framework is employed to enhance shelf allocation: Linear Programming (LP) for mathematically precise solutions, Genetic Algorithm (GA) for navigating complex, high-dimensional solution spaces, and Proximal Policy Optimization (PPO), a reinforcement learning approach enabling dynamic and adaptive decision-making. The system is designed to be scalable across various retail formats and customizable to different store sizes and product categories. The integration of predictive analytics, intelligent optimization, and visual decision aids addresses key industry pain points such as inefficient space utilization, inaccurate demand forecasts, and lack of actionable insights. By unifying these components, the proposed model empowers retailers with a strategic and adaptable solution for real-time, high-impact shelf space management. The proposed work presents a comparative evaluation between deterministic, evolutionary, and reinforcement learning approaches for retail shelf optimization under realistic constraints compared to the existing work.

2. Related Works

Shelf space optimization plays a critical role in retail operations by directly influencing product visibility, inventory turnover, and overall sales performance. As the retail industry embraces data-driven decision-making and AI-based technologies, researchers have explored a variety of models and algorithms to address the challenges of shelf space allocation, inventory control, and demand forecasting. This section presents a comprehensive review of relevant research contributions, categorizing them based on their alignment with our project's modules: forecasting, optimization, dashboard integration, and system scalability. Supply Chain 4.0 influences digital technologies such as AI, IoT and big data to create real-time, demand-driven operations, by ensuring the right products are placed in the right quantities at the right time [1]. The proposed work advances the integration of LP, GA and PPO. This integration reduces stockouts, minimizes excess inventory, and enhances customer satisfaction. In a systematic literature review [2], the authors highlight a wide range of approaches to the Shelf Space Allocation Problem (SSAP), pointing to the need for intelligent, adaptable methods in modern retail environments. Riazi [3] proposed an inventory control strategy for profitable perishable goods, which aligns closely with our forecasting objective using seasonal data. Author [4] introduced a bi-objective optimization model integrating assortment, promotion, shelf allocation, and inventory replenishment, providing direct inspiration for the integrated structure of our proposed framework. Similarly, Ostermeier [5] provided a full-store allocation model that serves as a structural reference for our system wide optimization logic. Gencosman and Begen [6] explored exact and decomposition-based strategies for optimizing shelf space. Their use of LP reinforces our implementation of Linear Programming to guarantee optimal outcomes under defined constraints. Czerniachowska [7] proposed a Genetic Algorithm for virtual segment shelf space allocation, which inspired our use of GA to tackle high-dimensional and dynamic layout scenarios. Murray [8] tackled the joint optimization of pricing, shelf display, and product orientation, underscoring the importance of multi-parameter tuning—an aspect we address through Proximal Policy Optimization (PPO). Sajadi and Ahmadi [9] extended this idea by integrating assortment planning, shelf space allocation, and replenishment in a real application, which validated our multi-layered optimization structure. Czerniachowska and Hernes [10] focused on heuristic support systems, incorporating visual shelf factors like capping and facings. These findings influenced our dashboard visualization logic, especially in the interactive planogram

module. The authors proposed re-allocation techniques to reduce out-of-stock scenarios, emphasizing the importance of dynamic forecasting goal served by our Prophet-based forecasting pipeline [11]. Flamand [12] showed that grouped product displays could increase impulse purchases, which we translated into our category-based allocation filters. Higuchi and Takeyasu [13] presented an expanded GA based shelf planning model, further supporting the adaptability of our GA component. Rautaray [14] suggested using big data for shelf planning, which supports our use of scalable frameworks and interactive tools like Streamlit and Plotly. Duan [15] discussed the opportunity cost of store brand placements, suggesting potential pricing integration in future versions of our system. Gajjar and Adil [16] linked shelf space allocation with replenishment strategies, which reinforces our data-driven link between forecasting and allocation. Geismar [17] introduced two-dimensional space optimization to maximize revenue, aligning with our model's ability to evaluate across multiple axes (profit, space, category). Digital supply chain practices in manufacturing boost company performance by improving demand forecasting, inventory visibility, and decision-making, which contributes to effective shelf space optimization in retail sectors [18]. Hubne [19] developed a model to maximize a retailer's profit under stochastic demand by selecting the number of facings and their shelf positions. However, several research gaps remain: most studies focus on a single optimization method, limited attention has been given to translating outcomes into practical decision support tools for retail planners, and PPO remains underexplored compared to traditional techniques. The proposed work addresses these gaps with the development of dashboard-based framework to visualize the allocation strategies and enabling quantitative performance evaluation.

3. Methodology

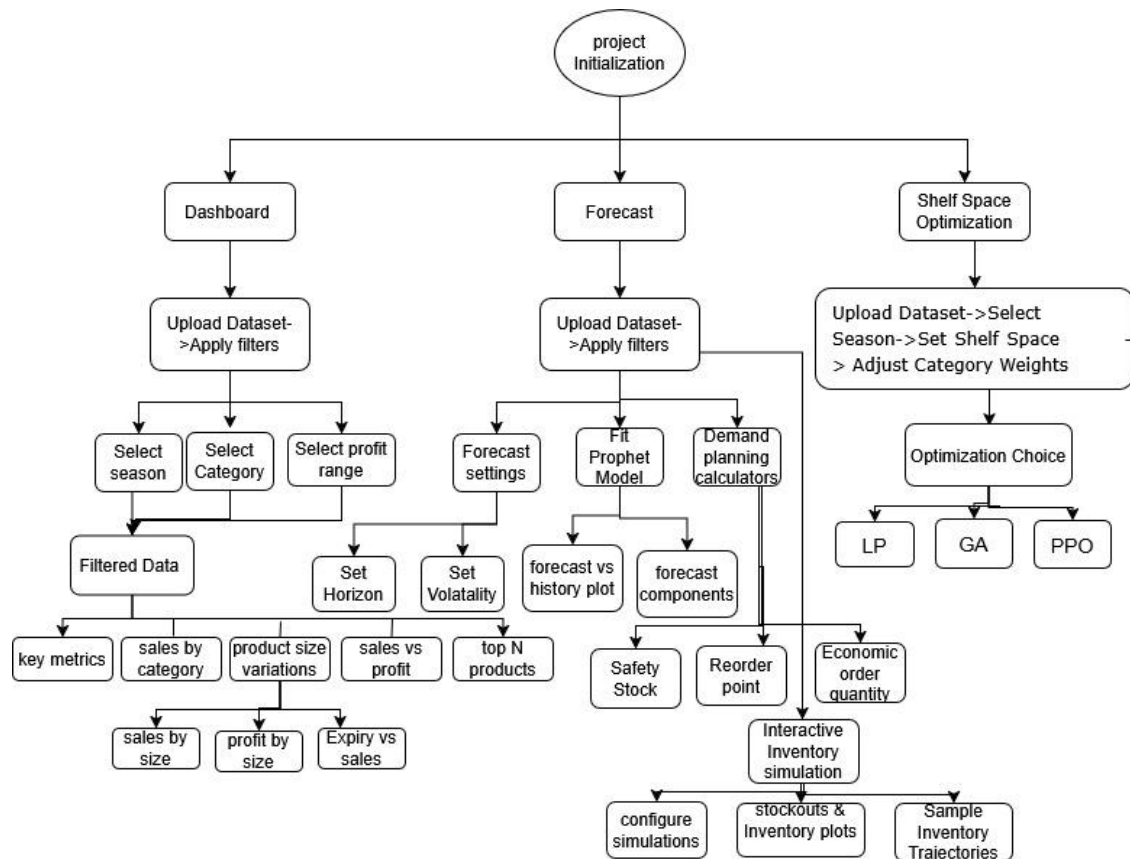


Fig. 1. Detailed Flow Chart of Data Processing and Module Execution

Figure 1 shows the flowchart of the work. The implementation flow consists of the following steps:

Step 1: Data Ingestion: The user initiates the process by uploading a CSV file containing raw sales and product metadata. The `upload_and_preview_data()` helper function reads the file, automatically detects common date formats, and coerces invalid dates to NaT (which are later dropped). Simultaneously, a new column `Shelf_Area` is computed as $\text{Width_cm} \times \text{Height_cm}$ to capture the physical space requirement for each product. If the upload fails (wrong file type or unreadable content), the flow charts back to an error handler that displays an explanatory message and prompts the user to retry.

Step 2: Preprocessing & Filtering: Once ingestion succeeds, the raw DataFrame enters a validation stage: Rows with

missing critical fields (Product_Name, Expiry_Date) are dropped. Outliers in numeric fields may be flagged for review or automatically clipped to reasonable bounds. The user is then presented with three filter controls:

- Season Selector: Picks one or more seasons (e.g., “Summer”, “All Season”).
- Category Filter: Narrows to specific product categories (e.g., “Beverages”, “Snacks”).
- Profit Range: Defines a lower and upper bound on per-unit profit.

“Apply Filters” action produces a Filtered Data subset, which drives all downstream modules in parallel.

Step 3: Module Execution (Branching Logic):

- Dashboard Module (dashboard.py): Computes key performance metrics (total sales, average profit, product turnover). Generates visualizations: sales by category, size-based profit breakdown, expiry vs. sales scatter. Renders interactive tables for Top N products and time-series drilldowns.
- Forecasting Module (forecast.py): Accepts user-specified Forecast Horizon and Volatility Multiplier. Fits a Prophet model to historical data and decomposes time series into trend, seasonality, and residuals. Runs a Monte Carlo inventory simulation for the projected horizon, outputting expected stockouts and average inventory levels. Produces two chart types: “Forecast vs. History” and “Sample Inventory Trajectories”.
- Optimization Module (optimization.py): Presents algorithm choices: Linear Programming, Genetic Algorithm, or PPO Reinforcement Learning. Translates filtered data and forecasted demand into optimization constraints (shelf area limits, category weights). Solves for an optimal planogram layout, balancing predicted sales against shelf real estate. Create a downloadable planogram mapping each product to fixed shelf coordinates.

Step 4: Output generation: As each module completes, its results are sent back to the Streamlit UI: Charts and tables appear in dedicated tabs. Simulation statistics and trade-off curves are displayed with explanatory tooltips. A unified “Download” button lets the user export: The processed CSV (with all computed fields and filtered rows). The finalized planogram in CSV or image format. If any module encounters an error (e.g., model convergence failure, infeasible optimization), a descriptive alert guides the user to adjust filters or parameters.

The optimization problem aims to allocate available shelf capacity, where Equation 1 shows the objective function to maximize profit and satisfies the space and demand constraints:

$$\max Z = \sum_{i=1}^n p_i x_i \quad (1)$$

Where: P_i represents the profit contribution of product i , x_i represents allocated shelf space.

The equation 2 and 3 gives shelf capacity constraint and demand constraints:

$$\sum_{i=1}^n s_i x_i \leq S \quad (2)$$

Where: s_i is shelf size required by product i , S is total available shelf capacity.

$$x_i \leq d_i \quad (3)$$

3.1. Framework diagram

Figures 2 and 3 present the overall framework, which is organized into four logical layers:

- Ingestion Layer: A single CSV upload endpoint reads raw sales and product data, standardizes formats, and computes derived fields (e.g. Shelf Area).
- Shared Data Cache: The cleaned DataFrame is held in memory once, ensuring that all downstream modules operate on an identical, preprocessed dataset without redundant I/O.
- Module Layer:
 - Dashboard: Enables interactive analytics with charts, tables, and key performance indicators (KPIs).
 - Forecast: Projects demand using Prophet model and synthetic simulation techniques.
 - Optimize: Generates planograms using optimization algorithms like Linear Programming (LP), Genetic Algorithm (GA), and Proximal Policy Optimization (PPO).
- Output Layer: A unified “Download” control exposes module-specific exports filtered CSVs, charts,

simulation metrics, or planogram files directly to the user.

4. Implementation

This section emphasizes how dynamic shelf space allocation problems have been addressed through modular design, reactive user interfaces, and advanced optimization techniques. End-to-end workflow is examined from data ingestion and interactive visualization to forecast and multi-method allocation engines, highlighting performance, scalability, and user experience considerations.

4.1. System Architecture Overview

At startup, app.py establishes the application shell, configures global CSS, and initializes Streamlit’s session state to manage page navigation. A centralized router handles three core pages: Dashboard, Forecast, and Optimization ensuring a seamless multipage experience without external dependencies. All shared utilities (data loaders, parsers, filters) reside in utils/helpers.py, promoting DRY principles and enabling consistent behavior across modules. Key architectural patterns include:

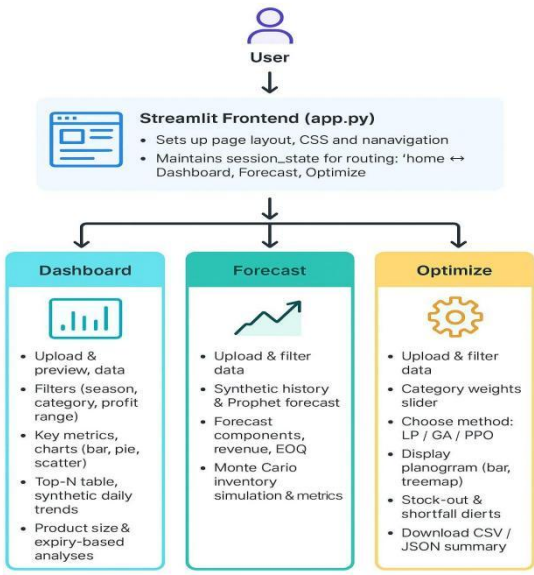


Fig. 2. Framework diagram of the Shelf Space Optimization System

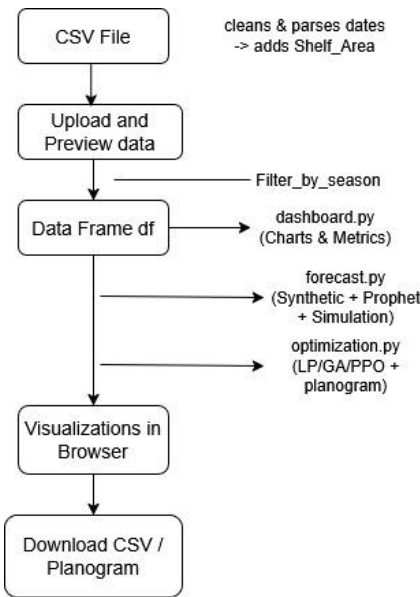


Fig. 3. Data Flow Diagram of Modules and Data Stores

- **Reactive UI Model:** Leveraging Streamlit's rerun-on-widget-change semantics, each user interaction triggers an efficient diff-based re-execution, preserving application state and minimizing latency.
- **Session State Management:** Distinct session keys (e.g., `dashboard_season`, `forecast_season`, `opt_season`) isolate user inputs per module, preventing cross-page collisions and enabling true multi-tab support.
- **Modular Utilities:** `upload_and_preview_data()` and `filter_by_season()` encapsulate data validation, type coercion (dates, numeric fields), and basic feature engineering (e.g., computing `Shelf_Area`). This centralization simplifies maintenance and ensures consistency.

4.2. Dashboard Module

The Dashboard module (`pages/dashboard.py`) transforms raw sales data into actionable insights within a single, dynamic interface:

- Data Ingestion and Preprocessing:** Users upload a CSV via Streamlit's `file_uploader`. Internally, the helper functions:
 - Reads data with Pandas, handling common encodings and CSV quirks.
 - Parses `Expiry_Date` across multiple date formats, coercing invalid entries to NaT and filtering them out.
 - Computes derived features such as `Shelf_Area` and validate numeric columns, issuing warnings for missing critical fields.

This early validation loop guarantees that downstream filters and charts operate on a clean, predictable DataFrame.

- Dynamic Filtering Pipeline:** Three sequential filters refine the dataset:
 - **Season Selection:** A selectbox triggers `filter_by_season()`, which uses case-insensitive string matching to accommodate user input variations.
 - **Category Multi-Select:** Available categories are derived from the seasonally filtered set, ensuring relevancy. The multiselect component allows multi-category comparison and automatically synchronizes with season changes.
 - **Profit Range Slider:** Using the filtered DataFrame's min/max values, the slider bounds update dynamically, preventing user selection of out-of-range values.

- Visualization and Metrics:**

Key Performance Indicators: A three-column layout leverages `st.columns` to display:

- **Total Sales (30d):** Sum of `Sales_Last_30_Days` via vectorized operations.
- **Avg Profit/Unit:** Computed as the mean of `Profit_Per_Unit` to two decimal places.
- **Total Profit (30d):** Product sum of sales and profit columns.

- Interactive Charts:** Plotly powers three main visuals:

Sales by Category, Sales vs. Profit Scatter and Top N Products Table.

- Synthetic Daily Trends:** Triggered by a checkbox, synthetic time series generation

4.3. Supply Forecast Module

The Forecast module extends the Dashboard by projecting demand and guiding inventory decisions:

- Synthetic History and Model Training:**

Synthetic History Generation: Users can adjust volatility via a slider. We generate 30-day demand traces using NumPy's random normal distribution, anchoring on average daily sales to illustrate stochastic demand scenarios.

ProphetIntegration: Instantiating `Prophet(daily_seasonality=True)`,

Indian holidays are added (`model.add_country_holidays(country_name='IN')`) to capture cultural seasonality. The model is fitted on the synthetic series inside a Streamlit spinner, ensuring responsive feedback during training.

- Forecast Visualization and Components:**

- **Forecast vs. History Plot:** Overlaid scatter (history) and line (forecast) traces highlight model performance.
- **Component Decomposition:** Using `model.plot_components()`, users explore trends, weekly seasonality, and holiday impact, illustrating how each factor contributes to predicted demand.

4.4. Revenue Forecast

When profit data is available, revenue is computed for each forecasted day and visualized as a bar chart, with cumulative revenue shown via `st.metric`.

Demand Planning Metrics: Safety stock and reorder points are computed using the standard formula below:

$$SS = z \times \sigma \sqrt{LT} \text{ and } ROP = \mu \times LT + SS \quad (4)$$

where z is the service-level z -score and LT is lead time. EOQ is calculated as $\sqrt{\frac{2DS}{H}}$ demonstrating classical inventory management principles. Interactive Monte Carlo Inventory Simulation: Within an expander, users configure initial inventory and run count. The `simulate_inventory()` routine executes multiple demand vs. inventory simulations, returning:

- Avg. Stockouts per Sim: Mean count of days with unmet demand.
- Avg. Ending Inventory: Typical on-hand stock at horizon's end.
- Histograms and Trajectories: Visual distributions and sample path plots, aiding risk assessment and safety stock calibration.

4.5. Shelf Space Optimization Module

Addressing the dynamic shelf space allocation problem, the Optimization module computes profit-maximizing placements under capacity constraints:

- i. Priority Weighting and Capacity: Users assign weights to each category (defaulted via normalized seasonal sales) to capture strategic priorities. The `total_space` slider sets the hard shelf unit limit, instantly triggering a re-execution of the allocation algorithms.
- ii. Optimization Engines:
 - a. Linear Programming (LP): Formulated as:

$$\begin{aligned} & \max \sum_i w_i p_i x_i \\ & s. t \sum x_i \leq C, x_i \geq 0, x_i \in Z \end{aligned} \quad (5)$$

where w_i is the category weight, p_i is unit profit, and C is capacity. Solved via `optimize_lp()`.

- b. Genetic Algorithm (GA): A population of allocation vectors evolves over generations using crossover and mutation operators, balancing exploration and exploitation. Progress logs and convergence metrics are displayed to the user.
- c. PPO (Reinforcement Learning): An RL agent trained using Proximal Policy Optimization treats allocation as a sequential decision task. The `optimize_ppo_speedup()` function runs multiple parallel environments for sample efficiency, and training progress is conveyed via a spinner and final policy evaluation metrics.

Result Visualization and Alerts: Regardless of the chosen method, results are fed into `display_planogram_interactive()`, producing:

- Category-Level Bars: Summarize total units per category.
- Product-Level Bars: Detail individual allocations, color-coded by category.
- Treemap View: Reveals proportional usage, providing both macro and micro perspectives.

Additionally, a post-allocation alert system: Computes Ideal Allocation vs. actual allocations to identify shortfall units. Aggregates Lost Revenue per product, sorting by highest impact. Presents interactive tables and charts (bar and pie) to highlight top at-risk products and category-level revenue exposure.

4.6. Performance and Caching Strategies

To maintain responsiveness under heavy computation, we employ several strategies:

- i. Streamlit Caching: Critical functions like synthetic history generator, GA optimizer is decorated with `@st.cache_data` to prevent redundant recomputation when inputs remain unchanged.
- ii. Lazy Loading: Large DataFrames and heavy computations are deferred until explicit user actions, minimizing initial load times and memory usage.
- iii. Asynchronous Feedback: Spinners (via `st.spinner()`) and progress bars inform users of long-running tasks, keeping the UI responsive and preventing lockups.

The implementation weaves together dynamic data ingestion, interactive visualization, probabilistic forecasting, and multi-method optimization into a coherent, user- centric application. By decoupling modules, applying rigorous caching, and providing real-time responsiveness, a powerful tool for dynamic shelf space allocation is delivered that scales from small boutiques to enterprise operations.

5. Results

5.1. Experimental Setup

This section presents the experimental results and comparative evaluation of different shelf space optimization techniques, namely PPO, GA, and LP, under different seasonal constraints.

- Platform: Python 3.10 with Streamlit interface.
 - Libraries: NumPy, Pandas, Plotly, Prophet, DEAP, SciPy, and PyTorch.
 - Frontend: Interactive dashboard allowing seasonal input, category weight adjustment, and visual analytics.
 - Shelf Units Constraint: Fixed at 5000 units for all experiments.
- i. Seasonal Filtering and Performance Insights: Figure 4 illustrates how users can upload the retail dataset and select a specific season for analysis. Products can be filtered by category and a profit-per-unit slider to focus on target margins. The dashboard displays key performance indicators such as total sales, average profit per unit, and overall profit, supporting seasonal and category-based decision-making.
- ii. Sales Visualization by Category: Figure 5 presents a pie chart depicting the distribution of unit sales across categories. This visualization allows for a quick understanding of which categories dominate sales and offers a high- level comparison useful for inventory and marketing decisions.

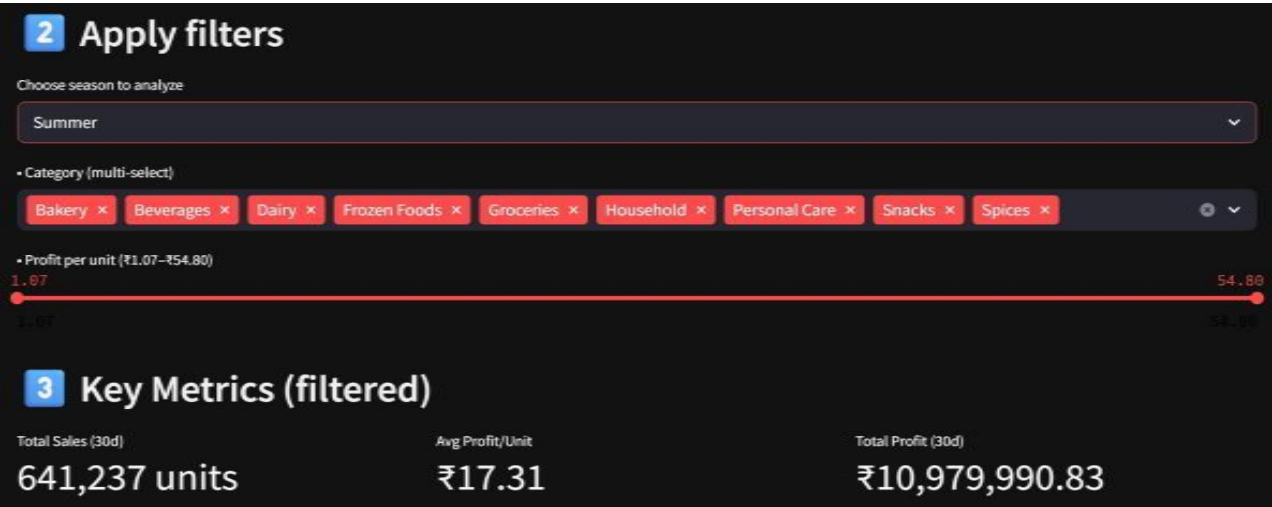


Fig. 4. Seasonal Filtering and Performance Insights (Summer, 5000 Rows)

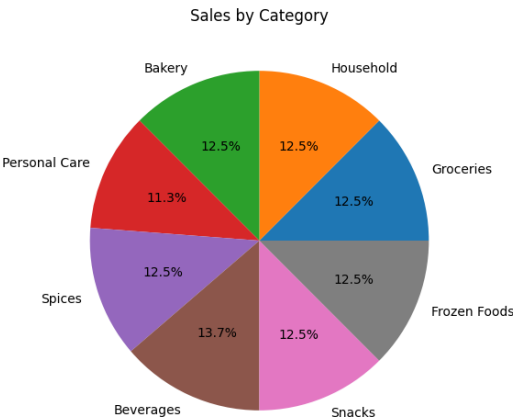


Fig. 5. Pie Chart Showing Proportion of Units Sold by Product Category (Filtered by Season and Category)

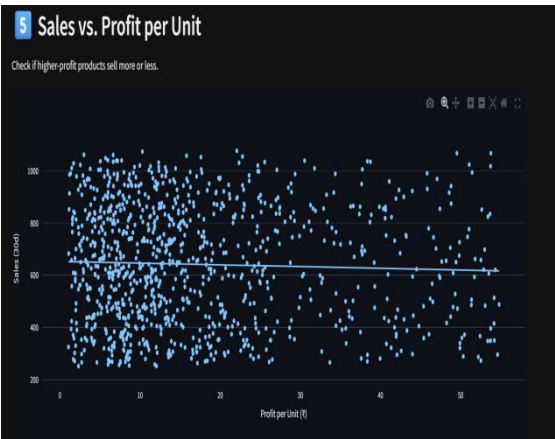


Fig. 6. Scatter Plot – Profit per Unit vs. Sales (Last 30 Days)

iii. Sales vs. Profit per Unit Scatter Plot Analysis: Figure 6 shows a scatter plot used to analyze the relationship between profit per unit and sales volume over the last 30 days. Each point represents a product, and tooltips reveal additional details on hover. A trendline assists in identifying pricing strategies and performance clusters.

Table 1. Top N Products Ranked by Sales (Last 30 Days)

Product Name	Category	Sales Last 30 Days	Profit Per Unit
MDH Urban Cumin	Spices	1076	22.07
Tropicana Vibrant Juice	Beverages	1073	10.3
Britannia Pure Cake	Bakery	1067	53.77
McCain Deluxe Nuggets	Frozen Foods	1067	49.53
Head & Shoulders Fresh Shampoo	Personal Care	1061	3
Lysol Savory Disinfectant	Household	1060	6.64
Ritz Classic Crackers	Snacks	1057	15.8
Green Giant Royal VegMix	Frozen Foods	1054	22.78
Ritz Royal Crackers	Snacks	1052	17.6
Tropicana Urban Juice	Beverages	1048	5.7

iv. Top N Products by Sales: Table 1 displays a ranked list of top-performing products based on sales in the past 30 days. A slider allows dynamic selection of N (e.g., Top 5, Top 1), enabling retailers to quickly spot high-volume items and optimize inventory and placement accordingly for N=18.



Fig. 8. Synthetic 30-Day Sales Trend for Selected Product

v. Synthetic Daily Sales Trend for Selected Product: Figure 8 illustrates a simulated 30-day sales trend for a chosen product, generated using average daily sales with random fluctuations. This aids in visualizing demand cycles even when real-time sales data is sparse or unavailable.

5.2. Explanation of Supply Forecast section

- i. Season and Product Selection with Key Metrics: The Forecast module lets users upload data, pick a season, and select a product from the filtered list. It then shows key metrics like total sales in the last 30 days and average daily sales, providing tailored forecasts based on season and product trends as shown in figure 9.

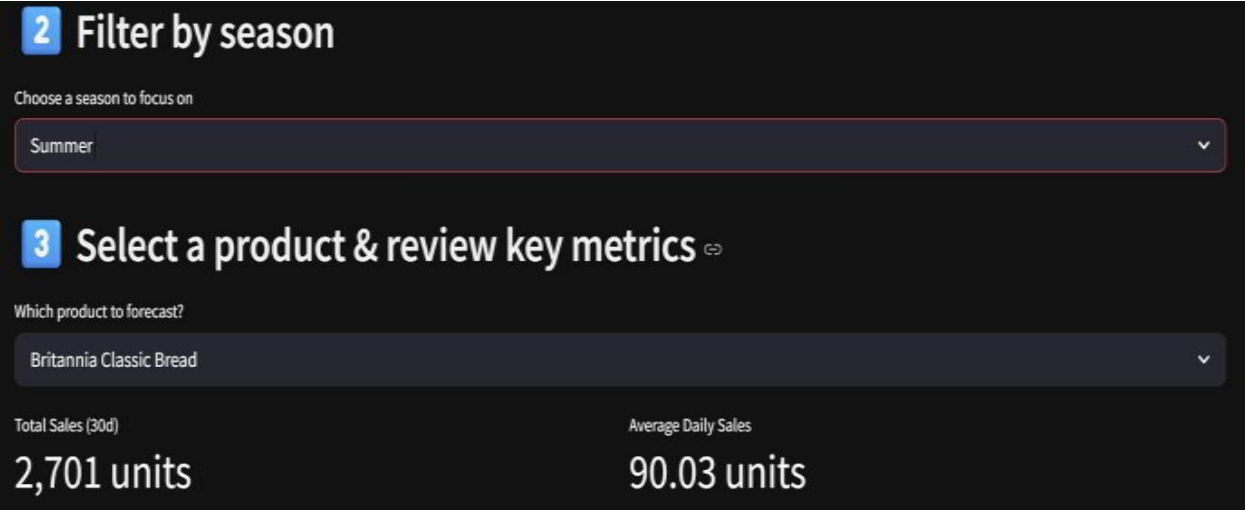


Fig. 9. Key Metrics of Britannia Classic Bread(Summer,5000rows)

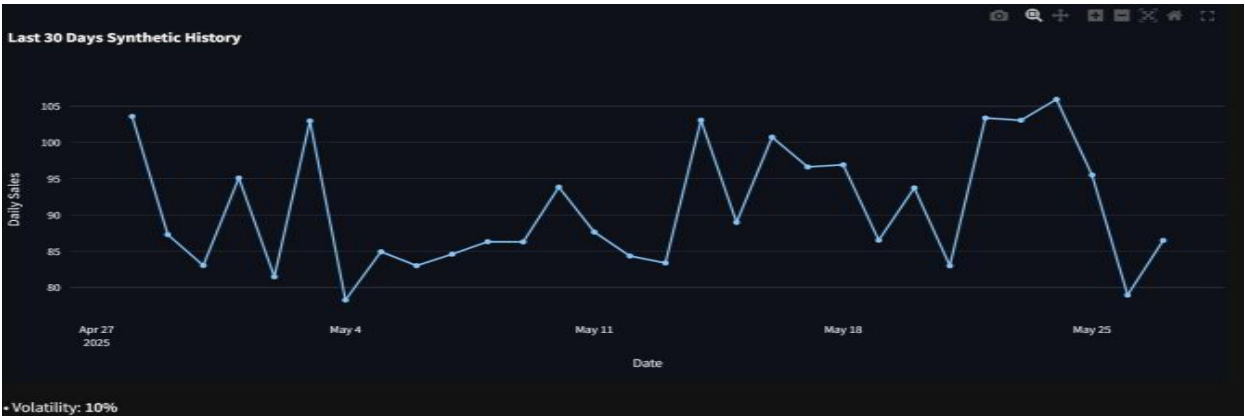


Fig. 10. Synthetic 30-day sales graph with realistic fluctuations showing recent trends

- ii. Forecast settings: Configure the forecast parameters to simulate future sales. Two key settings are provided: forecast horizon, which determines the number of days to predict ahead (ranging from 7 to 60 days), and volatility, which adjusts the variability of the synthetic sales history. These inputs allow users to explore different forecasting scenarios and understand potential fluctuations in product demand.
- iii. Generate synthetic 30-day history: Figure 10 generates a 30-day synthetic sales history by adding controlled noise to average daily sales, simulating realistic fluctuations. It helps visualize past sales variability, enhancing forecast accuracy.
- iv. Fit Prophet and forecast for the next 15 days: The Prophet forecasting model is trained on the synthetic 30-day sales history to capture underlying trends and seasonality, including country-specific holidays. Using this model, a 15-day forecast is generated, predicting future sales and highlighting short-term demand patterns to assist in inventory planning.
- v. Revenue Forecast: This section calculates and visualizes the forecasted revenue over the selected horizon by multiplying predicted sales with the product's profit per unit. The cumulative revenue metric provides a quick overview of expected earnings, while the bar chart shows daily revenue trends, aiding financial planning and decision-making.

5.3. Optimization Techniques

While three algorithms—LP, GA, and PPO are implemented, this section displays results from the PPO algorithm, focusing solely on maximizing overall profit.

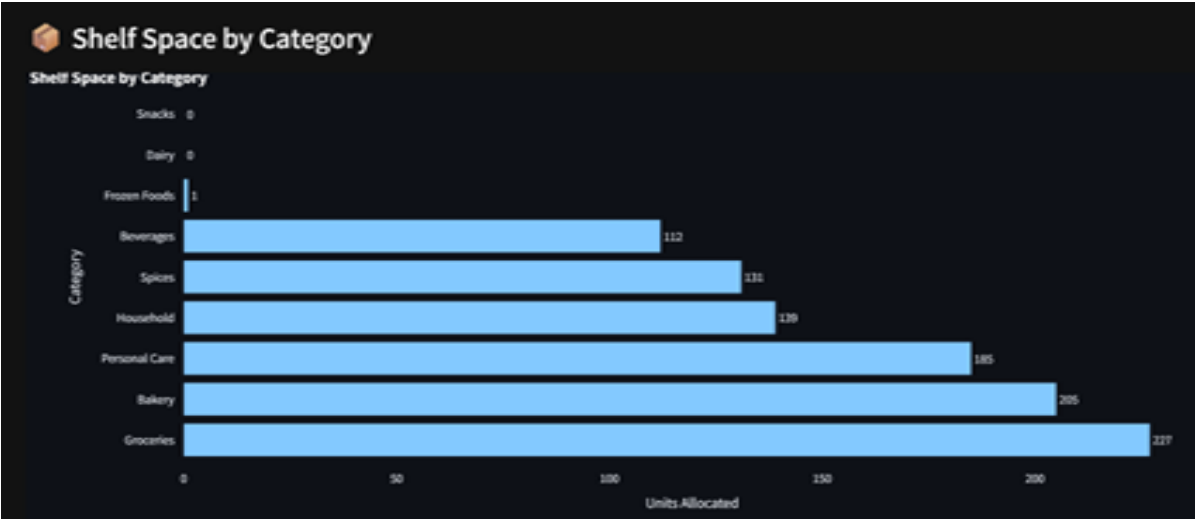


Fig. 11. Shelf space allocated to each category

- i. Shelf Space by Category: Figure 11 visualizes the total shelf space allocated to each product category after optimization. The horizontal bar chart ranks categories by their allocated units, helping users quickly see which categories are prioritized in the shelf layout. This insight supports informed decisions on inventory distribution and space management aligned with business goals.
- ii. Shelf Space by Product: Figure 12 presents a horizontal bar chart showing shelf space allocation for individual products, sorted by allocated units. Each bar is color-coded by category, with profit per unit available on hover. This view helps identify which products are prioritized on shelves, supporting detailed inventory and space management decisions.

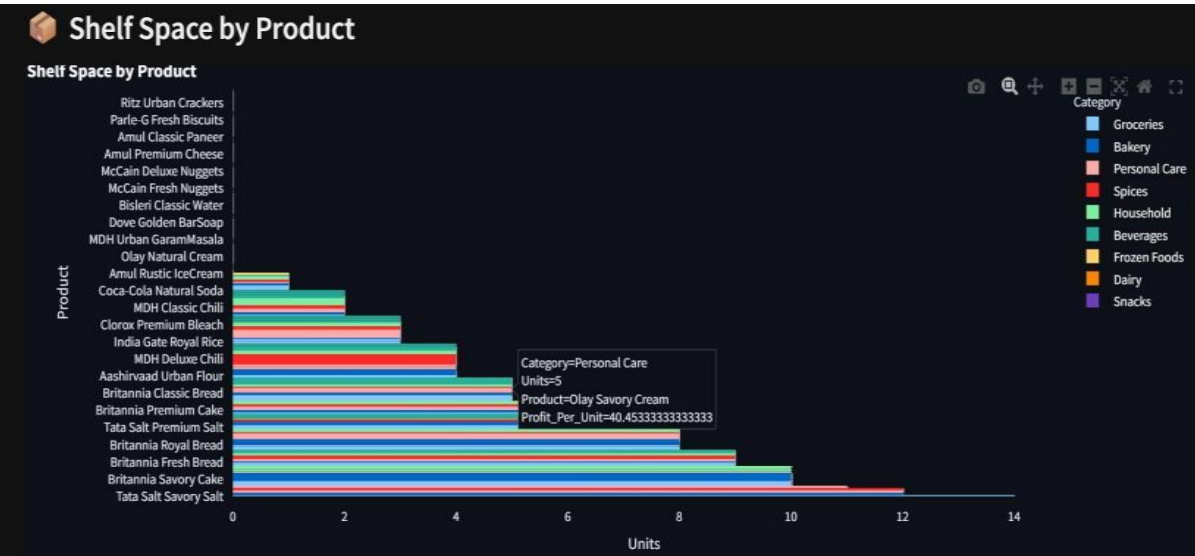


Fig. 12. Shelf space allocated to each product

- iii. Proportional View (Treemap): Figure 13 visualizes shelf space allocation hierarchically by category and product. Categories are sized by total allocated units, with average category profit shown on hover. Individual products are nested within categories, sized by their allocated space and annotated with their profit per unit. This view offers an intuitive snapshot of how shelf space and profitability are distributed across the product range.
- iv. Potential Stock-Out Alerts: Figure 14 highlights products at risk of stock-outs due to insufficient shelf allocation. If no risks are detected, it confirms all products are well stocked. Otherwise, it displays the number of at-risk products and estimated lost revenue, with an option to preview detailed data for top products based on potential revenue loss. The system displays the top 4 products with the highest potential revenue loss, allowing users to preview detailed data.
- v. Top 5 Lost-Revenue Shortfalls: Figure 15 presents a horizontal bar chart of the top five products with the highest potential revenue loss due to under- allocation. It visually highlights which stock-outs could impact

- revenue the most, helping prioritize corrective actions.
- vi. **Category Share of Lost Revenue:** The pie chart in Figure 16 shows how lost revenue from stock shortfalls is distributed across categories, based on the top 5 products with the highest shortfall impact. Users can adjust the number of products *n* to analyze, helping identify which categories contribute most to potential revenue loss and refine stocking strategies.



Fig. 13. Shelf Allocation Treemap — PPO Algorithm (Summer, 5000 Rows)



Fig. 14. Displaying number of products at risk, total lost revenue, top products at risk of stock-out.

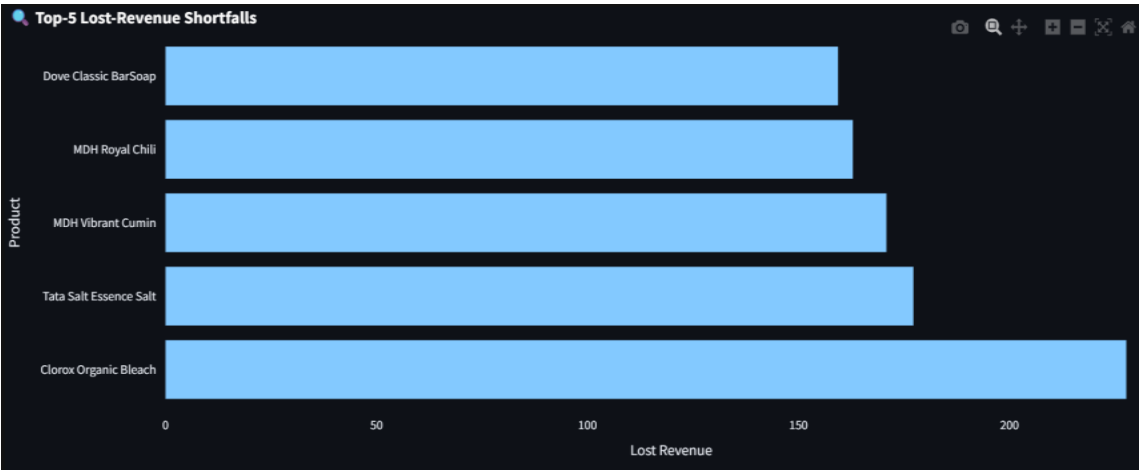


Fig. 15. Top 5 Lost-Revenue Shortfall Products (PPO, Summer)

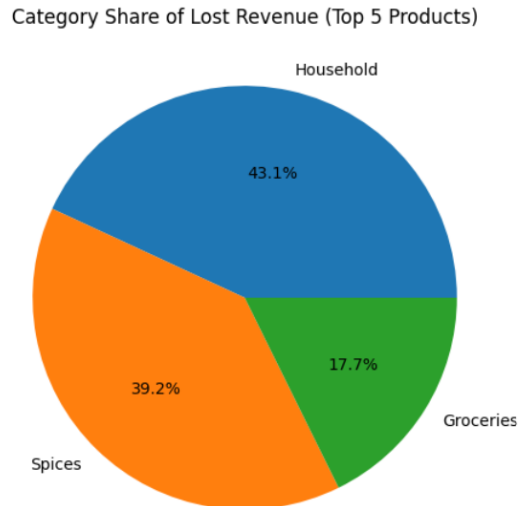


Fig. 16. Category Share of Lost Revenue Top 5 Products (PPO, Summer)

5.4. Evaluation Metrics

- Estimated Profit (): Total projected profit from allocated products.
- Products at Risk: Number of products with demand exceeding supply (stock-out risk).
- Lost Revenue (): Profit loss due to stock-outs.

As shown in Table 2, depending on the evaluation criterion, the relative performance of LP, GA, and PPO varies across seasonal scenario. PPO achieved the highest profit-oriented allocations during the Monsoon season, indicating its ability to adapt shelf-space decisions according to seasonal demand patterns. Furthermore, this profit improvement was accompanied by higher stock-out risk compared to GA. In contrast, GA exhibited more conservative allocation behavior and achieved lower or zero stock-outs during the winter season, resulting in improved inventory stability. LP produced the highest profit in certain summer while maintaining moderate risk levels. These results suggest that no single algorithm consistently dominates across all performance metrics; rather, the choice of optimization method depends on whether the retailer prioritizes profit maximization, inventory availability, or a balance between the two objectives.

Table 2. Optimization comparison across methods and seasons

Method	Season	Shelf units	Rows	Estimated Profit()	Product at Risk	Last Revenue()
PPO	Monsoon	5000	10000	144053.75	310	33277.00
GA	Monsoon	5000	10000	118486.67	1	26.13
LP	Monsoon	5000	10000	Infeasible	Infeasible	Infeasible
GA	Winter	5000	10000	113805.76	0	0.00
PPO	Winter	5000	10000	137700.12	305	31956.16
PPO	Summer	5000	5000	182006.13	152	35872.99
GA	Summer	5000	5000	145714.99	-	-
LP	Summer	5000	5000	208722.89	82	11154.96

Insights and Observations

- Proximal Policy Optimization achieves a good balance between profit maximization and stock-out risk minimization. Its reinforcement learning framework allows the model to adapt dynamically to varying data scales, seasonal demand patterns, and complex constraints. In high-volume datasets (e.g., 10,000 rows), PPO consistently outperforms heuristic methods in finding near-optimal allocations while minimizing critical product shortages.
- Linear Programming demonstrates the highest theoretical profitability under ideal, feasible constraint sets. However, it suffers significantly in scenarios with overlapping or conflicting constraints such as the Monsoon season case, where it becomes mathematically infeasible to allocate space while satisfying all defined conditions. This limits LP's applicability in real-world, high-dimensional, and dynamic retail environments.
- Genetic Algorithm proves to be highly effective for moderate-sized datasets and seasonal variations. Its population-based search process, inspired by natural evolution, helps escape local optima and find diverse, valid solutions. GA is especially robust under uncertainty and shows minimal product stock-outs, making it a reliable choice when model interpretability and feasibility are prioritized over maximum theoretical profit.

6. Conclusion

Dynamic shelf space allocation remains a critical challenge for modern retailers, driven by fluctuating demand, limited shelf area, and seasonal variation. Three distinct optimization approaches were implemented and evaluated: (1) Proximal Policy Optimization (PPO), a reinforcement learning method that dynamically learns allocation policies by maximizing cumulative rewards based on forecasted demand and profit; (2) Genetic Algorithm (GA), a robust heuristic approach well-suited for medium-scale datasets that effectively balances exploration and exploitation; and (3) Linear Programming (LP), a classical strategy that yields high profits under feasible constraints but struggles with infeasibility in over-constrained or highly variable scenarios. The proposed work in the paper addressed key retail constraints including limited shelf capacity, product demand, and profitability objectives. Experimental evaluation demonstrated that LP provided efficient deterministic allocation under fixed constraints, while GA improved search flexibility for complex optimization scenarios. PPO further enabled adaptive decision-making by learning allocation strategies dynamically from the retail environment. Further, we can enhance by including real world scenarios, which include larger datasets, repeated trials to enable statistical validation.

The developed system supports practical retail planning through visualization and real-time allocation analysis, enabling better utilization of shelf space and improved profit-oriented decision-making. The comparative analysis highlights the strengths of combining classical and reinforcement approaches for modern retail applications. Future work may focus on incorporating real-time sales data, customer behaviour patterns, and additional operational constraints to further improve adaptive shelf allocation performance in large-scale retail settings.

All the Declarations and Statements

Author Contributions Statement

Anitha Palakshappa – Conceptualization and Supervision: Constructed the overall framework, and supervised project execution. Review and Editing, and Project Management. Reviewed and edited the manuscript, ensured clarity and coherence, and helped coordinate project milestones and deadlines.

Shruti J. R. – Writing – Drafted the initial manuscript, contributed to the literature survey.

Sowmya Kyatanahalli Nanjappa – Model Training, and Validation,

Ashwitha Anni – Documented the technical background of the study.

Aditya Gaonkar – Data Curation and Software Implementation: Handled data acquisition, dataset preprocessing, and implementing the research model.

Aditya Gaonkar – Performed in-depth analysis of experimental results, prepared performance charts, and ensured the statistical robustness of the evaluation.

All authors have read and agreed to the published version of the manuscript.

Conflict of Interest Statement

The authors declare no conflicts of interest.

Funding Declaration

This research received no external funding.

Data Availability Statement

This study analyzed publicly available datasets. Our application relies on a comprehensive product sales dataset comprising 10,005 records of consumer goods variants tailored for the Indian retail context. A synthetic retail dataset containing 10,005 product-variant records was generated to emulate the characteristics of Indian FMCG retail environments. Product names and categories were inspired by commonly available consumer goods, while seasonal attributes, sales volumes, and profit margins were generated within realistic ranges to facilitate comparative evaluation of shelf-space optimization algorithms.

The dataset will be made available on request.

Ethical Declarations

The authors should declare the following: There is no human subjects or animals used for the study.

Acknowledgments

The authors would like to thank Ramaiah Institute of Technology for the support to conduct the research work. We sincerely thank the experts for their professional evaluation and valuable recommendations, which have contributed to

improving the quality of the experiment and the reliability of its results.

Declaration of Generative AI in Scholarly Writing

Authors used AI tool to improve their language by avoiding any grammatical mistakes in writing the manuscript.

Abbreviations

The following abbreviations are used in this manuscript:

LP – Linear Programming

PPO – Proximal policy optimization

GA – Genetic Algorithm

KPIs - Key performance indicators.

Appendix A/B/C..., with appendix title

None.

References

- [1] Frazzon, E. M., Rodriguez, C. M. T., Pereira, M. M., Pires, M. C., & Uhlmann, I. (2019). Towards supply chain management 4.0. *Brazilian Journal of Operations & Production Management*, 16(2), 180-191. doi: 10.14488/BJOPM.16.2.180.2019.
- [2] F. M. Isa, W. N. M. Ariffin, and M. S. Jusoh, "Shelf space allocation problem (SSAP) in the retail industry: A systematic literature review," *International Journal of Business Society*, vol. 25, no. 3, 2024.
- [3] H. Riazi, M. Doroodian, and B. Afshar-Nadjafi, "Improving the sales process of profitable perishable goods: an inventory control strategy in a planned economy," *International Journal of Retail Distribution Management*, vol. 52, no. 6, pp. 721–735, 2024. doi: 10.1108/IJRDM-03-2023-0104.
- [4] M. Afsahi, "A bi-level, bi-objective optimization model integrating promotion, assortment, shelf-space, and inventory replenishment in supermarkets," *Assortment, Shelf-Space, and Inventory Replenishment in Supermarkets*, 2024.
- [5] M. Ostermeier, T. Du"sterho"ft, and A. Hu"bner, "A model and solution approach for store-wide shelf space allocation," *Omega*, vol. 102, p. 102425, 2021. doi: 10.1016/j.omega.2021.102425.
- [6] C. Gencosman and M. A. Begen, "Exact optimization and decom- position approaches for shelf space allocation," *European Journal of Operational Research*, vol. 299, no. 2, pp. 432–447, 2022. doi: 10.1016/j.ejor.2021.12.028.
- [7] K. Czerniachowska, "A genetic algorithm for the retail shelf space allocation problem with virtual segments," *Opsearch*, vol. 59, no. 1, pp. 364–412, 2022. doi: 10.1007/s41269-021-00276-0.
- [8] C. Murray, D. Talukdar, and A. Gosavi, "Joint optimization of product price, display orientation and shelf-space allocation in retail category management," *Journal of Retailing*, vol. 86, no. 2, pp. 125–136, 2022. doi: 10.1016/j.jretai.2019.03.004.
- [9] S. J. Sajadi and A. Ahmadi, "An integrated optimization model and metaheuristics for assortment planning, shelf space allocation, and inventory management of perishable products: a real application," *PLOS ONE*, vol. 17, no. 3, p. e0264186, 2022. doi: 10.1371/journal.pone.0264186.
- [10] K. Czerniachowska and M. Hernes, "A heuristic approach to shelf space allocation decision support including facings, capping, and nesting," *Symmetry*, vol. 13, no. 2, p. 314, 2021. doi: 10.3390/sym13020314.
- [11] Frontoni, F. Marinelli, R. Rosetti, and P. Zingaretti, "Shelf space re-allocation for out of stock reduction," *Computers & Industrial Engi- neering*, vol. 106, pp. 32–40, 2017. doi: 10.1016/j.cie.2017.02.007.
- [12] T. Flamand, A. Ghoniem, and B. Maddah, "Promoting impulse buying by allocating retail shelf space to grouped product categories," *Journal of the Operational Research Society*, vol. 67, no. 7, pp. 953–969, 2016. doi: 10.1057/jors.2015.103.
- [13] K. Higuchi and K. Takeyasu, "Optimization in allocating goods to shop shelves utilizing genetic algorithm under expanded shelf position case," *Journal of Computations & Modelling*, vol. 6, no. 2, pp. 15–31, 2016.
- [14] S. S. Rautaray, M. Pandey, S. Chakraborty, and K. A. Barua, "Proposal for shelf placement optimization for retail industry using big data analytics," in *Data Science Congress*, Mumbai, India, 2017.
- [15] Y. Duan, Z. Mao, and J. Huo, "Introduction of store brands considering product cost and shelf space opportunity cost," *Mathematical Problems in Engineering*, pp. 1–19, 2018. doi: 10.1155/2018/5092476.
- [16] H. K. Gajjar and G. K. Adil, "Retail shelf space allocation considering inventory replenishment," *International Journal of Services and Operations Management*, vol. 22, no. 2, pp. 221–234, 2015. doi: 10.1504/IJSOM.2015.069401.
- [17] H. N. Geismar et al., "Maximizing revenue through two-dimensional shelf-space allocation," *Production and Operations Management*, vol. 24, no. 7, pp. 1148–1163, 2015. doi: 10.1111/poms.12326.
- [18] Khan, T., & Emon, M. M. H. (2025). The role of digital supply chain practices in enhancing firm performance: insights from the manufacturing sector of Bangladesh. *Brazilian Journal of Operations & Production Management*, 22(2), 2493 . <https://doi.org/10.14488/BJOPM.2493.2025>. doi: 10.14488/BJOPM.2493.2025.
- [19] Hübner, A., & Schaal, K. (2017). A shelf-space optimization model when demand is stochastic and space-elastic. *Omega*, 68, 139-154. doi: 10.1016/j.omega.2016.08.007.

Authors' Profiles



Anitha Palakshappa received her Ph.D in Faculty Computer and Information Sciences from Visvesvaraya Technological University in the area of Data Analytics in 2023. Currently working as an Associate Professor in the Department of Information Science and Engineering at Ramaiah Institute of Technology Bengaluru. Her research areas of interest are data mining and analytics, supply chain analytics, IoT, machine learning, and artificial intelligence.



Shruti J. R. is currently working in the Dept of ISE, Assistant Professor in the Department of Information Science and Engineering at Ramaiah Institute of Technology Bengaluru. She is pursuing her Ph.D Degree. Her research areas of interest are Machine Learning, Block Chain, Java programming, Devops.



Sowmya Kyathanahalli Nanjappa received her Ph.D. in Faculty of Computer and Information Sciences from Visvesvaraya Technological University for her work in the area of Video Forensics in 2020. Currently, she is an Associate Professor in the Department of Information Science and Engineering, JSS Academy of Technical Education, Bengaluru. Her research areas of interest are image and video forensics, cyber security, pattern recognition, machine learning, deep learning and computer vision.



Dr. Ashwitha Anni holds a Bachelor of Engineering (B.E.) in ISE, Master of Engineering (M.E) in Software Engineering and Ph.D. in CSE on Machine Learning. She is currently working as an Assistant Professor-Senior Scale in the department of School of Computer Engineering, Manipal Institute of Technology, Bengaluru, Manipal Academy of Higher Education, Manipal, India. Her research areas of interest include data science, artificial intelligence, machine learning, and deep learning. She has published multiple papers in international journals and conferences, since June 2016.



Aditya Gaonkar is a student in the Department of ISE, Ramaiah Institute of Technology Bengaluru. His research interests are Data Analytics, Machine Learning and Artificial Intelligence.



Bhawna Botra is a student in the Department of ISE, Ramaiah Institute of Technology Bengaluru. His research interests are Data Analytics, Machine Learning and Artificial Intelligence.

How to cite this paper

Anitha Palakshappa, Shruti J. R., Sowmya Kyathanahalli Nanjappa, Ashwitha Anni, Aditya Gaonkar, Bhawna Botra, "Shelf Space Optimization in Retail: A Study Using Linear Programming, Genetic Algorithm, and Proximal Policy Optimization", International Journal of Education and Management Engineering (IJEME), Vol.16, No.4, pp. 45-60, 2026. DOI:10.5815/ijeme.2026.04.04.